

**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

**CARRERA:
INGENIERÍA DE SISTEMAS**

**Trabajo de titulación previo a la obtención del título de:
INGENIERA E INGENIERO DE SISTEMAS**

**TEMA:
ANÁLISIS, DISEÑO Y DESARROLLO DE UN AMBIENTE VIRTUAL DE
APRENDIZAJE ACCESIBLE PARA ESTUDIANTES CON DISCAPACIDAD
VISUAL PARA LA UNIVERSIDAD POLITÉCNICA SALESIANA, SEDE
QUITO.**

**AUTORES:
ESTEFANI LORENA HERRERA HERRERA
HOLGER RICARDO FIGUEROA QUINGA**

**TUTOR:
GUSTAVO ERNESTO NAVAS RUILOVA**

Quito, julio del 2019

CESIÓN DE DERECHOS DE AUTOR

Nosotros: Estefani Lorena Herrera Herrera, con documento de identificación N° 1724397938, y Holger Ricardo Figueroa Quinga, con documento de identificación N° 1720204724, manifestamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de titulación con el tema: “ANÁLISIS, DISEÑO Y DESARROLLO DE UN AMBIENTE VIRTUAL DE APRENDIZAJE ACCESIBLE PARA ESTUDIANTES CON DISCAPACIDAD VISUAL PARA LA UNIVERSIDAD POLITÉCNICA SALESIANA, SEDE QUITO.”, mismo que ha sido desarrollado para optar por el título de: INGENIERA E INGENIERO DE SISTEMAS, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en nuestra condición de autores nos reservamos los derechos morales de la obra antes citada. En concordancia, suscribimos este documento en el momento que hacemos entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.



ESTEFANI LORENA
HERRERA HERRERA
C.I.: 1724397938



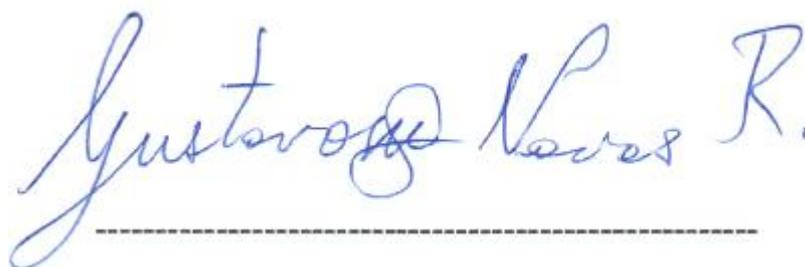
HOLGER RICARDO
FIGUEROA QUINGA
C.I.: 1720204724

Quito, julio de 2019

DECLARATORIA DE COAUTORÍA DEL AUTOR

Yo declaro que bajo mi dirección y asesoría fue desarrollado el trabajo de titulación, con el tema: ANÁLISIS, DISEÑO Y DESARROLLO DE UN AMBIENTE VIRTUAL DE APRENDIZAJE ACCESIBLE PARA ESTUDIANTES CON DISCAPACIDAD VISUAL PARA LA UNIVERSIDAD POLITÉCNICA SALESIANA, SEDE QUITO, realizado por Estefani Lorena Herrera Herrera y Holger Ricardo Figueroa Quinga, obteniendo un producto que cumple con todos los requisitos por la Universidad Politécnica Salesiana, para ser considerados como trabajo final de titulación.

Quito, julio de 2019



Handwritten signature in blue ink: Gustavo Ernesto Navas R.

GUSTAVO ERNESTO NAVAS RUILOVA

C.I.:1705675625

DEDICATORIA

Dedico este proyecto de titulación a las personas más importantes en mi vida, empezando por Dios, por bendecirme en cada paso que he dado.

A mis padres Libia Herrera y Mario Herrera, por su amor, trabajo y sacrificio, gracias a ustedes quienes han velado por mi bienestar y educación siendo mi apoyo en todo momento, enseñándome que no debo rendirme ante cualquier obstáculo.

Dios ha puesto en mi camino a los mejores padres del mundo. Los amo con mi vida.

Asimismo agradezco a mis hermanos Paul Herrera y Evelyn Herrera, por brindarme cariño y felicidad, durante todo este proceso.

Finalmente quiero dedicar y expresar mi más grande y sincero agradecimiento a mi Novio Marco Cazar, por tu paciencia, tu apoyo incondicional y por su ayuda en mi trabajo de titulación, gracias por hacer mis días maravillosos, ser mi confidente, me ayudaste a concluir este sueño, te amo.

Estefani Lorena Herrera Herrera

DEDICATORIA

Dedico este proyecto a Dios por bendecirme con la vida y guiar mi camino a lo largo de todo este tiempo, a mis padres Jorge Figueroa y Carmen Quinga ya que, con su trabajo, esfuerzo, consejos y apoyo constante, me encaminaron en esta meta, gracias por motivarme a seguir con mis estudios.

A mi esposa Silvana Pacheco por su apoyo incondicional también a mi hijo Matías Figueroa por su amor, alegría por ser la motivación en mi vida hoy puedo decirte hijo que los hemos logrado.

A todas las personas de mí trabajo que de una u otra manera me apoyaron a concluir esta meta.

Holger Ricardo Figueroa Quinga

AGRADECIMIENTO

Agradecemos a Dios por bendecirnos la vida, por guiarnos a lo largo de nuestra existencia.

De manera especial a nuestro Tutor de Tesis el Ingeniero Gustavo Ernesto Navas Ruilova quien con su experiencia, conocimiento y motivación nos orientó en la elaboración de este proyecto de titulación.

Y por último a todos nuestros docentes que tuvimos durante toda la carrera universitaria que con su sabiduría y apoyo, nos desarrollamos como profesionales.

Un eterno agradecimiento a la UNIVERSIDAD POLITÉCNICA SALESIANA quien ayudo en nuestra preparación profesional para un futuro competitivo.

Estefani Lorena Herrera Herrera

Holger Ricardo Figueroa Quinga

ÍNDICE

INTRODUCCIÓN	1
CAPITULO 1	2
1.1. Antecedentes.....	2
1.2. Descripción del Problema.....	2
1.3. Objetivos.....	2
1.3.1. Objetivo General	2
1.3.2. Objetivos Específicos.....	3
1.4. Justificación.....	3
1.5. Alcance	4
1.5.1. Roles de Usuario.....	5
1.6. Fuera de alcance o suposiciones.....	5
1.7. Marco metodológico.....	5
1.7.1. Scrum.....	6
1.7.1.1. Equipo Scrum.....	6
1.8. Cronograma referencial de desarrollo	7
1.9. Fases de la metodología Scrum	7
1.9.1. Reunión de planificación de Sprint	7
1.9.2. El Scrum Diario.....	8
1.9.3. Trabajo de desarrollo durante el Sprint	8
CAPITULO 2	9
2.1. Accesibilidad Universal.....	9
2.2. Herramientas y tecnología	9
2.3. Gestores de programación	9
2.4. Gestores de base de datos	10
2.5. Api reconocimiento de voz (Speech Recognition)	11
2.6. Sensor Biométrico Digital Persona U.are.u 4500.....	11
CAPÍTULO 3	13
3.1. Análisis de Requerimientos	13
3.1.1. Requerimiento Funcionales.....	13
3.1.1.1. Módulos de la aplicación	14
3.1.1.2. Diagramas de caso de uso	14
3.1.1.2.1. Módulo funcionalidad general de la aplicación	15
3.1.1.3. Módulo chat grupal y sala personal.....	18
3.1.1.4. Módulo Envió de tareas.....	19

3.1.1.5. Módulo Repositorio.....	21
3.2. Costo Estimado.....	24
CAPITULO 4	25
4.1. Modelo Físico de la Base de Datos	25
4.2. Modelo Conceptual de la Base de Datos.....	26
4.2.1. Diccionario de Datos	28
4.3. Diagrama de clases	31
4.4. Diagramas de secuencia	32
4.5. Diagrama de Flujo de Datos.....	41
4.6. Arquetipo de la aplicación.....	46
4.7. Descripción de clases (con código relevante)	49
4.7.1. Módulo Login.....	49
4.7.2. Módulo Chat Grupal y Personal	50
4.7.3. Módulo Tareas.....	55
4.7.4. Módulo Repositorio.....	55
4.7.4.1. Upload.....	55
4.7.4.2. Download	57
4.8. Prototipo final de la aplicación.....	57
CAPITULO 5	59
5.1. Pruebas de Caja Negra.....	59
5.2. Pruebas de Aceptación	63
5.3. Resultados pruebas de aceptación	65
CONCLUSIONES	68
GLOSARIO DE TÉRMINOS	70
REFERENCIAS BIBLIOGRÁFICAS.....	71

ÍNDICE DE FIGURAS

Figura 1. Cronograma de desarrollo.....	7
Figura 2. Cronograma de desarrollo.....	7
Figura 3. Fases metodología scrum.....	8
Figura 4. Digital Persona.....	12
Figura 5. Módulos de la aplicación de un ambiente virtual de aprendizaje.....	14
Figura 6. Caso de uso funcionalidad general de la aplicación.....	15
Figura 7. Caso de uso funcionalidad general de la aplicación.....	16
Figura 8. Caso de uso funcionalidad general de la aplicación.....	17
Figura 9. Caso de uso módulo chat grupal y sala personal.....	18
Figura 10. Caso de uso estudiante envía la tarea.....	19
Figura 11. Docente califica tarea.....	20
Figura 12. Caso de uso módulo repositorio upload.....	21
Figura 13. Caso de uso módulo repositorio download.....	23
Figura 14. Diagrama físico de la base de datos de la aplicación.....	25
Figura 15. Diagrama conceptual de base de datos de la aplicación.....	27
Figura 16. Diagrama de clases de la aplicación.....	31
Figura 17. Diagrama de secuencia ingreso a la aplicación.....	32
Figura 18. Diagrama de secuencia ingreso a la aplicación mediante huella digital...	33
Figura 19. Diagrama de secuencia registro de usuarios.....	33
Figura 20. Diagrama de secuencia modificación de usuarios.....	34
Figura 21. Diagrama de secuencia de cargar módulos.....	34
Figura 22. Diagrama de secuencia módulo chat grupal.....	35
Figura 23. Diagrama de secuencia módulo chat personal.....	36
Figura 24. Diagrama de secuencia estudiante guarda tarea.....	37
Figura 25. Diagrama de secuencia estudiante consulta nota de la tarea.....	38
Figura 26. Diagrama de secuencia docente ingresa la nota.....	39
Figura 27. Diagrama de secuencia upload.....	40
Figura 28. Diagrama de secuencia download.....	41
Figura 29. Diagrama de flujo del funcionamiento de la aplicación.....	42
Figura 30. Diagrama de flujo logeo usuario y opciones de administrador.....	43
Figura 31. Diagrama de flujo opciones de docente.....	44
Figura 32. Diagrama de flujo opciones estudiante.....	45
Figura 33. Pantalla de login de la aplicación.....	46
Figura 34. Pantalla del menú de la aplicación.....	47
Figura 35. Pantalla módulo chat de la aplicación.....	47
Figura 36. Pantalla módulo calificar tarea de la aplicación.....	48
Figura 37. Pantalla módulo manejador de archivos de la aplicación.....	48
Figura 38. Código login verificar huella digital.....	49
Figura 39. Código Identifica huella digital.....	50
Figura 40. Código procesar chat.....	50
Figura 41. Código cargar mensaje chat.....	51
Figura 42. Código reconocimiento de voz.....	52
Figura 43. Código método hablar chat.....	52
Figura 44. Código cambiar idioma chat.....	53
Figura 45. Código método para enviar mensajes chat js.....	53
Figura 46. Código enviar mensaje chat jsp.....	54
Figura 47. Código usuarios conectados.....	54

Figura 48. Código guarda documento.....	55
Figura 49. Código listar archivos.....	56
Figura 50. Código clase upload.....	56
Figura 51. Download.....	57
Figura 52. Pantalla final del login de la aplicación.....	57
Figura 53. Pantalla final menú de la aplicación.....	58
Figura 54. Pantalla final del módulo chat de la aplicación.....	58
Figura 55. Pantalla del módulo manejador de archivos de la aplicación.....	58
Figura 56. Resultados prueba con los usuarios.....	66
Figura 57. Prueba con los estudiantes no videntes.....	66
Figura 58. Prueba con los estudiantes no videntes.....	67

ÍNDICE DE TABLAS

Tabla 1. Características sensor de huella	12
Tabla 2. Caso de uso - Funcionalidad general de la aplicación	15
Tabla 3. Caso de uso - Funcionalidad general de la aplicación administrador- estudiante.....	16
Tabla 4. Caso de uso - Funcionalidad general de la aplicación docente.....	17
Tabla 5. Caso de uso - Módulo chat grupal y sala personal.....	18
Tabla 6. Caso de uso - Módulo tareas estudiante.....	19
Tabla 7. Caso de uso - Módulo tareas	20
Tabla 8. Caso de uso - Módulo repositorio upload	22
Tabla 9. Caso de uso - Módulo repositorio download	23
Tabla 10. Costo estimado	24
Tabla 11. Estudiantes materias.....	28
Tabla 12. Materias.....	29
Tabla 13. Mensaje salas	29
Tabla 14. Tareas	30
Tabla 15. Usuarios	30
Tabla 16. Prueba de caja negra login	59
Tabla 17. Prueba de caja negra módulo registro de usuarios	60
Tabla 18. Prueba de caja negra módulo chat grupal	61
Tabla 19. Prueba de caja negra módulo chat personal	61
Tabla 20. Prueba de caja negra módulo tareas	62
Tabla 21. Prueba de caja negra módulo upload	63
Tabla 22. Prueba de caja negra módulo download	63
Tabla 23. Prueba de aceptación login	64
Tabla 24. Prueba de aceptación chat	64
Tabla 25. Prueba de aceptación tareas	65
Tabla 26. Prueba de aceptación manejador de archivos	65

Resumen

Se desarrolló el prototipo de ambiente virtual AVIDA para la colaboración con los estudiantes con discapacidad visual en su proceso de aprendizaje.

El prototipo se construyó utilizando Java, MySQL, Apache y Netbeans debido a que no se trata de software propietario y por lo tanto no se tendrá problemas con licenciamiento a futuro.

La primera funcionalidad que se entregó en el prototipo es la utilización de la huella digital como método alternativo para el ingreso a la aplicación. Se utilizó el sensor U.are.u 4500 debido a su facilidad de instalación, al ser un dispositivo USB plug&play que es compatible con el sistema operativo Windows desde la versión 2003.

La segunda funcionalidad que posee el aplicativo es el reconocimiento de voz por medio del API Speech Recognition que transforma la voz captada por el micrófono del computador en texto, que puede ser enviado como mensaje en el chat que posee la aplicación. La utilización de este API generó que la herramienta tenga dependencia de una conexión a internet para su funcionamiento, utilizando el mouse como disparador de esta funcionalidad.

La tercera funcionalidad que se dio es la posibilidad de que la aplicación en forma de audio indique las opciones de menú que dispone el usuario, esta funcionalidad también se utiliza en el repositorio que lee el contenido de la carpeta local y remota. Este también se utiliza en él envió de tareas al leer la pantalla indicándonos que se tiene pendiente, utilizando el teclado como disparador de esta funcionalidad.

Abstract

The prototype of AVIDA virtual environment was developed for collaboration with students with visual impairment in their learning process.

Prototype was built using Java, MySQL, Apache and Netbeans because it is not proprietary software and therefore there will be no problems with future licensing.

First usability that was delivered in the prototype is the use of the fingerprint as an alternative method for entering the application. U.are.u 4500 sensor was used due to its ease of installation, being a plug&play USB device that is compatible with the Windows operating system since the 2003 version.

Second usability of the application has is voice recognition through API Speech Recognition that transforms the voice captured by the computer microphone into text, which can be sent as a message in the chat that the application has. Use of this API resulted in the tool being dependent on an internet connection for its operation, using the mouse as a trigger for this usability.

Third usability that was given is the possibility that the application in the form of audio indicates the menu options available to the user, usability is also used in the repository that reads the contents of the local and remote folder. This one is also used in it sent tasks when reading the screen indicating us that it is pending, using the keyboard as trigger for this usability.

INTRODUCCIÓN

En la época actual en la que el conocimiento es un pilar fundamental para el desarrollo personal, es indispensable que las instituciones educativas cuenten con herramientas que permitan la integración de las personas con discapacidad visual al mundo educativo con un acceso a la web sin fronteras.

La estructura de este proyecto de titulación es la siguiente:

En el capítulo 1: Se explica la estructura del proyecto, justificación, planteamiento del problema, objetivos planteados, técnicas, métodos y herramientas que contribuyen para su desarrollo.

En el capítulo 2: Se realiza los procesos detallados en la metodología SCRUM además del marco teórico.

En el capítulo 3: Análisis, muestra los requerimientos previamente definidos.

En el capítulo 4: Diseño y construcción muestra la creación del código fuente de los diferentes módulos previamente definidos y sus diferentes diagramas.

En el capítulo 5: Pruebas, muestra el resultado de las pruebas realizadas a la aplicación.

La parte final se detalla las conclusiones y recomendaciones que surgen después del desarrollo del sistema.

CAPITULO 1

PLANTEAMIENTO DEL PROBLEMA

1.1. Antecedentes

En la actualidad en el Ecuador se han dado grandes pasos para la inclusión de las personas con discapacidad en la vida cotidiana por parte del Gobierno, con organismos como el CONADIS encargado de velar que en las políticas públicas sea considerado este grupo vulnerable.

Tener soluciones que ayuden en esto es bastante costoso, además de no existir una gran variedad para escoger en el entorno local. Esto ha impedido que en el ámbito particular se invierta demasiado en el tema aun con los esfuerzos realizados por el Ministerio de Inclusión Económica y Social.

1.2. Descripción del Problema

Los estudiantes con discapacidad visual tienen varios inconvenientes para acceder por sí solos a la información alojada en la web, por lo que ocasionalmente necesitan la ayuda de un tercero.

Este es nuestro punto de partida, el cual nos enfrenta a la enorme tarea de desarrollar un ambiente virtual para estudiantes con discapacidad visual, de tal modo que la interacción con la plataforma sea fácil, sencilla y aporte en el proceso de aprendizaje del estudiante.

1.3. Objetivos

1.3.1. Objetivo General

Analizar, diseñar y desarrollar un prototipo de ambiente virtual de aprendizaje accesible para estudiantes con discapacidad visual para la Universidad Politécnica Salesiana, Sede Quito.

1.3.2. Objetivos Específicos

Analizar los principales problemas de accesibilidad que enfrentan los estudiantes con discapacidad visual al AVAC con el que actualmente cuenta la UPS.

Diseñar y Construir un prototipo de ambiente virtual con metodología ágil como es el caso de SCRUM para estudiantes con discapacidad visual.

Generar un repositorio de objetos de aprendizaje con el fin de que los estudiantes puedan cargar o descargar material educativo referente a la asignatura que está cursando.

Adaptar un audio chat en el ambiente virtual para permitir la comunicación entre docente-estudiante y entre estudiantes.

1.4. Justificación

Según datos estadísticos obtenidos del Consejo Nacional para la Igualdad de Discapacidades (CONADIS) existe alrededor 54.529 personas registradas con discapacidad visual en el Ecuador (Consejo Nacional para la Igualdad de Discapacidades, 2019).

Actualmente en la Universidad Politécnica Salesiana sede Quito, tiene cuatro estudiantes con discapacidad visual de los cuales tres tienen ceguera total y una con ceguera nocturna.

La posibilidad de utilizar comandos del teclado, el reconocimiento de voz a los estudiantes con discapacidad visual ayuda a que puedan establecer cierto tipo de igualdad con las personas videntes.

Con el trabajo propuesto se busca el diseño de un prototipo de ambiente virtual que les sirva de apoyo a este grupo de estudiantes, con el fin de facilitar el acceso a todas las funcionalidades desarrolladas.

Con la dotación de estas funcionalidades se busca que el estudiante con discapacidad visual puede tener autonomía y no dependa de una tercera persona para realizar actividades propias.

1.5. Alcance

El proyecto tiene como propósito entregar un prototipo de ambiente virtual que sirva de apoyo a estudiantes con discapacidad visual, siendo esta la primera versión.

Según los requerimientos obtenidos el presente proyecto tiene como entregables los módulos de login, registros de usuarios, consulta de usuarios, chat grupal, chat personal, tareas y repositorio. A continuación brevemente se describirá cada uno de ellos:

- Módulo Login: Realizará el acceso al sistema mediante dos tipos de ingreso.
 - Ingreso mediante credenciales
 - Ingreso mediante lector de huella digital
- Módulo Registro y Consulta de Usuarios: Realizará la inserción y de nuevos usuarios al sistema y permitirá consultar todos los usuarios registrados.
- Módulo Chat Grupal y Personal: Permitirá el envío de mensajes por voz y/o mediante el método tradicional.
- Módulo Tareas: Este proceso comprende la asignación y envío de tareas.
- Módulo Repositorio: Este proceso comprende la carga y descarga de archivos.

1.5.1. Roles de Usuario

La aplicación tendrá tres roles y se detalla las acciones que cada uno podrá utilizar:

- Administrador: Controla los módulos de acceso al sistema permitiéndole insertar y consultar usuarios.
- Docente: Tiene acceso a los módulos de tareas, chat grupal y personal.
- Estudiante: Tiene acceso a los módulos de tareas, chat grupal, chat personal y repositorio.

1.6. Fuera de alcance o suposiciones

En esta sección se detalla los puntos de los entregables que quedan fuera de alcance:

- No se considera la implementación del sistema.
- No se considera las necesidades fuera de este documento.
- No se considera la implementación tanto del hardware y software de este proyecto.

1.7. Marco metodológico

En esta sección del documento se pretende explicar a breves rasgos la metodología SCRUM empleada en el proyecto.

Esta fue seleccionada debido a sus beneficios de agilidad, flexibilidad y adaptación garantizando el cumplimiento de los objetivos y empleando las buenas prácticas en el desarrollo.

1.7.1. Scrum

Scrum es una metodología para equipos de trabajo eficientes, que busca lograr el óptimo resultado del proyecto planteado.

Fomentando la comunicación entre el equipo y el cliente para entender sus necesidades, además de priorizar las diferentes tareas que forman parte del proyecto (Luis Goncalves, 2019).

1.7.1.1. Equipo Scrum

Formado básicamente por: el Product Owner, Development team y Scrum Master (Luis Goncalves, 2019).

- Product Owner: Es el encargado de tomar decisiones y hacer cumplir los objetivos del proyecto.

Nombre: Estefani Herrera

- Scrum Master: Es el encargado de liderar al equipo tanto técnico y funcional.

Nombre: Gustavo Navas

- Development team: Son los encargados de manera conjunta de desarrollar el producto del proyecto.

Nombres: Estefani Herrera y Ricardo Figueroa

1.8. Cronograma referencial de desarrollo

En la Figura 1 y 2 se describe el cronograma de desarrollo que se utilizará para realizar la aplicación.

Cronograma sprint parciales						
Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos	
AMBIENTE VIRTUAL	339 días?	mar 13/03/18	vie 28/06/19			
▾ Análisis	24 días	mar 13/03/18	vie 13/04/18		Estefani Herrera;Ricardo Figueroa	
Análisis y diseño	9 días	mar 13/03/18	vie 23/03/18		Estefani Herrera;Ricardo Figueroa	
Generación de prototipo	13 días	lun 26/03/18	mié 11/04/18	3	Estefani Herrera;Ricardo Figueroa	
Revisión de prototipo	2 días	jue 12/04/18	vie 13/04/18	4	Estefani Herrera;Ricardo Figueroa	
▾ Construcción	315 días?	lun 16/04/18	vie 28/06/19		Estefani Herrera;Ricardo Figueroa	
Preparación de ambientes de desarrollo	3 días	lun 16/04/18	mié 18/04/18	5	Estefani Herrera;Ricardo Figueroa	
Desarrollo Login	15 días	mar 01/05/18	lun 21/05/18	7	Estefani Herrera;Ricardo Figueroa	
Desarrollo login e integración con huella digital	15 días	mar 22/05/18	lun 11/06/18	8	Estefani Herrera;Ricardo Figueroa	
Desarrollo registro y consulta de usuarios e integración con huella digital	13 días	vie 13/07/18	mar 31/07/18	9	Estefani Herrera;Ricardo Figueroa	
Registro de usuarios e Integración con huella digital	14 días	jue 02/08/18	mar 21/08/18	10	Estefani Herrera;Ricardo Figueroa	
Desarrollo chat grupal y personal	15 días	mié 22/08/18	mar 11/09/18	11	Estefani Herrera;Ricardo Figueroa	
Desarrollo integración api voz con chat grupal y personal	15 días	mié 12/09/18	mar 02/10/18	12	Estefani Herrera;Ricardo Figueroa	
Desarrollo manejador de archivos	13 días	lun 05/11/18	mié 21/11/18	13	Estefani Herrera;Ricardo Figueroa	
Desarrollo intergración api voz con manejador de archivos	15 días	vie 04/01/19	jue 24/01/19	14	Estefani Herrera;Ricardo Figueroa	
Desarrollo Tareas	15 días	vie 25/01/19	jue 14/02/19	15	Estefani Herrera;Ricardo Figueroa	
Desarrollo integración api de voz con tareas	14 días	vie 15/02/19	mié 06/03/19	16	Estefani Herrera;Ricardo Figueroa	
Desarrollo de correcciones	16 días	jue 07/03/19	jue 28/03/19	17	Estefani Herrera;Ricardo Figueroa	
Documento Final	66 días	vie 29/03/19	vie 28/06/19	18	Estefani Herrera;Ricardo Figueroa	

Figura 1. Cronograma de desarrollo.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

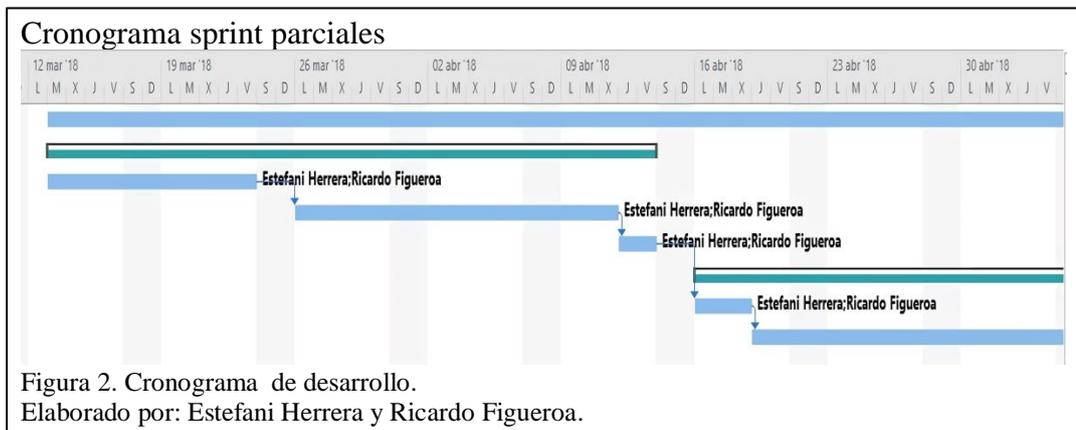


Figura 2. Cronograma de desarrollo.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

1.9. Fases de la metodología Scrum

1.9.1. Reunión de planificación de Sprint

En ella el equipo determina cual es la funcionalidad y el entregable, así como la forma en la que se logrará alcanzar este objetivo (Universitat de Barcelona, 2019).

1.9.2. El Scrum Diario

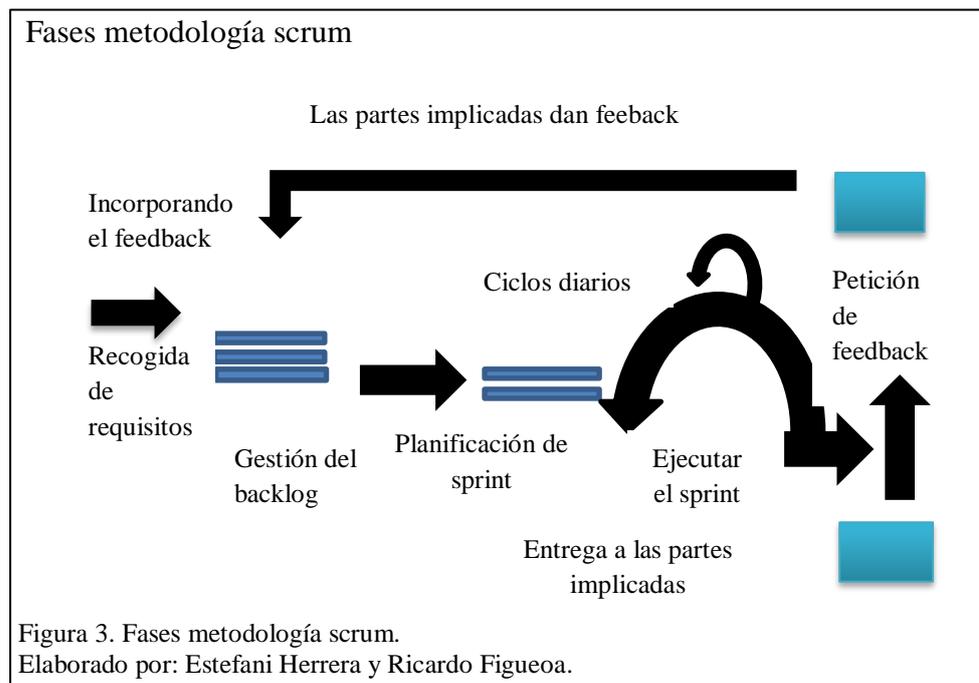
Forma de validar el progreso del proyecto de forma diaria y de ser necesario realizar el ajuste requerido (Universitat de Barcelona, 2019).

1.9.3. Trabajo de desarrollo durante el Sprint

Los sprints son entregables que permiten validar que se esté avanzando al objetivo planificado, estas salidas se entregarán por lo menos una vez al mes (Universitat de Barcelona, 2019).

- Revisión del Sprint: Reunión de feedback para revisar el Sprint dentro del equipo de trabajo (Universitat de Barcelona, 2019).
- Retrospectiva del Sprint: Consiste en aplicar la mejora continua para el siguiente sprint, consolidando el equipo de trabajo (Universitat de Barcelona, 2019).

En la Figura 3 muestra las fases de la metodología Scrum.



CAPITULO 2

MARCO TEÓRICO

2.1. Accesibilidad Universal

Beneficia a cualquier persona incluyendo a aquellas que tengan alguna discapacidad cognitiva, física, visual, etc.

Considerando que discapacidad es una deficiencia o limitación para que la persona realice sus actividades (Universidad de Alicante, 2019).

2.2. Herramientas y tecnología

Las herramientas y tecnologías seleccionadas para la creación del sistema se describen a continuación:

2.3. Gestores de programación

Las herramientas que se utilizó para el desarrollo de la aplicación son:

- Java: Lenguaje de programación para desarrollar aplicaciones útiles y de fácil uso en sistemas operativos por ejemplo Windows, Linux, Apple (Java, s.f).
- NetBeans: Es una herramienta de desarrollo el cual nos permite codificar en varios lenguajes de programación (NetBeans, s.f).
- JavaScript: Es un lenguaje dinámico el cual está orientado a eventos para el manejo de datos convirtiéndolo en un sistema multiplataforma, permitiendo de esta manera ser un sistema cliente servidor (MDN web docs, 2019).
- JDK (Java Development Kit): Es un conjunto de utilidades, herramientas, programas y librerías que desarrollan, compilan, ejecutan y generan documentación en el lenguaje Java, es decir permite la traducción del código de programación a un lenguaje de usuario (Victor Ramos, s.f).

- JSP (Java Server Pages): Permite crear aplicativos que se ejecutan de forma distribuida en múltiples servidores. En el sistema permite la interconexión de códigos de programación en un sistema multicapa (IBM Knoeledge Center, s.f).
- JQuery: Librería de JavaScript simplifica la tarea del desarrollo en JavaScript que permite crear sitios web interactivos (Laura Chuburu, 2018).
- Start Bootstrap: Crea de forma gratuita, código abierto, licencia MIT, temas, plantillas y fragmentos de código de Bootstrap para su uso en cualquier proyecto (David Miller, s.f).
- Apache: Servidor web que está diseñado para enviar datos de hipertexto, es decir, páginas web estáticas y dinámicas (Culturacion, s.f).
- OpenSSH: Es la principal herramienta de conectividad para el inicio de sesión remoto con el protocolo SSH. Cifra todo el tráfico para eliminar las escuchas ilegales. La suite OpenSSH consta de las siguientes herramientas:
 - Las operaciones remotas se realizan utilizando ssh , scp y sftp.
 - Administración de claves con ssh-add, ssh-keysign, ssh-keyscan y ssh-keygen.
 - El lado del servicio consiste en sshd, sftp-server y shh-agent (OpenSSH, 2019).

2.4. Gestores de base de datos

Las herramientas que se utilizó para acceder, actualizar, insertar mediante procedimientos almacenados son:

- SQL: Lenguaje interactivo que permite el acceso a base de datos relacionales, para la interacción usuario-sistema mediante sentencias definidas (TechTarget, 2019).

- SQLyog Community: Es el cliente de MySQL Workbench integral GUI para el RDBMS MySQL (MariaDB, 2019).
- MySQL: Gestor de base de datos con código abierto, basado en lenguaje SQL (Margaret Rouse, 2015).
- MySQL Workbench: Herramienta para el diseño de base de datos de MySQL (UBUNLOG, s.f).

2.5. Api reconocimiento de voz (Speech Recognition)

Web Speech API facilita la adición de reconocimiento de voz a sus páginas web.

Esta API permite un control preciso y flexibilidad sobre las capacidades de reconocimiento de voz en Chrome versión 25 y posteriores.

Chrome ha implementado la especificación del W3C, utilizando motores de reconocimiento de voz de Google (Kai Wedekind, 2018).

2.6. Sensor Biométrico Digital Persona U.are.u 4500

Dispositivo portátil USB plug&play para la identificación personal por medio de huella digital (SIASA, 2011).

La utilización consiste en colocar el dedo sobre el dispositivo, mismo que mediante tecnología óptica escaneara de forma rápida la huella digital sin importar el ángulo de colocación (SIASA, 2011).

En la Tabla 1 detalla las características principales del sensor de huella.

Tabla 1. Características sensor de huella

Digital Persona U.are.u 4500	
Imagen	Características Principales
	Rechazo de impresión latente
	Diseño compacto
	Excelente calidad de imagen
	Datos encriptados
	Rechazo de huella falsa
	Funciona con huella secas, húmedas o ásperas
	Compatible con Windows desde la versión 2003 a la actual.

Figura 4. Digital Persona
Elaborado por: Estefani Herrera y Ricardo Figueroa.

Nota: Características principales del sensor de huella.

Elaborado por: Estefani Herrera y Ricardo Figueroa, envasé a las especificaciones del sensor de huella (SIASA, 2011).

CAPÍTULO 3

ANÁLISIS

La necesidad de la aplicación empezó al momento de realizar la toma de requerimientos de los estudiantes con discapacidad visual de la Universidad Politécnica Salesiana involucrados determinando los aspectos globales para realizar una aplicación completa, lógica y de fácil aprendizaje hacia el usuario.

3.1. Análisis de Requerimientos

En el proyecto se cuenta con dos tipos de requerimientos: Funcionales y No Funcionales.

3.1.1. Requerimiento Funcionales

Consisten en obtener los requisitos para el desarrollo de la aplicación, considerando los diferentes tipos de usuarios. Para ellos se tiene las diferentes funcionalidades:

- Roles de la aplicación:

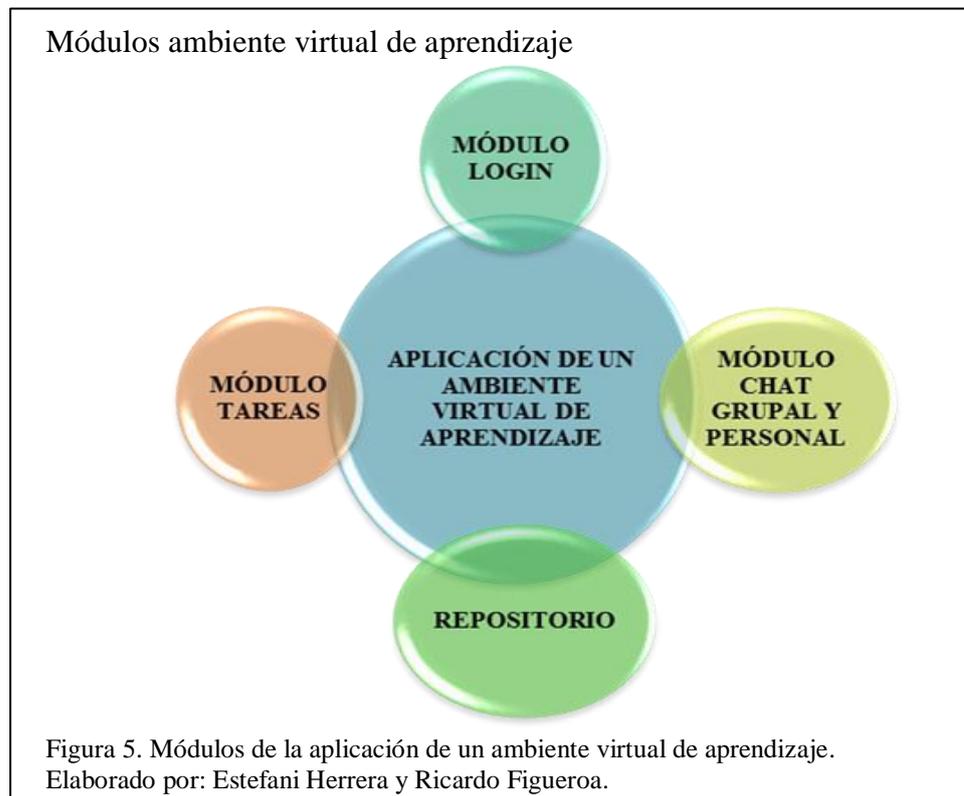
La aplicación cuenta con tres actores:

- Administrador: Usuario con todos los permisos para el manejo total del sistema. Todo su trabajo podrá ser realizado a través de la aplicación, previa autenticación.
- Docente: Representa la persona que se encargará de realizar el registro de notas y la utilización del módulo audio chat en línea.

- Estudiante: Representa la persona que va ingresar y debe poner su huella sobre el lector para que la aplicación proceda con la autenticación, posee el manejo de los siguientes módulos que son repositorio donde puede cargar y descargar archivos, audio chat en línea, envió de tareas y revisión de notas.

3.1.1.1. Módulos de la aplicación

Para desarrollar esta aplicación con características que permitan la flexibilidad para un futuro escalamiento de requerimientos se lo ha dividido en los siguientes módulos que se muestran en la figura 5:



3.1.1.2. Diagramas de caso de uso

Se presenta todas las interfaces y funcionalidades de la aplicación tanto en el diagrama de caso de uso, secuencia y actividad de cada evento a desarrollar.

3.1.1.2.1. Módulo funcionalidad general de la aplicación

En la Figura 6 se observa el diagrama de caso de uso del ingreso a la aplicación por medio de sus credenciales.

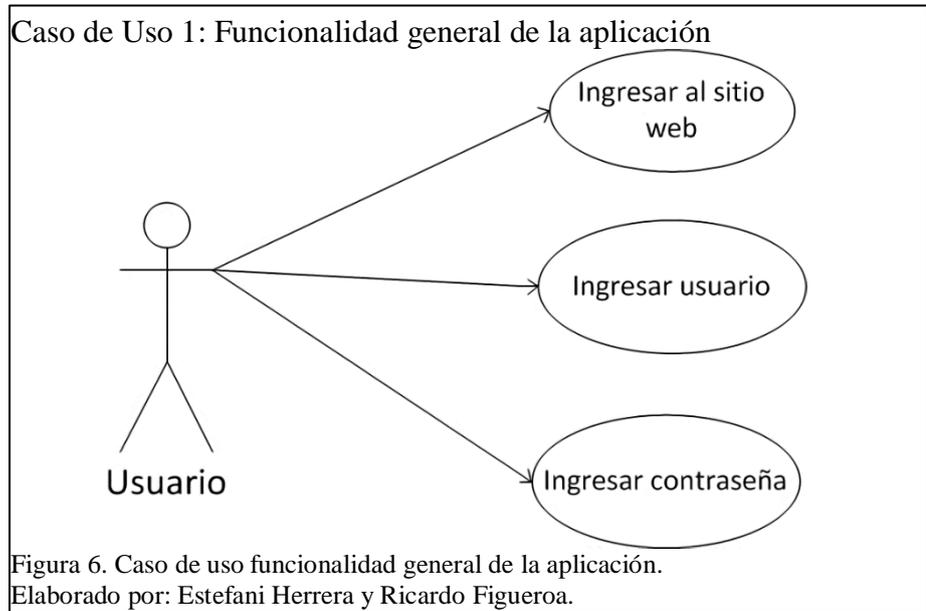


Tabla 2. Caso de uso - Funcionalidad general de la aplicación

Caso de uso # 1 Funcionalidad general de la aplicación	
Actores	Usuario
Camino Principal	1. Ingresar al sitio web 2. Ingresar usuario 3. Ingresar contraseña
Precondiciones	Estar conectado siempre a internet ya que la aplicación hace uso del api de Google para su reconocimiento de voz. Instalación de sensor biométrico (Huella Digital). Conexión a la base de datos. Poseer credenciales.
Post-condiciones	Cargar perfiles. Almacenamiento en base de datos.

Nota: Descripción del usuario al usar la aplicación.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 7 se observa el diagrama de caso de uso de la funcionalidad de la aplicación para el administrador y el estudiante, carga el siguiente menú de opciones, módulo chat grupal o sala personal, módulo tareas y módulo repositorio, selecciona el estudiante a que opción desea ingresar.

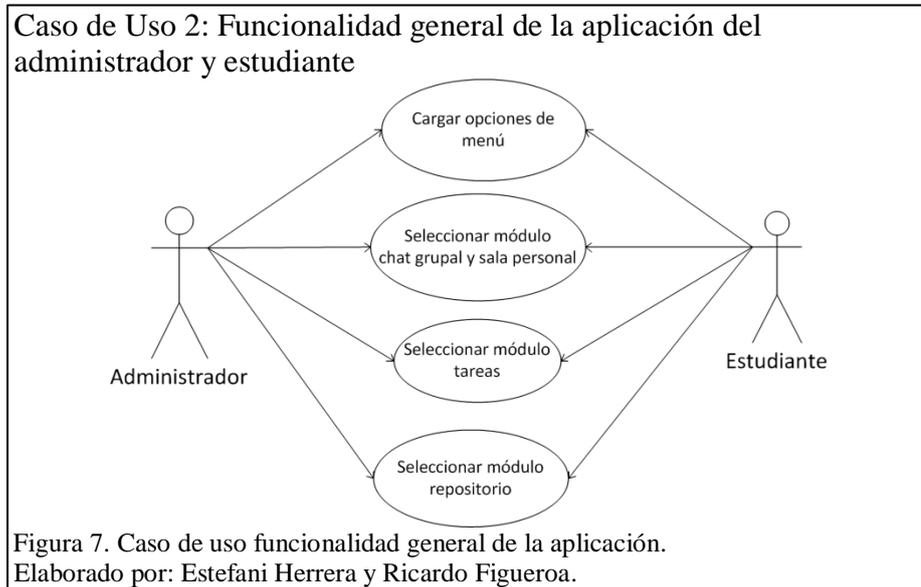


Tabla 3. Caso de uso - Funcionalidad general de la aplicación administrador-estudiante

Caso de uso # 2 Funcionalidad general de la aplicación	
Actores	Administrador/Estudiante
Camino Principal	1. Cargar opciones de menú. 2. Seleccionar módulo chat y sala personal. 3. Seleccionar módulo tareas. 4. Seleccionar módulo repositorio.
Precondiciones	Estar conectado siempre a internet ya que la aplicación hace uso del api de Google para su reconocimiento de voz. Instalación de sensor biométrico (Huella Digital). Conexión a la base de datos. Poseer credenciales.
Post-condiciones	Cargar perfiles. Almacenamiento en base de datos.

Nota: Descripción del docente y estudiante al usar la aplicación.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 8 se observa el diagrama de caso de uso de la funcionalidad de la aplicación para el docente, carga las siguientes opciones de menú, módulo chat grupal o sala personal, módulo tareas, selecciona el docente a que opción desea ingresar.

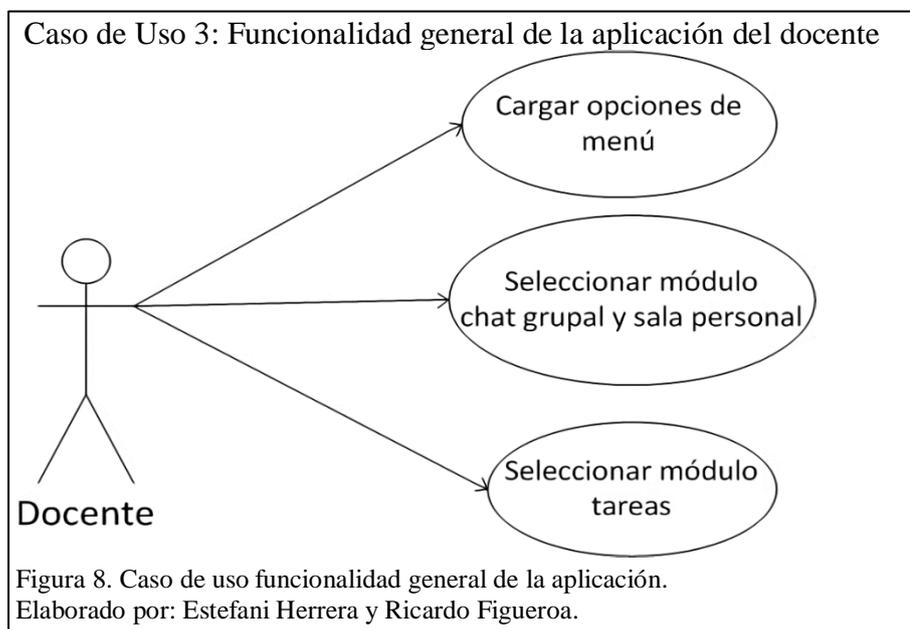


Tabla 4. Caso de uso - Funcionalidad general de la aplicación docente

Caso de uso # 3 Funcionalidad general de la aplicación	
Actores	Docente
Camino Principal	<ol style="list-style-type: none"> 1. Cargar opciones de menú. 2. Seleccionar módulo chat y sala personal. 3. Seleccionar módulo tareas.
Precondiciones	<ul style="list-style-type: none"> • Estar conectado siempre a internet ya que la aplicación hace uso del api de Google para su reconocimiento de voz. • Instalación de sensor biométrico (Huella Digital). • Conexión a la base de datos. • Poseer credenciales.
Post-condiciones	<ul style="list-style-type: none"> • Cargar perfiles. • Almacenamiento en base de datos.

Nota: Descripción del docente al usar la aplicación.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

3.1.1.3. Módulo chat grupal y sala personal

En la Figura 9 se observa el diagrama de caso de uso, el que se detalla cómo funciona el módulo, carga las opciones de menú, muestra dos opciones chat grupal y sala personal, si el usuario escoge la opción chat grupal puede establecer comunicación por audio o texto con todos los usuarios que estén conectados, si su opción es sala personal establece comunicación por audio o texto escogiendo los usuarios que desee.

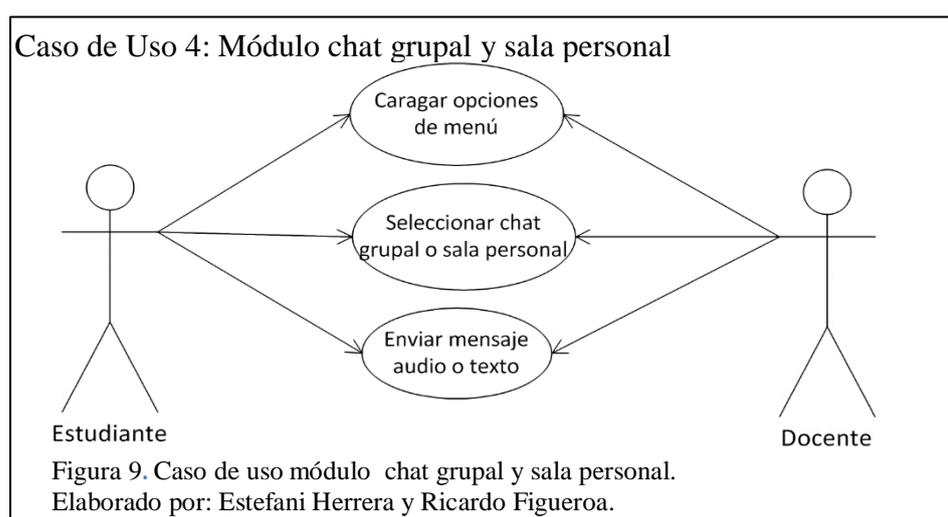


Tabla 5. Caso de uso - Módulo chat grupal y sala personal

Caso de uso # 4 Módulo chat grupal y sala personal	
Actores	Estudiante/Docente
Camino Principal	<ol style="list-style-type: none"> 1. Cargar opciones de menú. 2. Seleccionar módulo chat grupal o sala personal. 3. Enviar mensaje de audio o texto.
Precondiciones	Estar conectado siempre a internet ya que la aplicación hace uso del api de Google para su reconocimiento de voz. Instalación de sensor biométrico (Huella Digital). Conexión a la base de datos. Poseer credenciales.
Post-condiciones	Cargar perfiles. Almacenamiento en base de datos.

Nota: Descripción del docente y estudiante al usar el módulo chat grupal y personal.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

3.1.1.4. Módulo Envió de tareas

En la Figura 10 se observa el diagrama de caso de uso, en el que se detalla cómo funciona el módulo, carga opciones de menú, el estudiante ingresa al módulo tareas, selecciona materia, carga la tarea, envía tarea y consulta notas.

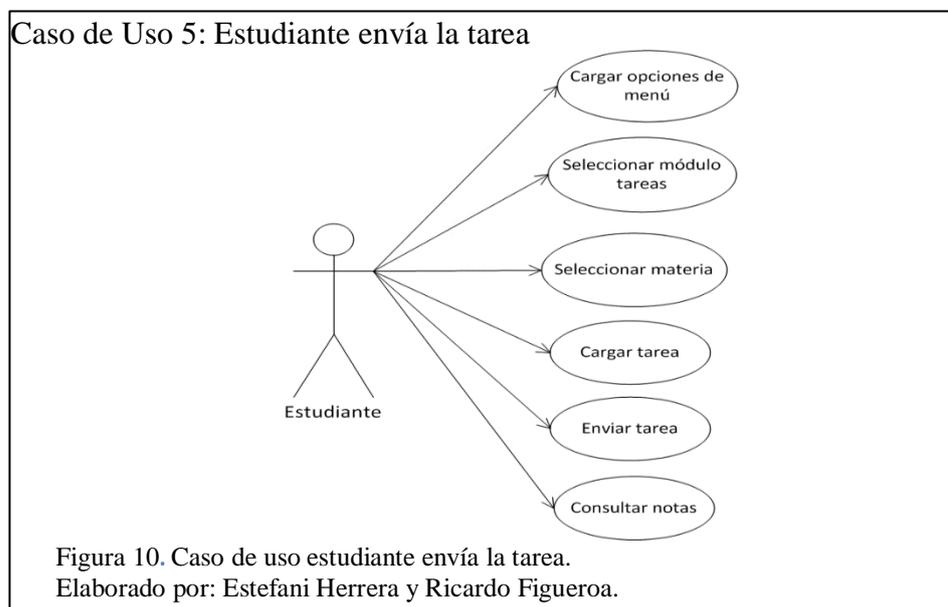


Tabla 6. Caso de uso - Módulo tareas estudiante

Caso de uso # 5 Módulo tareas	
Actores	Estudiante
Camino Principal	<ol style="list-style-type: none"> 1. Cargar opciones de menú. 2. Seleccionar módulo tareas. 3. Seleccionar materia. 4. Cargar tarea. 5. Enviar tarea. 6. Consultar las notas.
Precondiciones	Estar conectado siempre a internet ya que la aplicación hace uso del api de Google para su reconocimiento de voz. Instalación de sensor biométrico (Huella Digital). Conexión a la base de datos. Poseer credenciales.
Post-condiciones	Cargar perfiles. Almacenamiento en base de datos.

Nota: Descripción del módulo tareas por el estudiante.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 11 se observa el diagrama de caso de uso, en el que se detalla cómo funciona el módulo, carga opciones de menú, el docente, ingresa al módulo tareas, selecciona materia y califica la tarea.

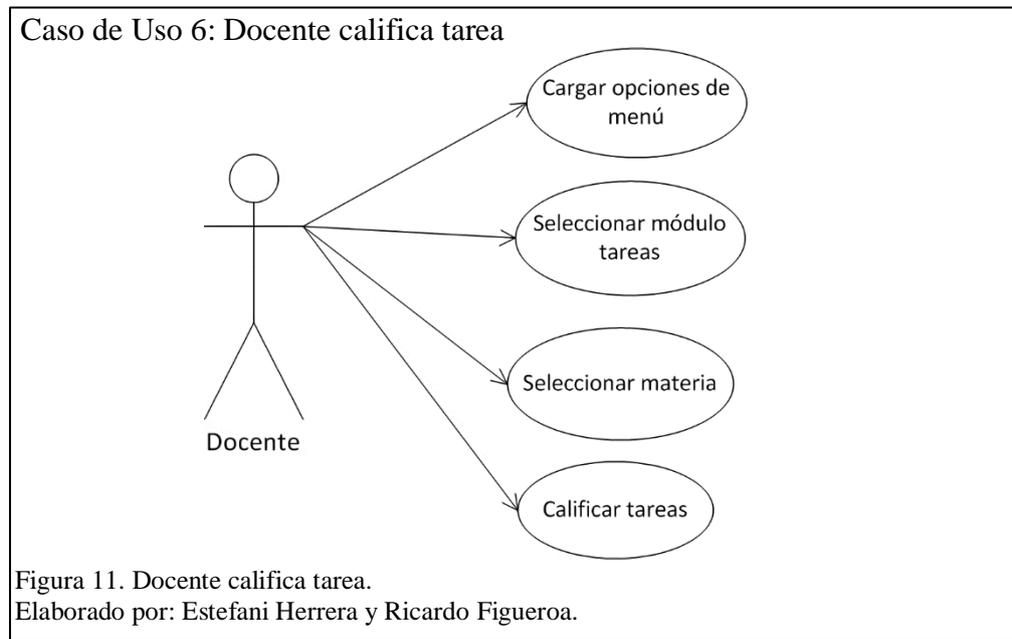


Tabla 7. Caso de uso - Módulo tareas

Caso de uso # 6 Módulo tareas	
Actores	Docente
Camino Principal	<ol style="list-style-type: none"> 1. Cargar opciones de menú. 2. Seleccionar módulo tareas. 3. Calificar tareas.
Precondiciones	Estar conectado siempre a internet ya que la aplicación hace uso del api de Google para su reconocimiento de voz. Instalación de sensor biométrico (Huella Digital). Conexión a la base de datos. Poseer credenciales.
Post-condiciones	Cargar perfiles. Almacenamiento en base de datos.

Nota: Descripción del docente al usar el módulo tareas.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

3.1.1.5. Módulo Repositorio

En la Figura 12 se observa el diagrama de caso de uso, en el que se detalla cómo funciona el módulo, carga el menú de opciones, ingresa al menú donde se visualiza la opción módulo repositorio, se muestran dos opciones upload y download, si se escoge upload carga un archivo desde la carpeta local llamada upload en el disco D y lo almacena en el servidor en la carpeta repositorio en el disco D.

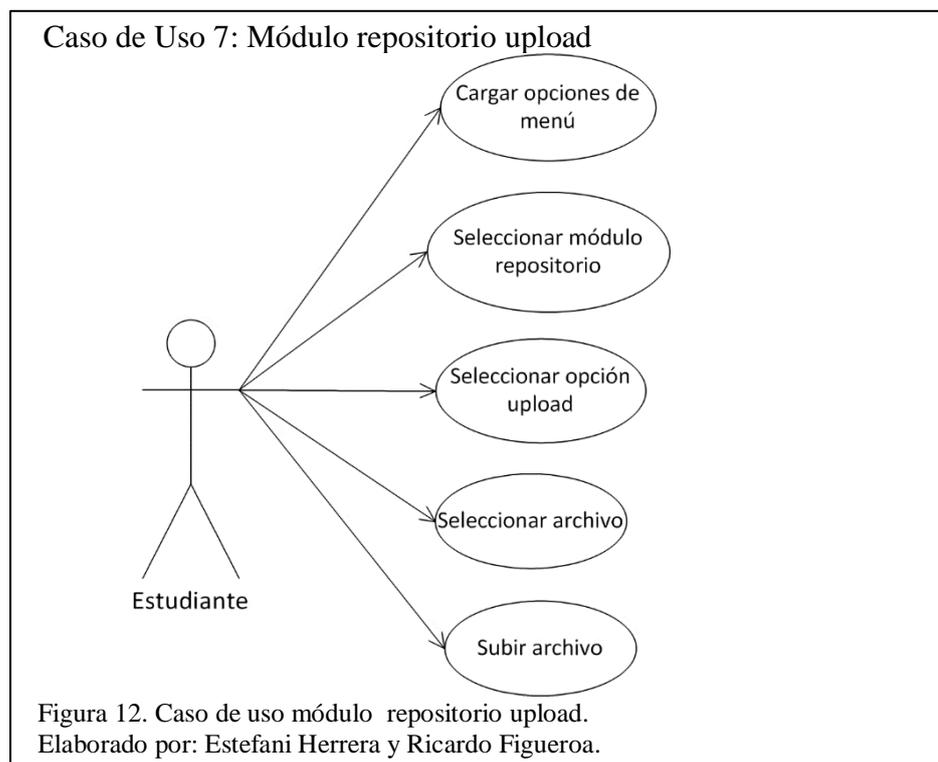


Tabla 8. Caso de uso - Módulo repositorio upload

Caso de uso # 7 Módulo repositorio upload	
Actores	Estudiante
Camino Principal	<ol style="list-style-type: none"> 1. Cargar opciones de menú. 2. Seleccionar módulo repositorio. 3. Seleccionar opción upload. 4. Seleccionar archivo. 5. Cargar archivo.
Precondiciones	<p>Estar conectado siempre a internet ya que la aplicación hace uso del api de Google para su reconocimiento de voz.</p> <p>Servicio OpenSSH disponible. Su instalación se describe en el Anexo 2.</p> <p>Instalación de sensor biométrico (Huella Digital).</p> <p>Conexión a la base de datos.</p> <p>Poseer credenciales.</p> <p>Crear una carpeta upload disco D cliente.</p> <p>Crear una carpeta repositorio disco D en el servidor.</p>
Post-condiciones	<p>Cargar perfiles.</p> <p>Almacenamiento en base de datos.</p>

Nota: Descripción del estudiante al usar el módulo repositorio upload.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 13 se observa el diagrama de caso de uso, en el que se detalla cómo funciona el módulo, carga el menú de opciones, ingresa al menú donde muestra la opción módulo repositorio, se va a visualizar dos opciones upload y download, si se escoge download, lista los archivos en la carpeta repositorio del disco D en el servidor y los descarga al cliente.

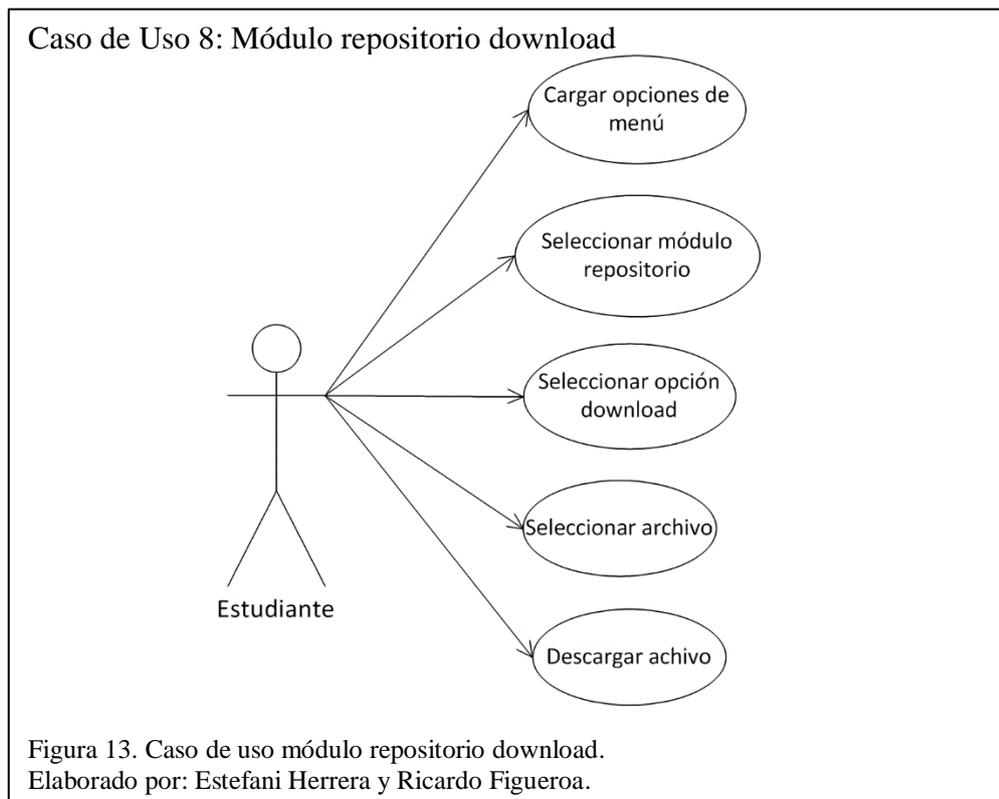


Tabla 9. Caso de uso - Módulo repositorio download

Caso de uso # 8 Módulo repositorio download	
Actores	Estudiante
Camino Principal	<ol style="list-style-type: none"> 1. Cargar opciones de menú. 2. Seleccionar módulo repositorio. 3. Seleccionar opción download. 4. Seleccionar archivo. 5. Descargar archivo.
Precondiciones	<p>Estar conectado siempre a internet ya que la aplicación hace uso del api de Google para su reconocimiento de voz.</p> <p>Servicio OpenSSH disponible. Su instalación se describe en al Anexo 2.</p> <p>Instalación de sensor biométrico (Huella Digital).</p> <p>Conexión a la base de datos.</p> <p>Poseer credenciales.</p> <p>Crear una carpeta repositorio disco D en el servidor.</p>
Post-condiciones	<p>Cargar perfiles.</p> <p>Almacenamiento en base de datos.</p>

Nota: Descripción del estudiante al usar el módulo repositorio download.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

3.1.2. Requerimientos no funcionales

La aplicación trabaja en ambiente web y hace usos de las siguientes aplicaciones:

- Windows: Sistema Operativo en el que se instala los programas de desarrollo, mínimo versión 7.
- Apache: Versión 9.0, utilizado para el envío y publicación de las páginas web.
- MySQL: Versión 4.8.5, utilizado para almacenar la información que será consumida por el aplicativo.
- NetBeans IDE: Versión 8.2, utilizado para facilitar el desarrollo de las interfaces de usuario.
- Bootstrap Grid: Versión 4.1.3, utilizado para facilitar la creación de interfaces agradables al usuario.
- JQuery: Versión 3.4.1, permite manipular elementos como textos, imágenes, enlaces y cambiar el diseño CSS.

3.2. Costo Estimado

Tabla 10. Costo estimado

N	Descripción	Cantidad	Valor Unitario (dólares)	Valor Total (dólares)
1	Investigador	1440 horas/persona	3	8640
2	Material de escritorio	1	100	100
3	Internet	18 meses	25	450
4	Lector de huella digital	1	108	108
5	Audífonos	1	10	10
6	Transporte	144 veces	1.60	230.4
7	Luz	18 meses	10	180
TOTAL				9.718.40

Nota: Descripción costo estimado.

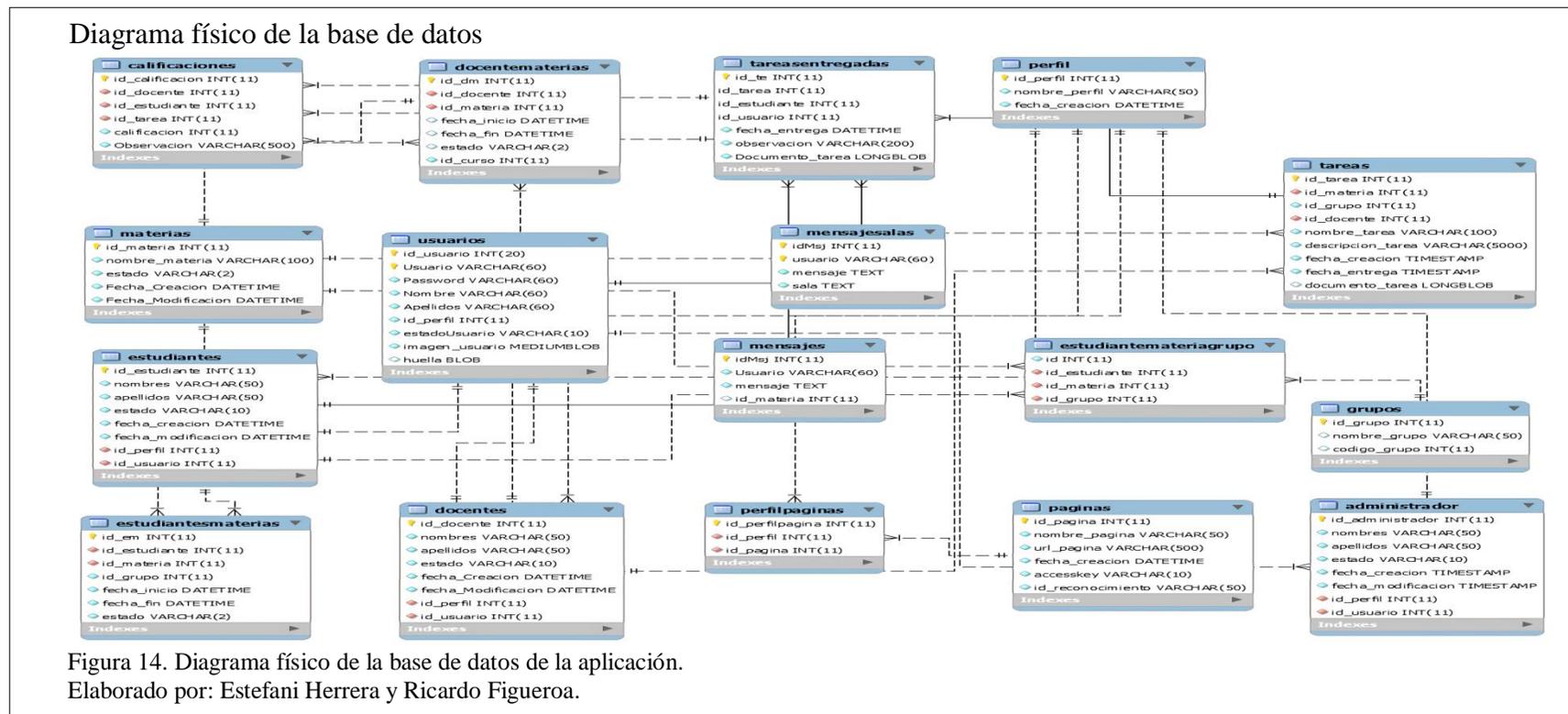
Elaborado por: Estefani Herrera y Ricardo Figueroa.

CAPITULO 4

DISEÑO Y CONSTRUCCIÓN

Durante este capítulo, se muestra el diseño y la codificación de las diferentes clases que permiten construir los módulos previamente diseñados.

4.1. Modelo Físico de la Base de Datos



En la Figura 14 se observa el diagrama físico de base de datos donde se visualiza las tablas que conforman los diferentes módulos.

Consta de 17 tablas, las cuales están relacionadas de manera lógica, garantizando la integridad y consistencia de los datos.

4.2. Modelo Conceptual de la Base de Datos

En la Figura 15 se visualiza el diagrama conceptual del proyecto en donde se detalla entidades, propiedades y relaciones de la aplicación. Esto con la finalidad de verificar que no exista redundancia y duplicidad de información.

4.2.1. Diccionario de Datos

En esta sección indica las tablas más relevantes que conforman la base de datos de la aplicación. Todas las tablas que la integran se encuentran en el Anexo 1.

La Tabla 11 se muestra la relación entre las materias que se encuentran cursando el estudiante.

Tabla 11. Estudiantes materias

Columna	Tipo de dato	Campo nulo	Relación a	Comentarios
id_estudiante (Primaria)	int(11)	No		Identificador Estudiante
nombres		No		Nombres completos del estudiante
apellidos	varchar(50)	No		Apellidos completos del estudiante
estado	varchar(10)	No		Estado del estudiante ACTIVO - INACTIVO
id_perfil	int(11)	No	Perfil -> id_perfil	Identificador del perfil del estudiante, de la tabla perfil

Nota: Detalles de los atributos y breve descripción.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

La Tabla 12 muestra el listado de todas las materias que pueden ser cursadas por un estudiante.

Tabla 12. Materias

Columna	Tipo de dato	Campo nulo	Comentarios
id_materia (Primaria)	int(11)	No	Identificador de la materia
nombre_materia	varchar(100)	No	Materia
estado	varchar(2)	No	Estado de la materia, ACTIVO – INACTIVO
fecha_creacion	datetime	No	Fecha de creación de la materia
fecha_modificacion	datetime	No	Fecha de modificación de la materia

Nota: Detalles de los atributos y breve descripción.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

La Tabla 13 muestra en donde se guarda los mensajes que hayan sido enviados dentro de la sala personal.

Tabla 13. Mensaje salas

Columna	Tipo de dato	Campo nulo	Comentarios
idMsj (Primaria)	int(11)	No	Identificador del mensaje de la sala
usuario (Primaria)	varchar(60)	No	Código del usuario, que envía – recibe mensaje
mensaje	text	No	Detalle mensaje
sala	txt	No	Sala de chat

Nota: Detalles de los atributos y breve descripción.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

La Tabla 14 muestra en donde se guarda la tarea por materia que debe realizar el estudiante.

Tabla 14. Tareas

Columna	Tipo de dato	Campo nulo	Relación	Comentarios
id_tarea (Primaria)	int(11)	No		Identificador de la tarea
id_materia	int(11)	No	materias - > id_materia	Identificador de la materia, de la tabla materias
id_grupo	int(11)	No		Identificador del grupo de la tabla grupos
id_docente	int(11)	No	docente s- > id_docente	Identificador del docente, de la tabla docentes
fecha_inicio	datetime	No		Fecha de inicio de la tarea
fecha_fin	datetime	No		Fecha del fin de la tarea
documento_tarea	blob	No		Documento de indicaciones para realizar la tarea

Nota: Detalles de los atributos y breve descripción.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

La Tabla 15 muestra la información de los usuarios incluyendo su registro biométrico en forma de template de la huella digital almacena en un campo de tipo blob.

Tabla 15. Usuarios

Columna	Tipo de dato	Campo nulo	Comentarios
id_usuario (Primaria)	int(20)	No	Identificador único de usuario
usuario (Primaria)	varchar(60)	No	Nombre identificador de usuario
Password	varchar(60)	No	Clave de usuario
Nombre	varchar(60)	No	Nombre completo del usuario
Apellidos	varchar(60)	No	Apellido completo del usuario
id_perfil	int(11)	No	Identificador del perfil del usuario
imagen_usuario	mediumblob	No	Fotografía del usuario
huella	blob	No	Huella del usuario

Nota: Detalles de los atributos y breve descripción.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

4.3. Diagrama de clases

Aquí podemos observar el esquema de clases donde se detalla los atributos y métodos que corresponden a cada una de las clases que se utiliza para la solución de la aplicación como se muestra en la figura 16.

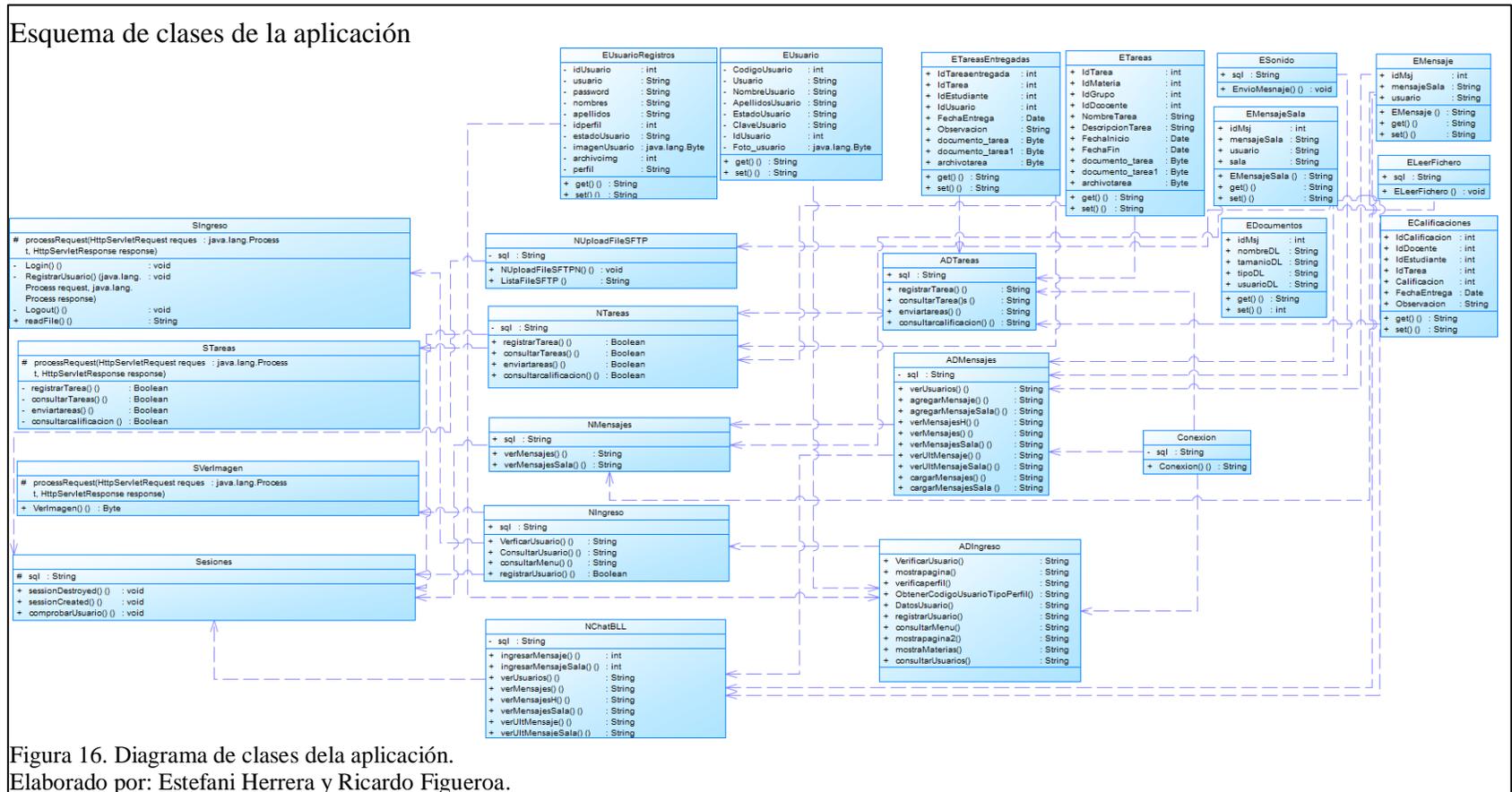


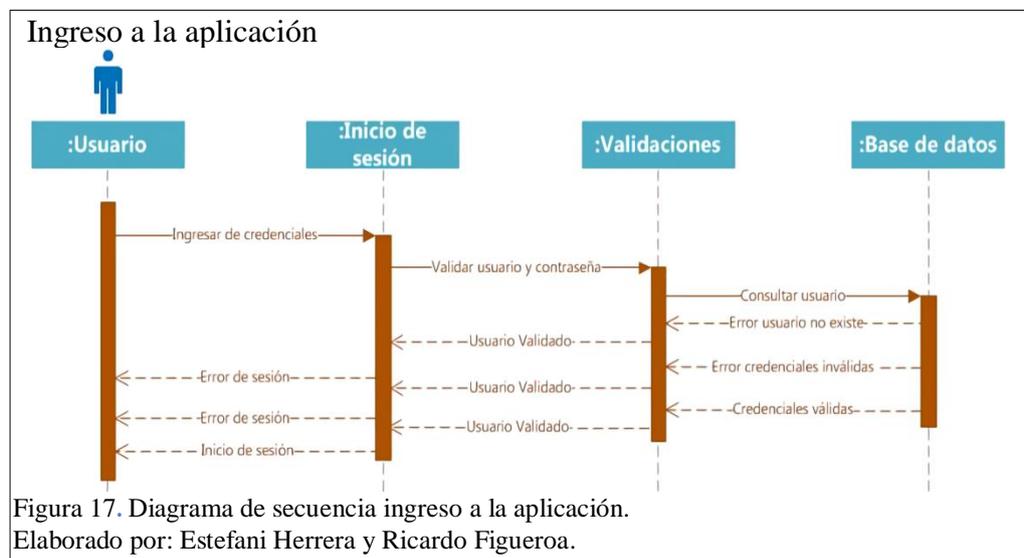
Figura 16. Diagrama de clases de la aplicación.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

4.4. Diagramas de secuencia

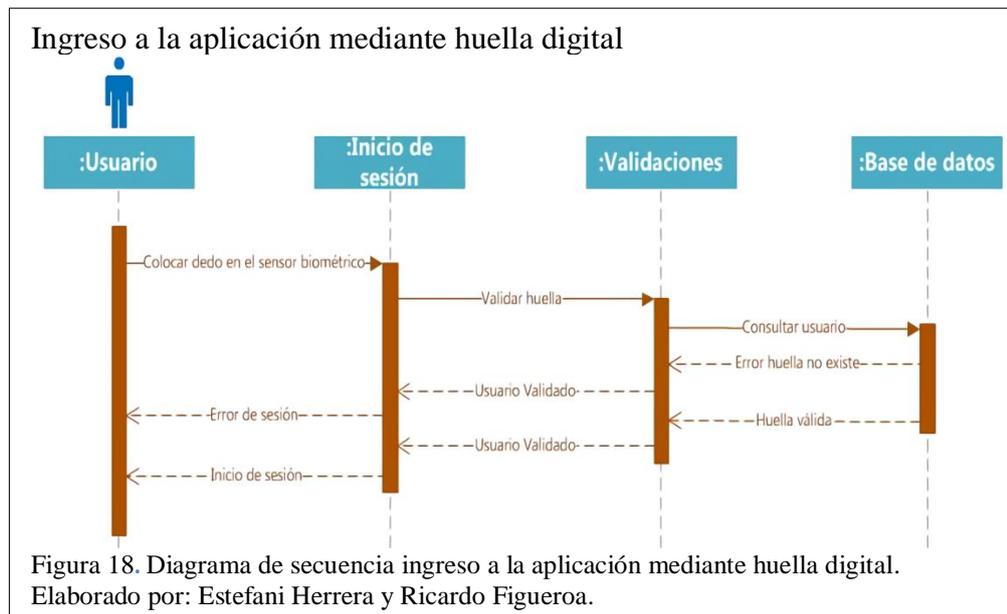
Este ítem muestra los esquemas de secuencia de los módulos que conforman la aplicación:

- Registro y Consulta de usuarios
- Modulo chat grupal y personal
- Módulo de tareas
- Módulo repositorio

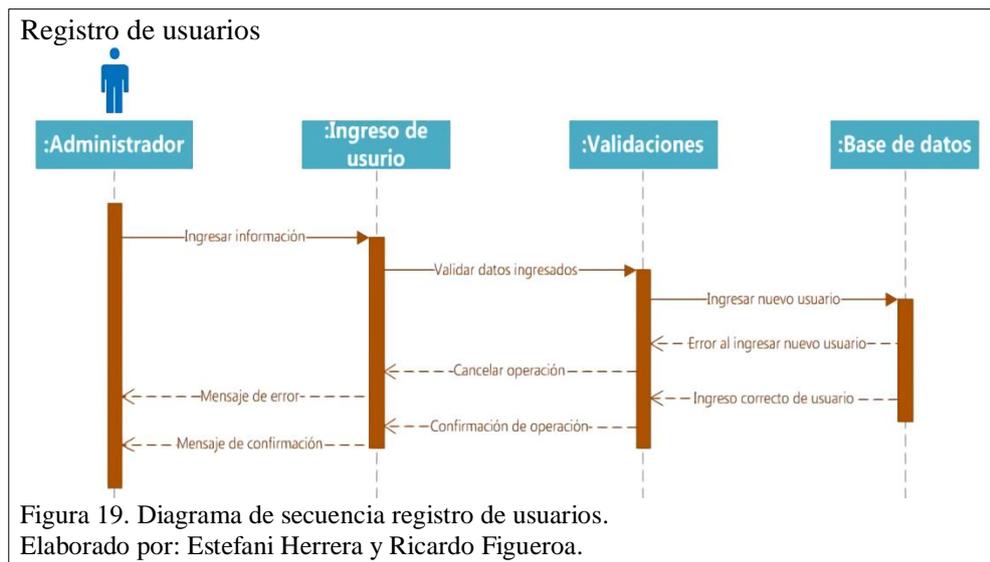
La Figura 17 el diagrama de secuencia que detalla las acciones realizadas por el actor en este caso el usuario, ingresa credenciales, validación de las mismas, consulta el usuario en la base de datos e inicia sesión.



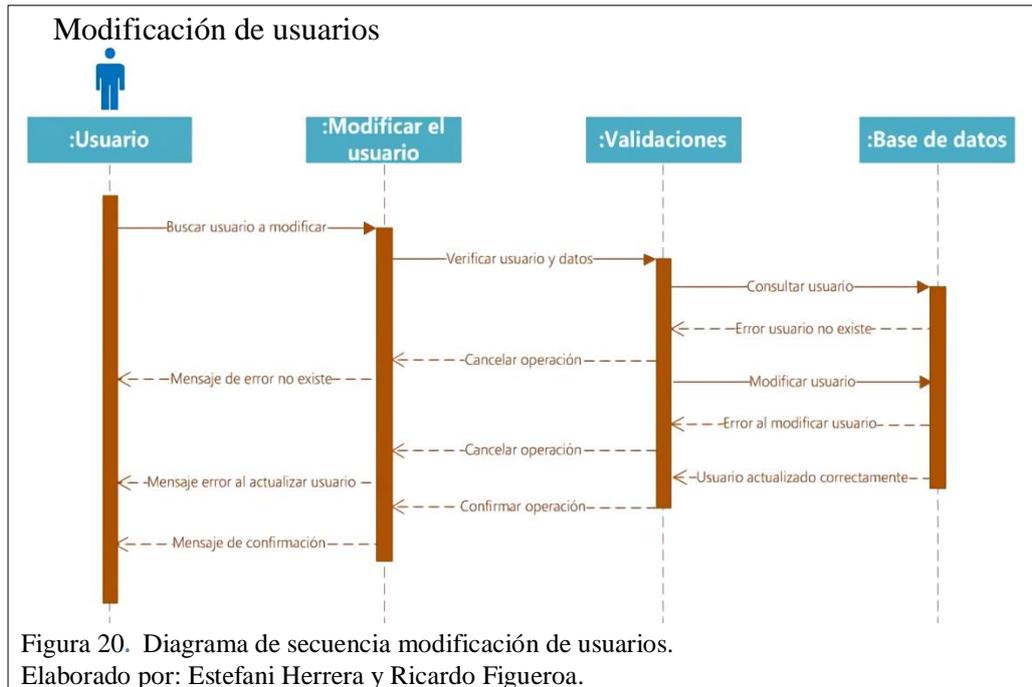
La Figura 18 el diagrama de secuencia que detalla las acciones realizadas por el actor en este caso el usuario, coloca el dedo en el sensor biométrico, valida la huella digital, consulta el usuario en la base de datos e inicia sesión.



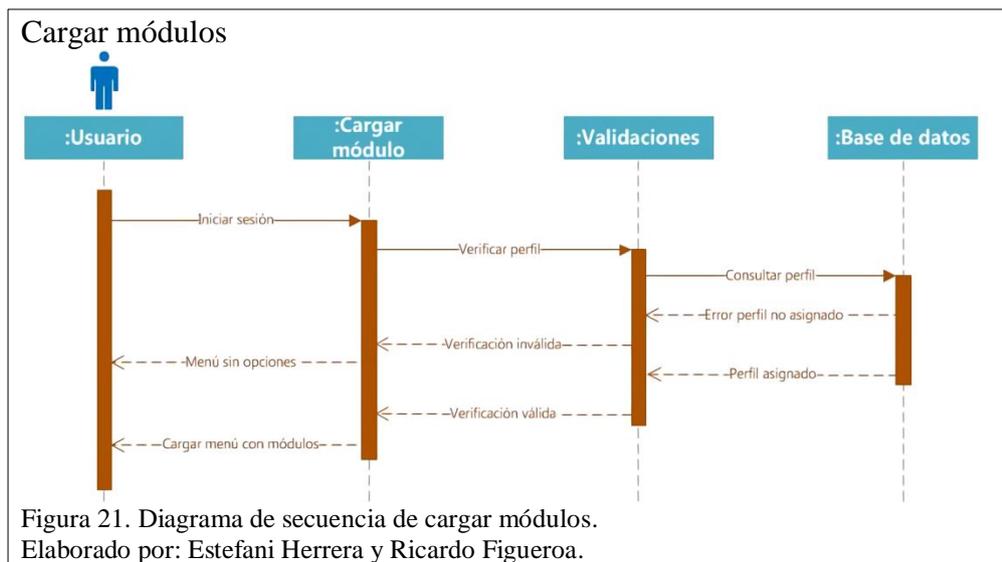
La Figura 19 el diagrama de secuencia que detalla las acciones realizadas por el actor en este caso el administrador, ingresa información, valida los datos ingresados e ingresa nuevo usuario.



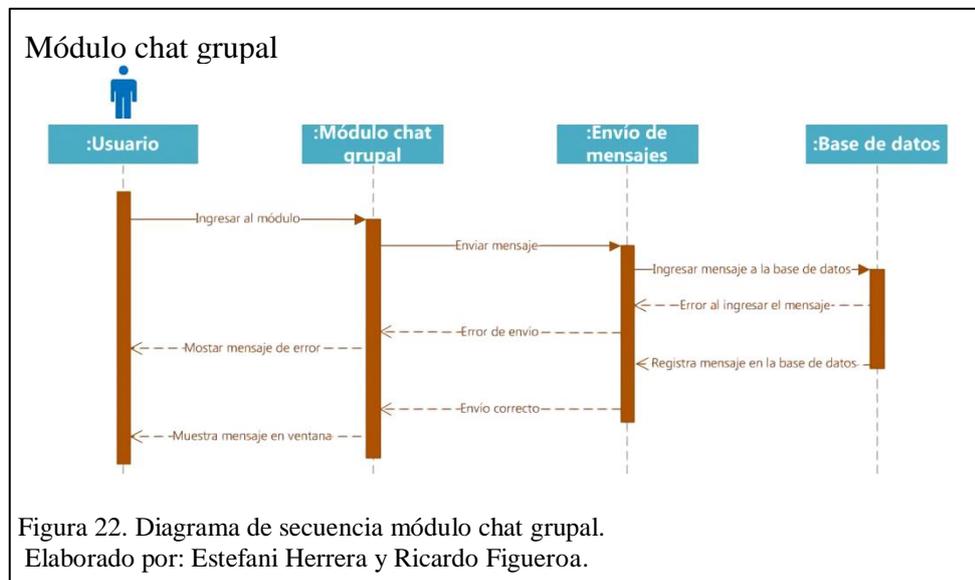
La Figura 20 el diagrama de secuencia que detalla las acciones realizadas por el actor en este caso el administrador, busca usuario a modificar, verifica usuario y datos, consulta usuario en la base de datos y modifica usuario.



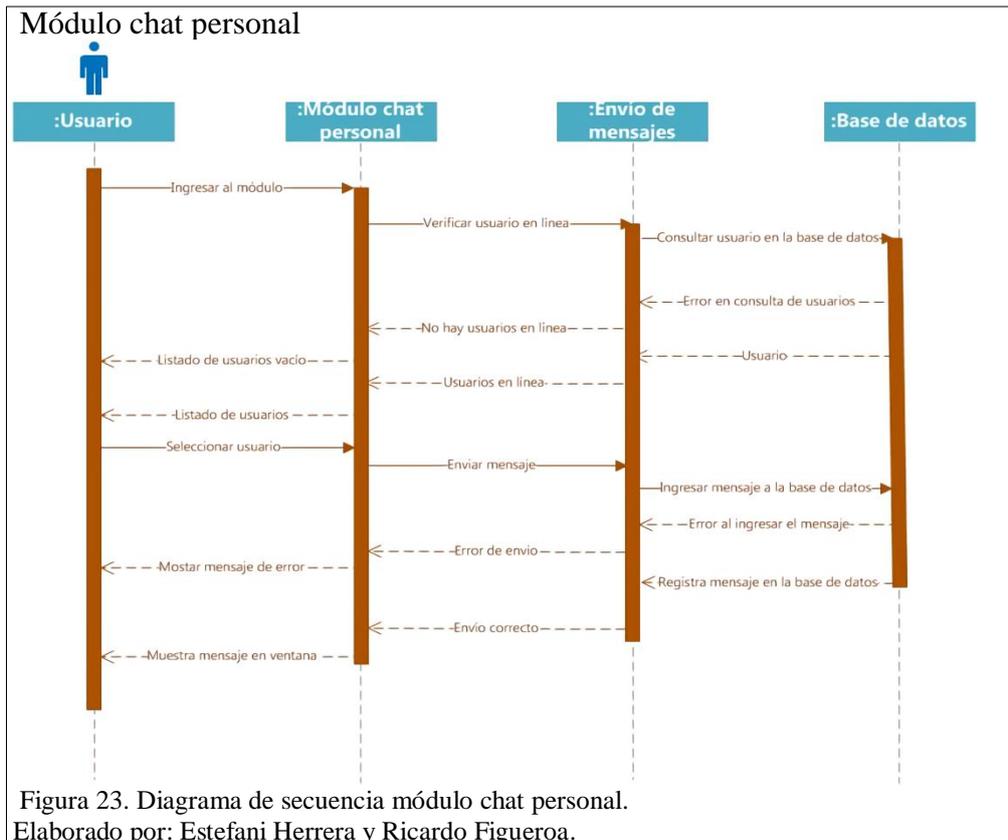
La Figura 21 el diagrama de secuencia que detalla las acciones realizadas por el actor en este caso el administrador, inicia sesión, verifica perfil, consulta perfil y carga el menú de los módulos.



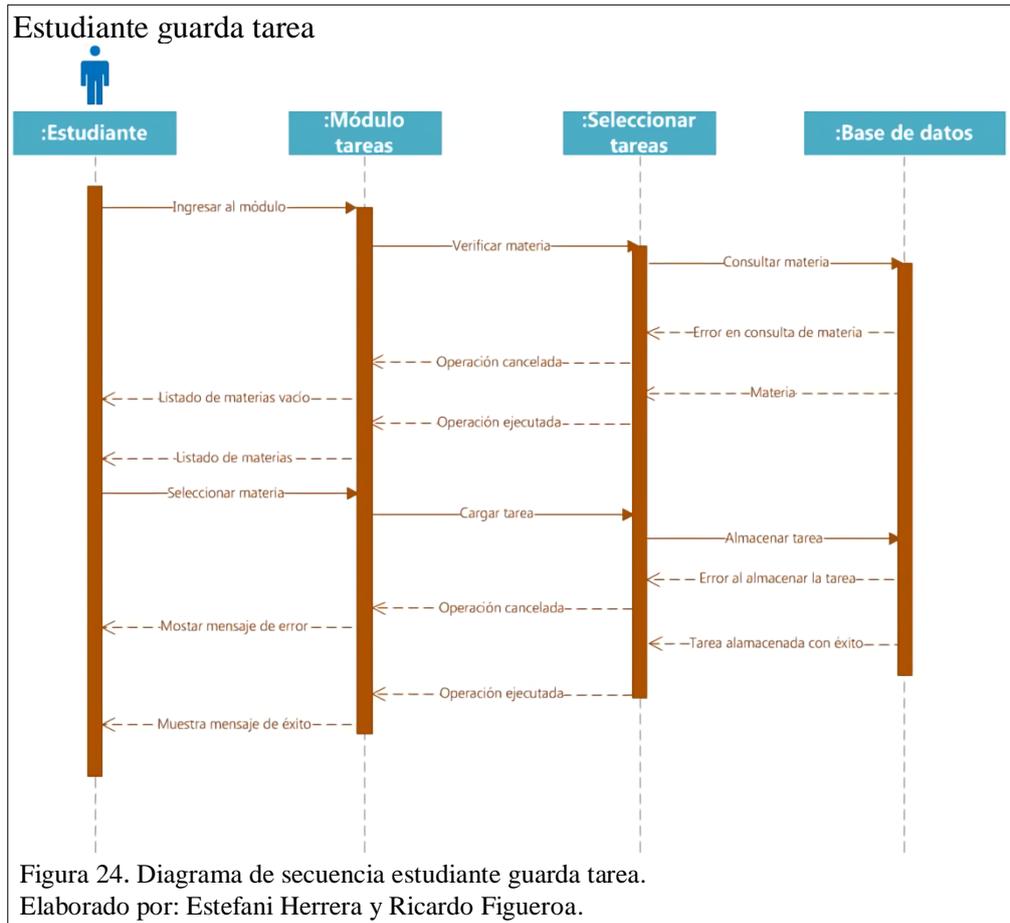
La Figura 22 el diagrama de secuencia que detalla las acciones realizadas por el actor en este caso el usuario, ingresa al módulo chat grupal, envía el mensaje y el mismo se registra en la base de datos, posteriormente se muestra el mensaje en la ventana.



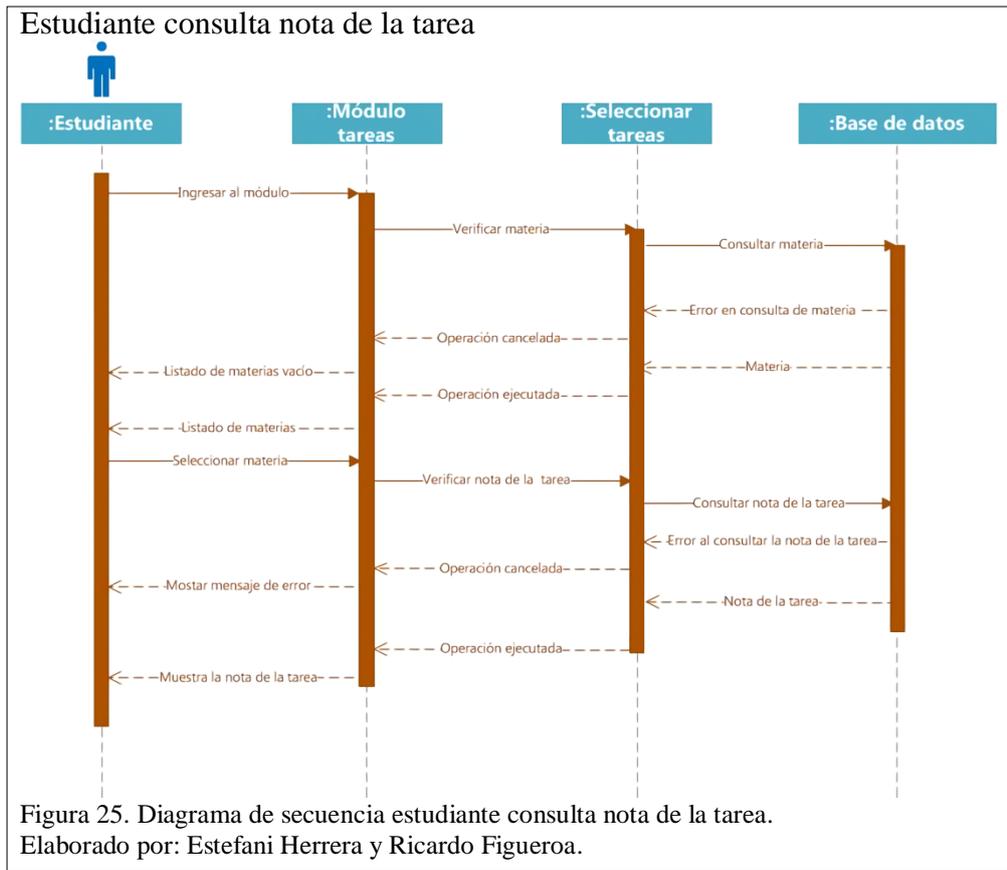
La Figura 23 el diagrama de secuencia que detalla las acciones que realiza el actor en este caso el usuario, ingresa al módulo chat personal, verifica usuario en línea, selecciona el usuario, envía el mensaje y el mismo se registra en la base de datos, posteriormente se muestra el mensaje en la ventana.



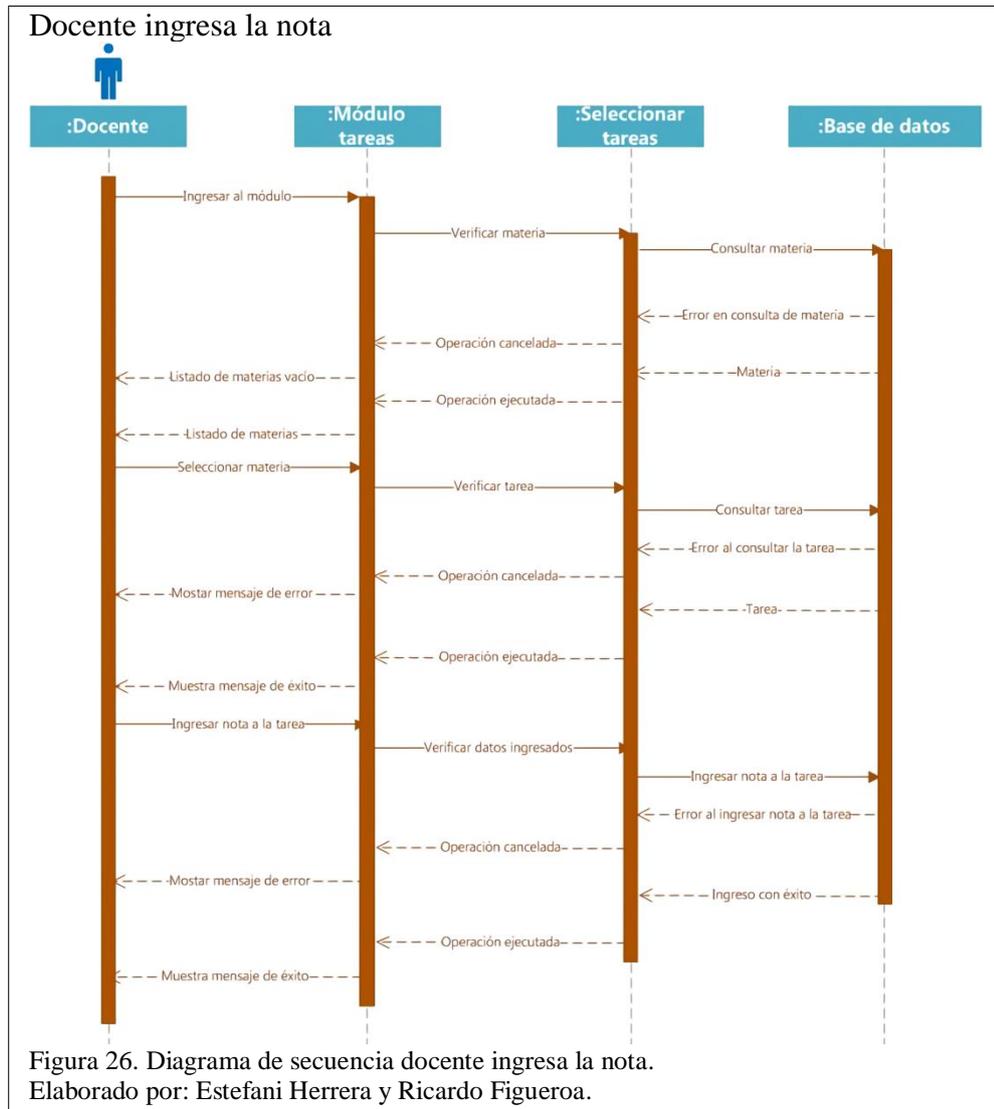
La Figura 24 el diagrama de secuencia que detalla las acciones realizadas por el actor en este caso el estudiante, ingresa al módulo tareas, selecciona la materia, verifica la materia, carga la tarea, consulta materia y guarda la tarea en la base de datos.



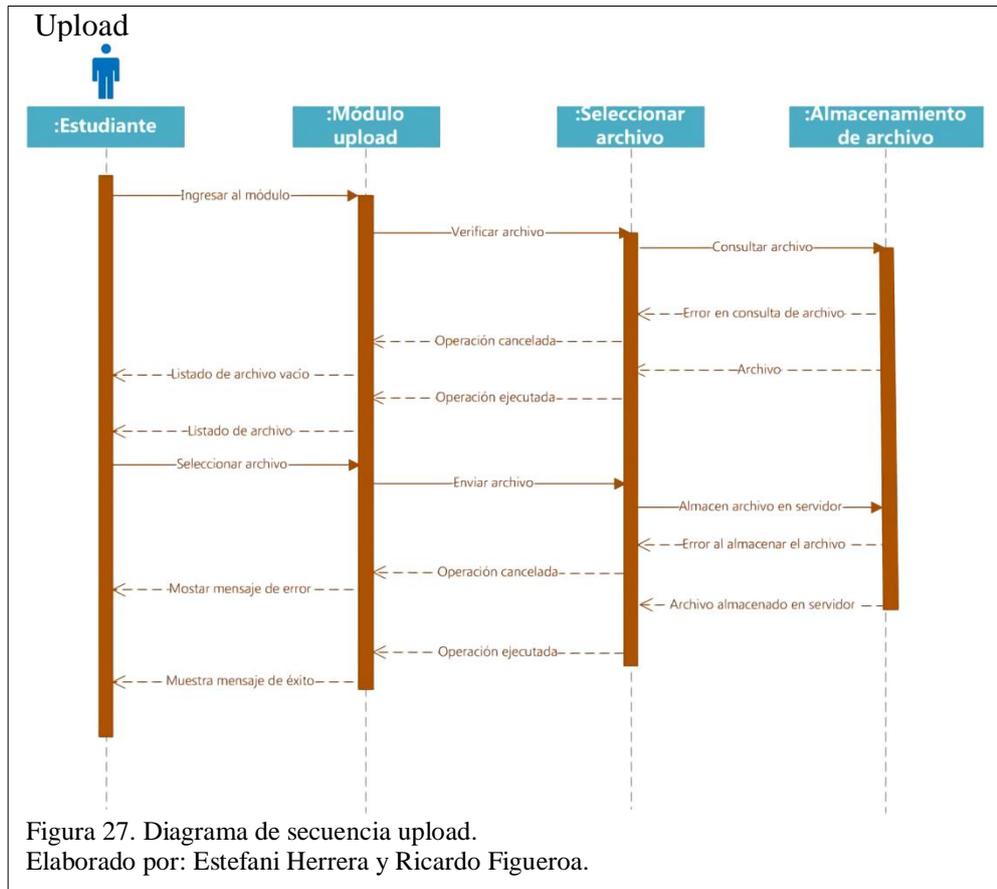
La Figura 25 el diagrama de secuencia que detalla las acciones realizadas por el actor en este caso el estudiante, ingresa al módulo tareas, selecciona la materia, verifica la materia, verifica nota de la tarea, consultar materia, consultar nota de la tarea y muestra la nota de la tarea.



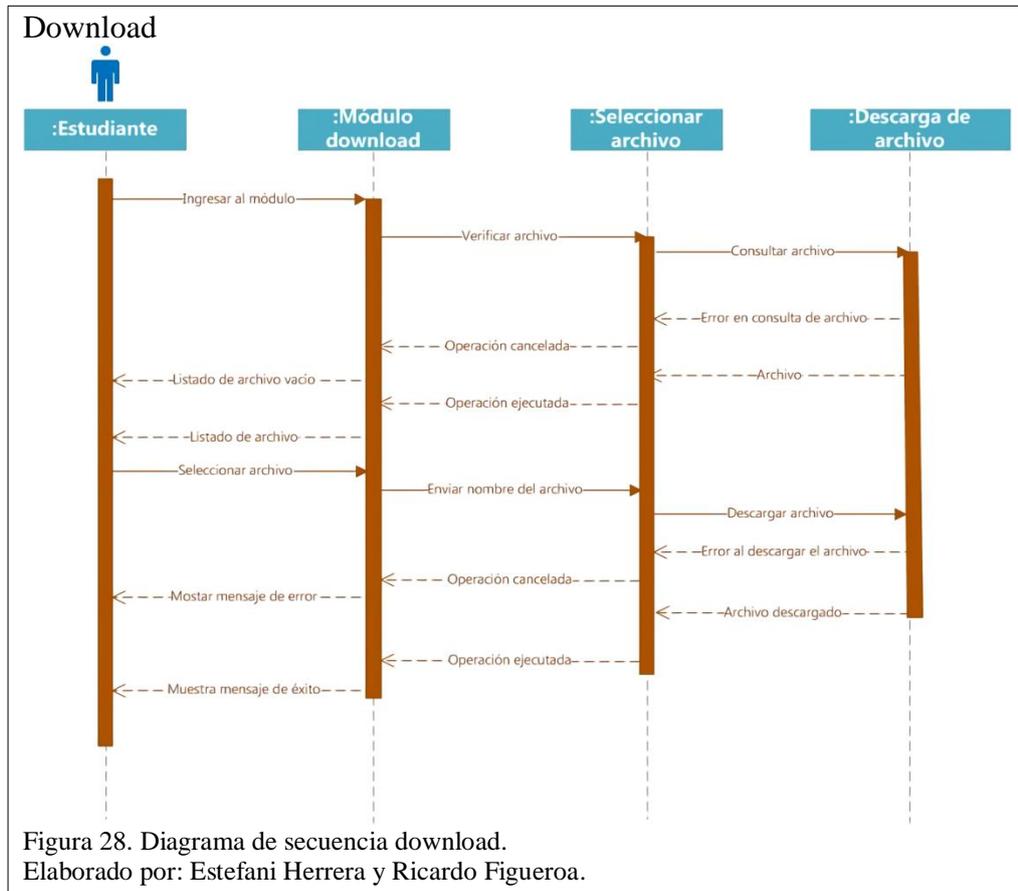
La Figura 26 el diagrama de secuencia que detalla las acciones realizadas por el actor en este caso el docente, ingresa al módulo tareas, selecciona la materia, verifica materia, tarea y datos ingresados, consulta materia y tarea, ingresa nota.



La Figura 27 el diagrama de secuencia que detalla las acciones realizadas por el actor en este caso el estudiante, ingresa al módulo repositorio, ingresa a upload, selecciona el archivo, verifica archivo, envía el archivo, consulta archivo y almacena archivo en el servidor.



La Figura 28 el diagrama de secuencia que detalla las acciones realizadas por el actor en este caso el estudiante, ingresa al módulo repositorio, ingresa a download, selecciona el archivo, verifica archivo, enviar nombre del archivo, consulta archivo y descarga el archivo.



4.5. Diagrama de Flujo de Datos

En esta sección muestra el esquema de la aplicación.

La Figura 29 en este diagrama de flujo de datos se muestra la interacción total de los diferentes módulos de la aplicación a nivel esquemático.

Diagrama de flujo del funcionamiento de la aplicación

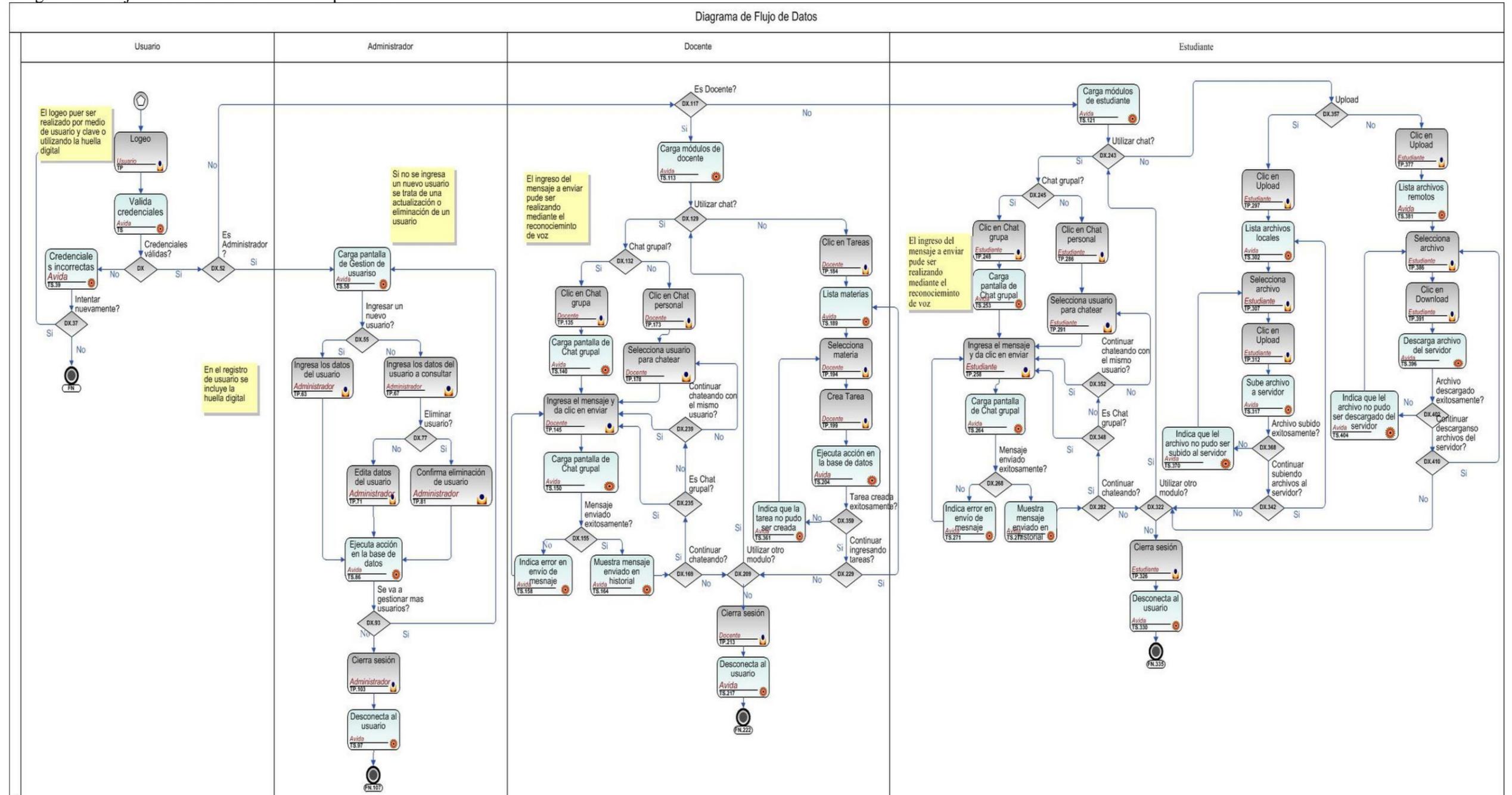


Figura 29. Diagrama de flujo del funcionamiento de la aplicación.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

La Figura 30 se muestra el logeo de usuarios y opciones de administrador de la aplicación, en donde ingresa al logeo, valida las credenciales ingresadas, carga la pantalla de gestión de usuarios para que el administrador ingrese, edite y elimine los datos de los usuarios.

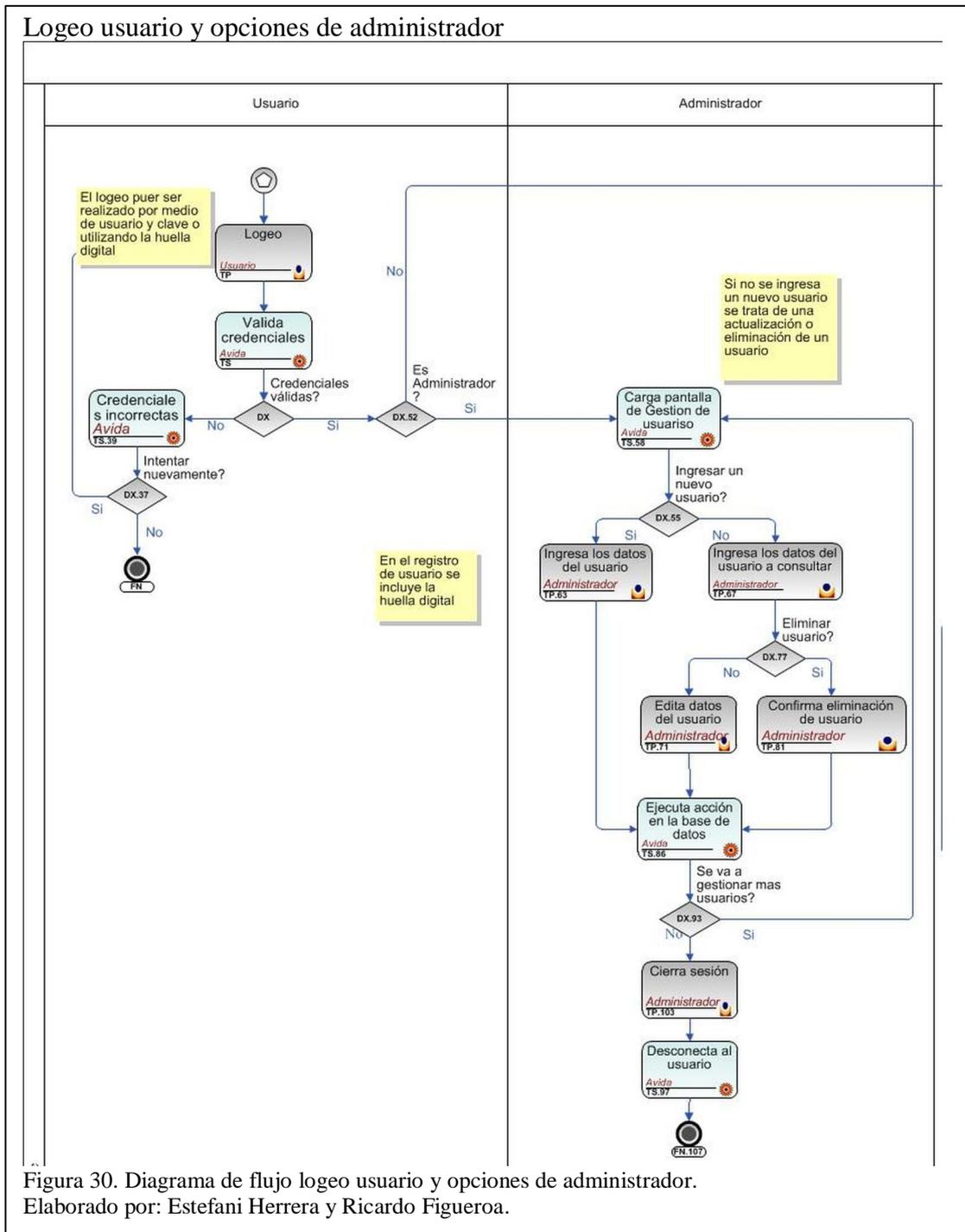


Figura 30. Diagrama de flujo logeo usuario y opciones de administrador. Elaborado por: Estefani Herrera y Ricardo Figueroa.

La Figura 31 muestra las opciones del docente, en donde carga los módulos para el docente: módulo chat grupal, personal y módulo tareas.

En el caso del módulo chat grupal, carga la pantalla del chat, muestra el historial de mensajes, ingresa el mensaje y envía. La diferencia del chat personal es que debe escoger el usuario.

En el caso de las tareas el docente escoge la materia e ingresa la nota.

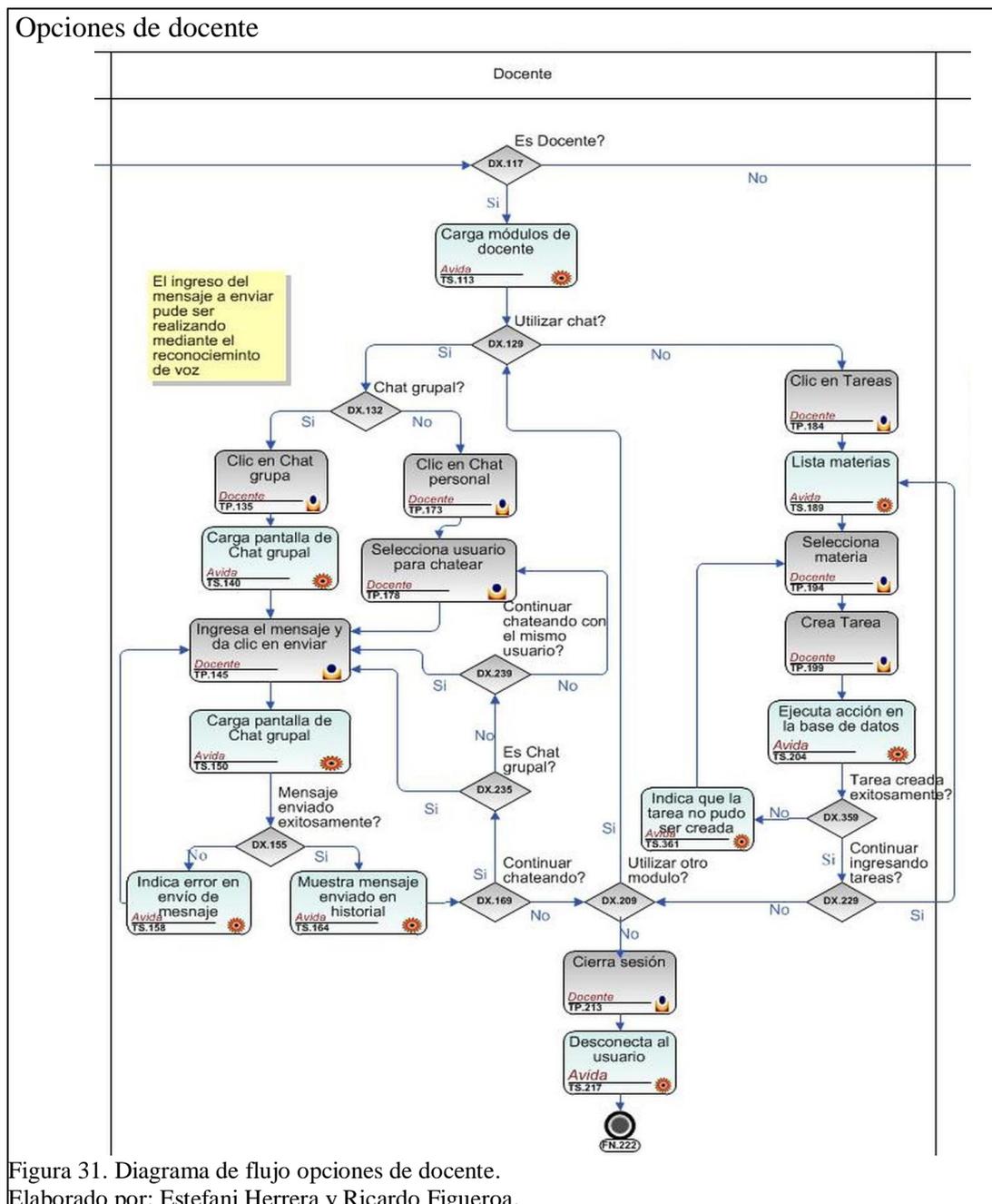


Figura 31. Diagrama de flujo opciones de docente.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

La figura 32 muestra las opciones del estudiante, en donde se carga los módulos: módulo chat grupal, chat personal y módulo repositorio

En el caso del módulo chat grupal, carga la pantalla del chat, muestra el historial de mensajes, ingresa el mensaje y envía. La diferencia del chat personal es que debe escoger el usuario.

En el caso del repositorio el estudiante tiene dos opciones upload y download, escoge cualquier opción, en el caso de upload, lista archivos locales, escoge el archivo y lo carga al servidor, escoge download y descarga el archivo.

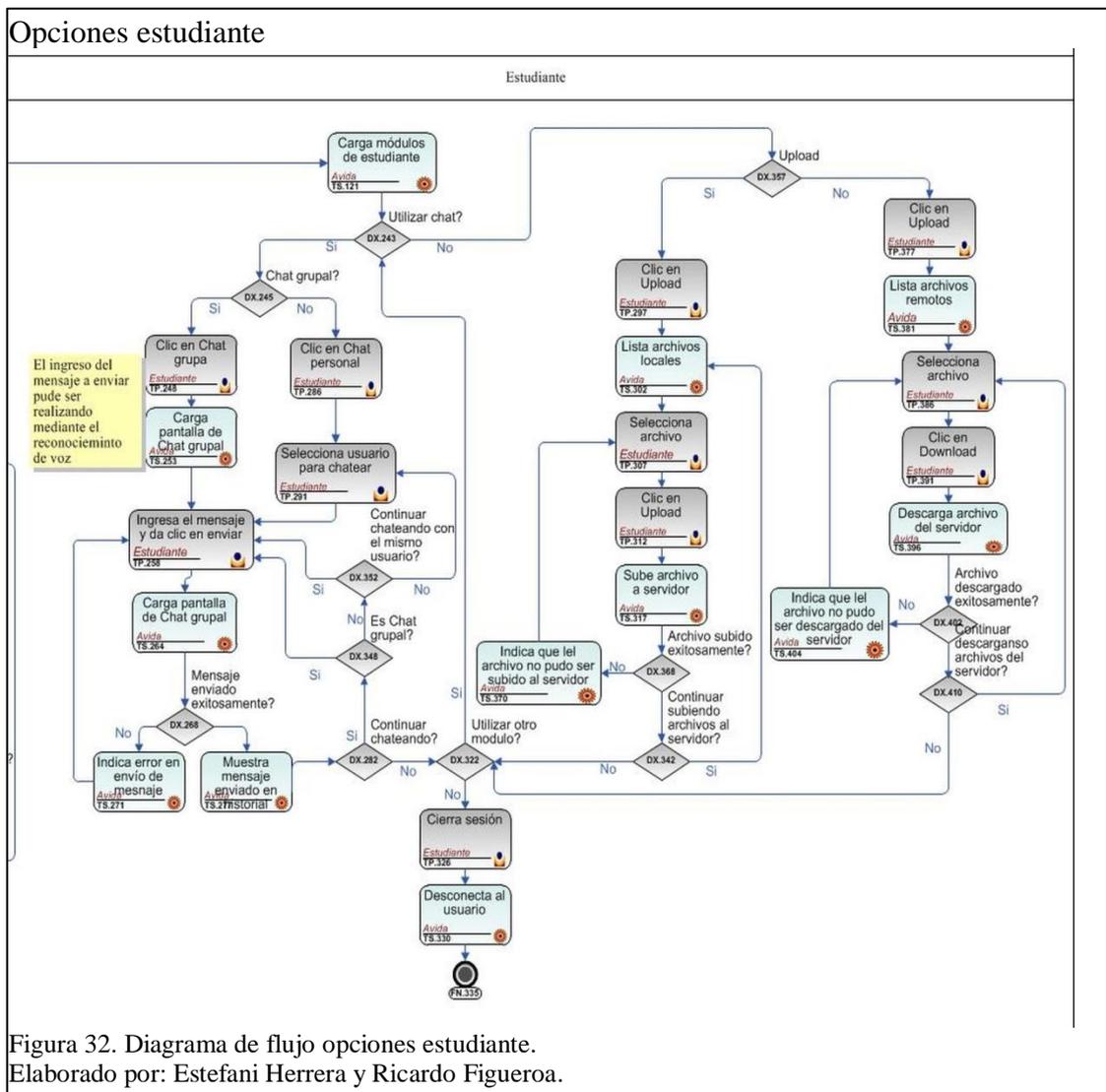


Figura 32. Diagrama de flujo opciones estudiante.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

4.6. Arquetipo de la aplicación

En esta sección se muestra las ilustraciones principales de la evolución del prototipo hasta llegar a su versión final.

El bosquejo que conforman todos los módulos de la aplicación se encuentran en el Anexo 4.

En la Figura 33 se visualiza el bosquejo del login de la aplicación.

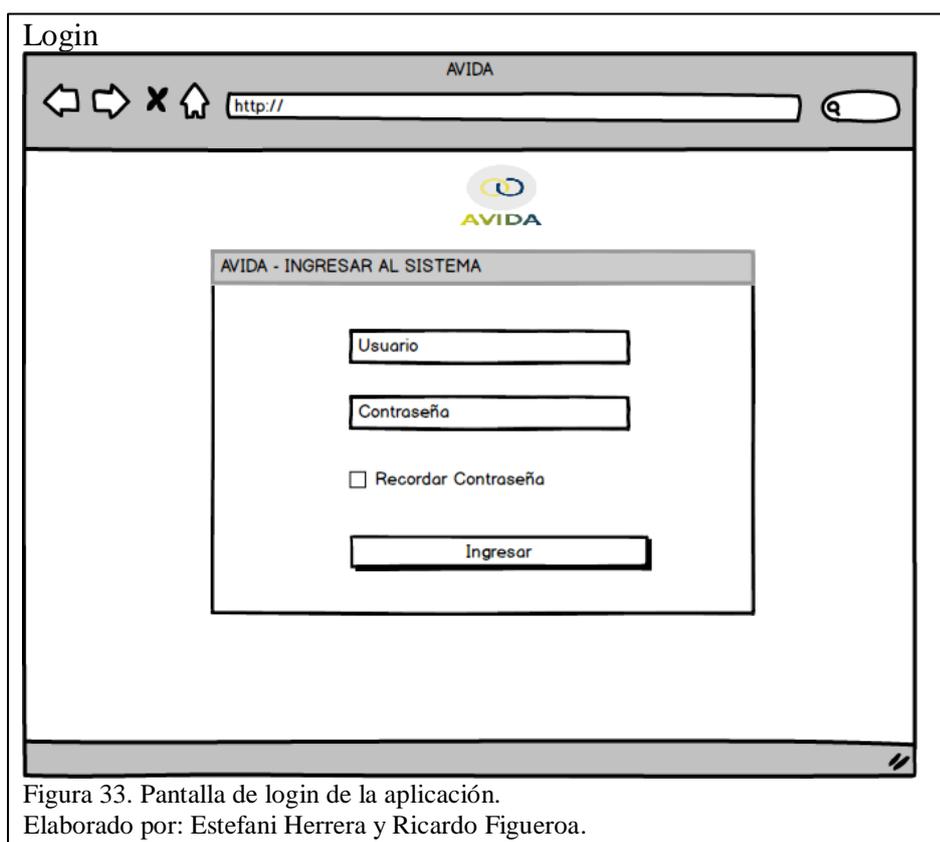


Figura 33. Pantalla de login de la aplicación.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 34 se visualiza el bosquejo del menú de la aplicación.

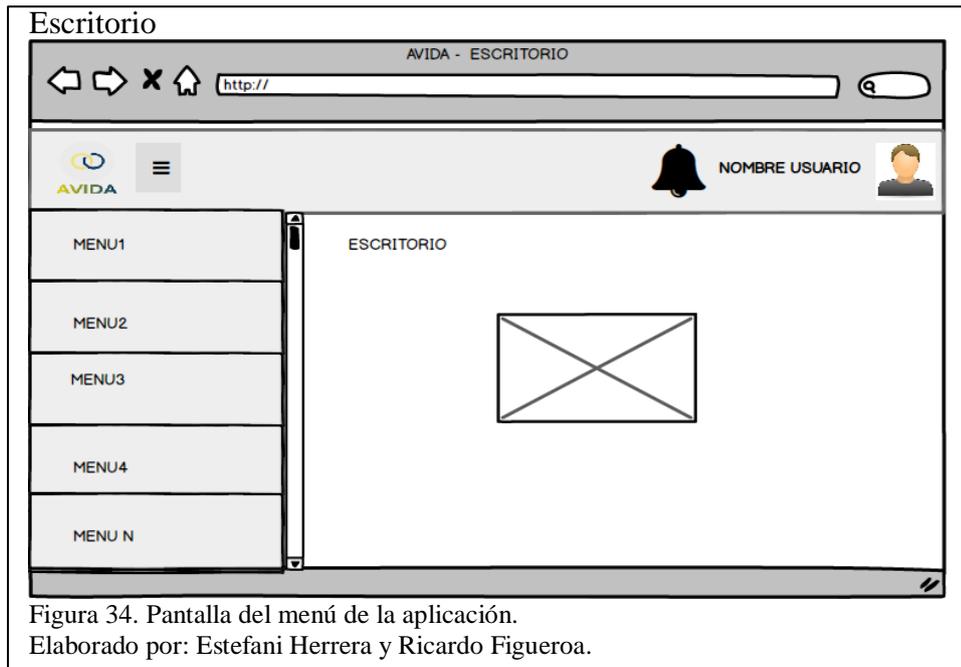


Figura 34. Pantalla del menú de la aplicación.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 35 se visualiza el bosquejo del módulo chat de la aplicación.

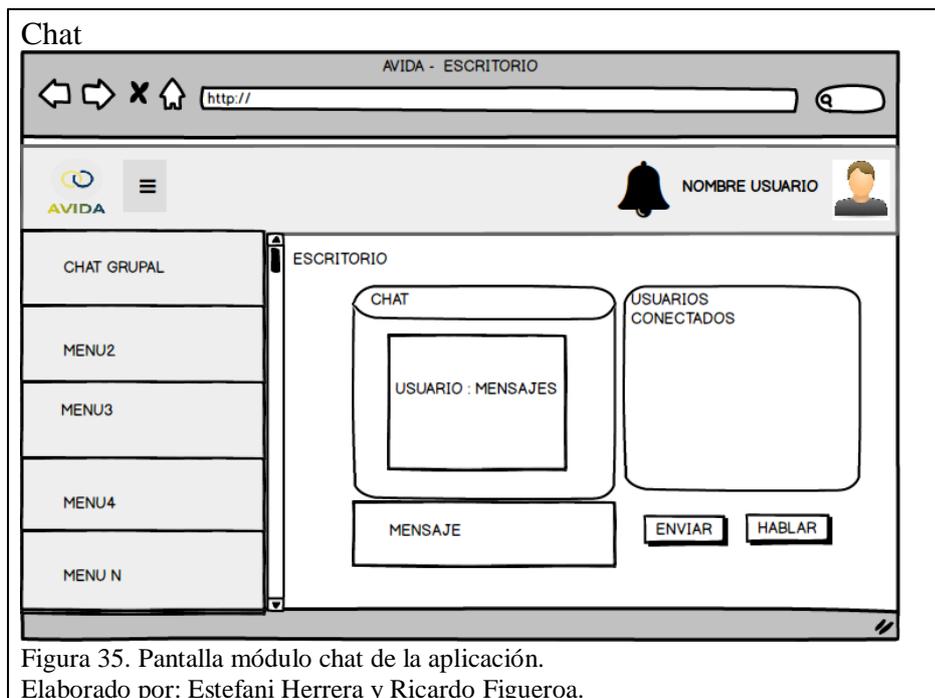


Figura 35. Pantalla módulo chat de la aplicación.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 36 se visualiza el bosquejo del módulo calificar tarea de la aplicación.

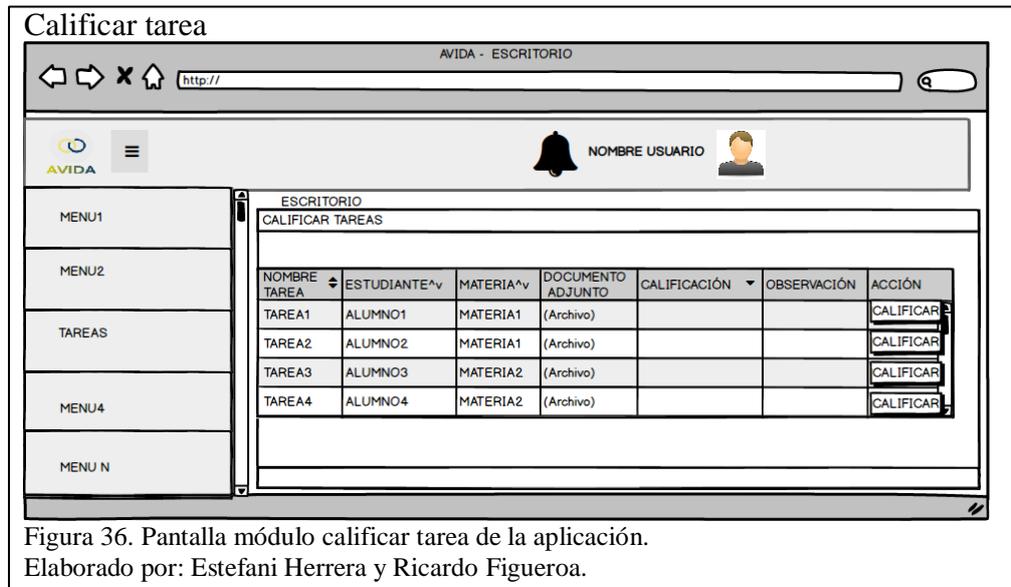


Figura 36. Pantalla módulo calificar tarea de la aplicación.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 37 se visualiza el bosquejo del módulo manejador de archivos de la aplicación.

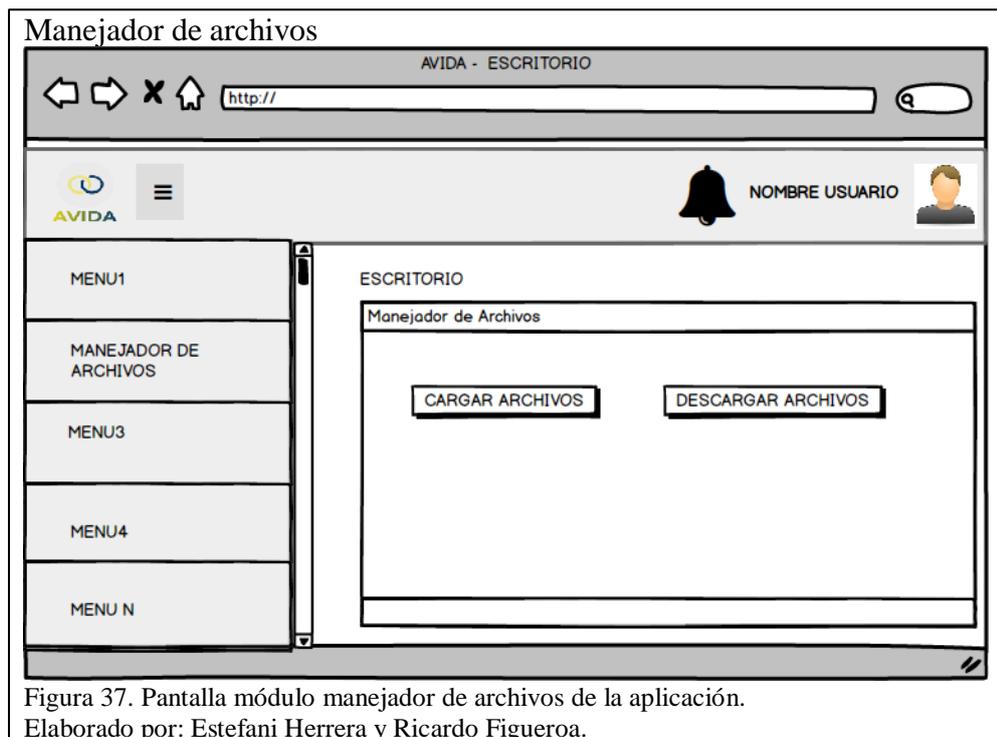


Figura 37. Pantalla módulo manejador de archivos de la aplicación.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

4.7. Descripción de clases (con código relevante)

Durante este capítulo, se muestra la codificación de las diferentes clases que permiten construir los módulos previamente diseñados.

4.7.1. Módulo Login

La codificación en la Figura 38 hace referencia al login utilizando el lector de huella digital, en la línea 81 se indica el evento de teclado y en este caso el accionante es la tecla ctrl que al presionarle ejecutará la validación de huella tal como se indica en la línea 86.

```
Login verificar huella digital
77 <script>
78     document.onkeydown = function(evt) {
79         //tecla CONTROL invocación huella digital
80         evt = evt || window.event;
81         if (evt.keyCode == 17) {
82             var text = "POR FAVOR COLOCA TU DEDO EN EL LECTOR DE HUELLA DIGITAL";
83             text= encodeURIComponent(text);
84             var url="https://audio1.spanishdict.com/audio?lang=es&text="+text;
85             $("audio").attr('src',url).get(0).play();
86             $("#capa").load('VerificaHuella.jsp');
```

Figura 38. Código login verificar huella digital.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 39 presenta el código utilizado para identificar la huella del usuario, en la línea 365 se llama a un procedimiento almacenado que nos traerá la información del usuario incluido la huella digital, en la línea 376 con el dato de la huella digital se genera un template para su verificación como lo indica la línea 380 (JARC BLOG, 2011).

Identifica huella digital

```
365     CallableStatement cs = c.prepareStatement("CALL PROC_OBTENER_HUELLA_DIGITAL (0)");
366     ResultSet rs = cs.executeQuery();
367     while(rs.next()){
368         byte templateBuffer[] = rs.getBytes("huella");
369         String nombre=rs.getString("Nombre");
370         String apellido=rs.getString("Apellidos");
371         String usuario=rs.getString("Usuario");
372         String estadoU=rs.getString("estadoUsuario");
373         String Clave= rs.getString("Password");
374         String id_usuario=Integer.toString(rs.getInt("id_usuario"));
375         //Genera una plantilla de la huella digital guardada en la BDD
376         DFPPTemplate referenceTemplate = DFPGlobal.getTemplateFactory().createTemplate(templateBuffer);
377         //Envia la plantilla creada al objeto contenedor de Template del componente de huella digital
378         setTemplate(referenceTemplate);
379         // Compara huella guardada en la BDD con la capturada en el lector
380         DFPVerificationResult result = Verificador.verify(featuresverificacion, getTemplate());
```

Figura 39. Código Identifica huella digital.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

4.7.2. Módulo Chat Grupal y Personal

En la Figura 40 presenta el código para el envío de mensajes utilizando el reconocimiento de voz, se tiene en la línea 140 las acciones que permiten iniciar el reconocimiento de voz previo al envío del mensaje línea 151 (Kai Wedekind, 2018).

Método procesar chat

```
138     function procesar() {
139         if (recognizing == false) {
140             recognition.start();
141             recognizing = true;
142             document.getElementById("procesar").innerHTML = "Detener";
143         } else {
144             recognition.stop();
145             recognizing = false;
146             document.getElementById("procesar").innerHTML = "Escuchar";
147
148             if (document.getElementById("txtMensaje").value == "") {
149
150             } else {
151                 enviarMensaje();
152             }
153         }
154     }
```

Figura 40. Código procesar chat.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 41 presenta el código desarrollado para cargar el historial de mensajes enviados, en la línea 24 se visualiza que en la variable mensajes se carga la lista de mensajes del historial y se procede a imprimirlo en la pantalla línea 25.

```
Cargar mensaje chat
11 <%@page import="java.nio.charset.StandardCharsets"%>
12
13 <%@page contentType="text/html" pageEncoding="UTF-8"%>
14 <!DOCTYPE html>
15 <html>
16 <head>
17 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
18 <meta name="viewport" content="width=device-width, initial-scale=1">
19 <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min
20 </head>
21 <body>
22 <%
23
24 String mensajes= NMensajes.verMensajes();
25 out.println(URLDecoder.decode(mensajes, StandardCharsets.UTF_8.toString()));
```

Figura 41. Código cargar mensaje chat.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 42 presenta el código utilizado para realizar el reconocimiento de voz y convertirlo en texto.

En la línea 100 se verifica que el navegador soporte el API de reconocimiento de voz, en las líneas 103 a 106 se configura el API primero creando una nueva instancia e indicándole que el idioma a ser utilizado es el español.

Una vez iniciado el reconocimiento de voz, lo escuchado será escrito en un cuadro de texto llamado txtMensaje como lo indica la línea 114, para controlar el reconocimiento de voz se utiliza las acciones de inicio onstart línea 107, de resultado onresult línea 111 y de detener la escucha onend línea 119 (Kai Wedekind, 2018).

Reconocimiento de voz chat

```
98     var recognition;  
99     var recognizing = false;  
100    if (!('webkitSpeechRecognition' in window)) {  
101        alert("¡API no soportada!");  
102    } else {  
103        recognition = new webkitSpeechRecognition();  
104        recognition.lang = "es-EC";  
105        recognition.continuous = true;  
106        recognition.interimResults = true;  
107        recognition.onstart = function () {  
108            recognizing = true;  
109            console.log("empezando a escuchar");  
110        }  
111        recognition.onresult = function (event) {  
112            for (var i = event.resultIndex; i < event.results.length; i++) {  
113                if (event.results[i].isFinal)  
114                    document.getElementById("txtMensaje").value += event.results[i][0].transcript;  
115            }  
116        }  
117        recognition.onerror = function (event) {  
118        }  
119        recognition.onend = function () {  
120            recognizing = false;  
121            document.getElementById("procesar").innerHTML = "Escuchar";  
122            console.log("terminó de escuchar, llegó a su fin");  
123        }  
124    }
```

Figura 42. Código reconocimiento de voz.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 43 presenta el código desarrollado para convertir el texto en audio, en la línea 158 se obtiene el texto que se ha escrito en el cuadro de texto llamado txtMensaje y lo se lo transforma en audio en la línea 159.

Método hablar chat

```
157     hablar = function () {  
158         h.text = document.getElementById("txtMensaje").value;  
159         speechSynthesis.speak(h);  
160     }  
161     function updateDiv()  
162     {  
163         $("#aux").load('LeerMensaje.jsp');  
164     }
```

Figura 43. Código método hablar chat.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 44 presenta el código utilizado para configurar el idioma tanto en el reconocimiento de voz y convertir texto en audio, como se observa en la línea 126 se ha definido que el idioma sea español (Kai Wedekind, 2018).

```
Cambiar idioma chat
125     var h = new SpeechSynthesisUtterance();
126     h.lang = "es-EC";
127     // Cargamos la voz de 'es-EC'
128     window.speechSynthesis.onvoiceschanged = function () {
129         voices = window.speechSynthesis.getVoices();
130         for (var i = 0; i < voices.length; i++) {
131             if (voices[i]['lang'] == 'es-EC') {
132                 h.voice = voices[i];
133             }
134         }
135     };
```

Figura 44. Código cambiar idioma chat.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 45 presenta el código utilizado para enviar el texto ingresado por el usuario, en la línea 54 se obtiene el mensaje y se lo envía a enviarmensaje.jsp como parámetro líneas 57 y 59 (Foros del web, 2013).

```
Método para enviar mensajes chat js
53     function enviarMensaje() {
54         var x = $("#txtMensaje").val();
55         $.ajax({
56             type: "POST",
57             url: "enviarmensaje.jsp",
58             beforeSend: inicioCargaMSJ,
59             data: 'mensaje='+x,
60             success: llegadaMSJ,
61             error: problemasMSJ
62         });
63         $("#txtMensaje").val("");
64         $("#txtMensaje").focus();
65     }
```

Figura 45. Código método para enviar mensajes chat js.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 46 presenta en las líneas 20 y 21 que se obtiene como parámetros el usuario y el texto del mensaje, esta información es ingresada en la base de datos utilizando el método ingresarMensaje línea 26 (Foros del web, 2013).

```
Enviar mensaje chat jsp
19      <%
20      String mensaje = request.getParameter("mensaje");
21      String user = request.getSession().getAttribute("userId").toString();
22      if(mensaje.equalsIgnoreCase("")){
23      }else{
24          EMensaje m = new EMensaje(mensaje,user);
25
26          int res = new NChatBLL().ingresarMensaje(m);
27          if(res!=-1){
28              out.println("Error al ingresar mensaje");
29          }else{
30              new ESonido().EnvioMensaje("0");
31          }
32      }
33      %>
```

Figura 46. Código enviar mensaje chat jsp.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 47 presenta el código utilizado para visualizar el listado de usuarios conectados, en la línea 21 se verifica que en la variable de sesión haya usuarios logeados, de haber usuarios logeados se observa en línea 23 (Foros del web, 2013).

```
Usuarios conectados
19      <%
20      String img = "<img src=\"imagenes/on.png\" />";
21      if(!Sesiones.usuarios.isEmpty()){
22          for(String s:Sesiones.usuarios){
23              out.println(img+" <b>"+s+"</b><br>");
24          }
25      }
26      else{
27          out.println("<b>No existen usuarios conectados.</b>");
28      }
29      %>
```

Figura 47. Código usuarios conectados.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

4.7.3. Módulo Tareas

En la Figura 48 presenta el código utilizado para guardar la tarea en la base de datos, entre las líneas 13 y 18 se obtiene el archivo, su nombre, tamaño y contenido, se lo almacena en un inputStream para guardar en el repositorio de datos indicando quién lo guardo como se muestra en la línea 31.

```
Guarda documento
11      InputStream inputStream = null;
12      try {
13          Part filePart = request.getPart("documentotarea");
14          if (filePart.getSize() > 0) {
15              System.out.println(filePart.getName());
16              System.out.println(filePart.getSize());
17              System.out.println(filePart.getContentType());
18              inputStream = filePart.getInputStream();
19          }
20      } catch (Exception ex) {
21          System.out.println("fichero: "+ex.getMessage());
22      }
23
24      ETareas tareas = new ETareas();
25
26      tareas.setIdMateria(materia)
27      tareas.setDescripcion(descripcion)
28      tareas.setNombre(nombretarea)
29      tareas.setFechaInicio(fechaInicio)
30      tareas.setFechaFin(fechaFin)
31      boolean resp = NTareas.registrarTarea(usu);
```

Figura 48. Código guarda documento.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

4.7.4. Módulo Repositorio

4.7.4.1. Upload

En la figura 49 presenta el código desarrollado para listar los archivos que se encuentran en la carpeta local definida, en las líneas 63 a la 67 se indica el destino de la conexión (IP), el puerto a utilizar, el usuario, la contraseña y el protocolo que será utilizado (SFTP), y en la línea 70 se lista el contenido del directorio.

Listar archivos

```
57 public ArrayList<String> ListaFileSFTP(String ip) {
58     JSch jsch = new JSch();
59     Session session = null;
60     Vector<ChannelSftp.LsEntry> lsVec=null;
61     ArrayList<String> list = new ArrayList<String>();
62     try {
63         session = jsch.getSession("openssh", ip, 22);
64         session.setConfig("StrictHostKeyChecking", "no");
65         session.setPassword("openssh2019!");
66         session.connect();
67         Channel channel = session.openChannel("sftp");
68         channel.connect();
69         ChannelSftp sftpChannel = (ChannelSftp) channel;
70         lsVec=(Vector<ChannelSftp.LsEntry>)sftpChannel.ls("/C:/upload/");
```

Figura 49. Código listar archivos.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 50 presenta el código desarrollado para subir el archivo al servidor evitando que se sobrescriba por tener el mismo nombre utilizando la fecha línea 31 y hora línea 32 en la que el archivo fue subido como se observa en las línea 35.

Clase upload

```
29 DateFormat hourdateFormat = new SimpleDateFormat("dd-MM-yyyy");
30 DateFormat hourdateFormat1 = new SimpleDateFormat("hh:mm:ss");
31 fecha=hourdateFormat.format(date);
32 hora=hourdateFormat1.format(date);
33 fechaN=fecha.replaceAll("-", "");
34 horaN=hora.replaceAll(":", "");
35 String nombre=part1+fechaN+horaN+"."+part2;
36 JSch jsch = new JSch();
37 Session session = null;
38 try {
39     session = jsch.getSession("openssh", ip, 22);
40     session.setConfig("StrictHostKeyChecking", "no");
41     session.setPassword("openssh2019!");
42     session.connect();
43     Channel channel = session.openChannel("sftp");
44     channel.connect();
45     ChannelSftp sftpChannel = (ChannelSftp) channel;
```

Figura 50. Código clase upload.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

4.7.4.2. Download

```
Download
165 try{
166     FileInputStream archivo = new FileInputStream("D:/download/"+nomb);
167     int longitud = archivo.available();
168     byte[] datos = new byte[longitud];
169     archivo.read(datos);
170     archivo.close();
171     ServletOutputStream ouputStream = response.getOutputStream();
172     ouputStream.write(datos);
173     ouputStream.flush();
174     ouputStream.close();
```

Figura 51. Download.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 51 presenta el código desarrollado para la descarga de archivos, en la línea 160 se obtiene el nombre del archivo a ser descargado, crea la variable archivo de tipo FileInputStream utilizando el archivo seleccionado por el usuario, en la línea 172 realiza la descarga del archivo a la máquina local.

4.8. Prototipo final de la aplicación

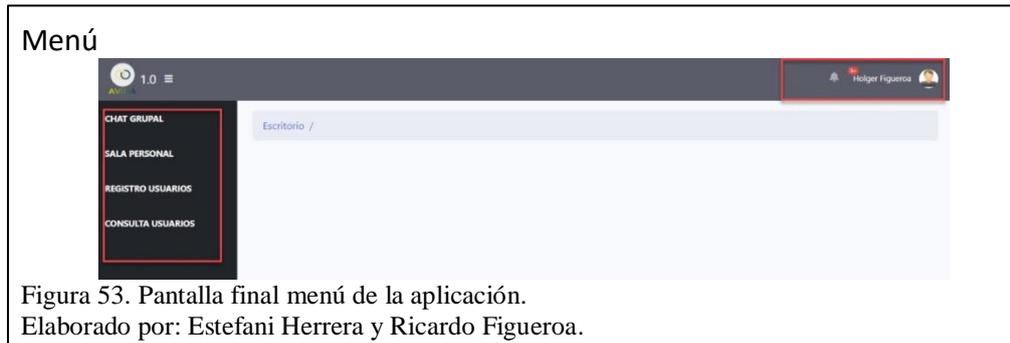
En esta sección se muestra las ilustraciones finales de la aplicación. El manual de usuario de la aplicación se encuentra en el Anexo 3.

En la Figura 52 presenta la pantalla final del login de la aplicación.

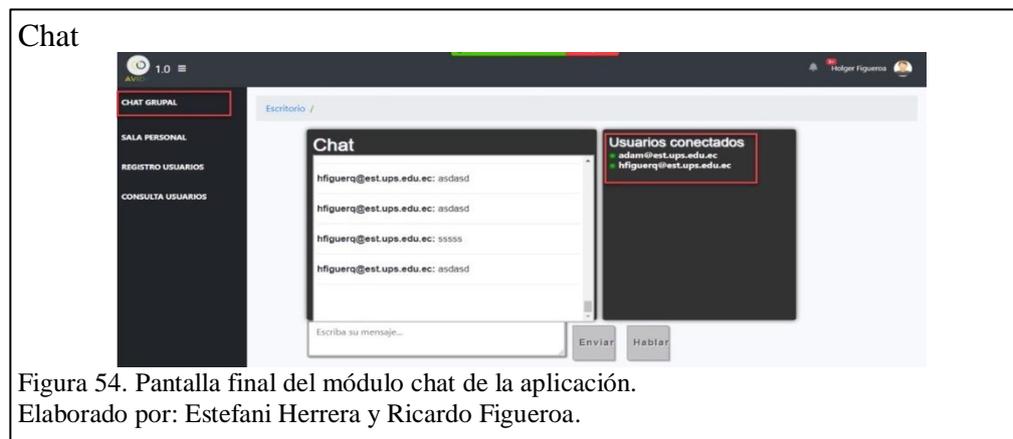


Figura 52. Pantalla final del login de la aplicación.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Figura 53 presenta la pantalla final del menú de la aplicación.



En la Figura 54 presenta la pantalla final del módulo chat de la aplicación.



En la Figura 55 presenta la pantalla final del módulo manejador de archivos de la aplicación.



CAPITULO 5

PRUEBAS

En este capítulo se procede a realizar las diferentes pruebas de funcionalidad de la aplicación.

5.1. Pruebas de Caja Negra

Verifica la funcionalidad de los módulos que conforman la aplicación. Se enfoca en entradas y salidas de la aplicación, para obtener este detalle se basa en los requerimientos funcionales.

En la Tabla 16 muestra los resultados para el módulo login.

Tabla 16. Prueba de caja negra login

N	Módulo	Acción	Resultado
1	LOGIN	Ingreso de login incorrecto	Error: Credenciales no válidas
2		Ingreso de clave incorrecta	Error: Credenciales no válidas
3		Huella	Error de conexión al repositorio de datos
4		Huella	Error: Credenciales no válidas
5		Huella	Logeo
6		Ingreso de datos correctos	Error de conexión al repositorio de datos
7		Ingreso de datos correctos	Logeo

Nota: Detalles prueba de caja negra módulo login.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Tabla 17 muestra los resultados del módulo registro de usuarios.

Tabla 17. Prueba de caja negra módulo registro de usuarios

N	Módulo	Acción	Resultado
1	REGISTRO DE USUARIOS	Ingreso de nuevo usuario	Error de conexión al repositorio de datos
2		Ingreso de nuevo usuario	Error al insertar registro
3		Ingreso de nuevo usuario	Registro ingresado correctamente
4		Edición de usuario	Error de conexión al repositorio de datos
5		Edición de usuario	Error al actualizar el registro
6		Edición de usuario	Registro actualizado correctamente
7		Consulta de usuario	Error de conexión al repositorio de datos
8		Consulta de usuario	Resultado de la consulta
9		Eliminar usuario	Error de conexión al repositorio de datos
10		Eliminar usuario	Error al ejecutar eliminar registro
11		Eliminar usuario	Registro eliminado exitosamente

Nota: Detalles prueba de caja negra módulo registro de usuarios.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Tabla 18 muestra los resultados del módulo chat grupal.

Tabla 18. Prueba de caja negra módulo chat grupal

N	Módulo	Acción	Resultado
1	CHAT GRUPAL	Envío de mensaje	Error de conexión al repositorio de datos
2		Envío de mensaje	Error: Insertar datos
3		Envío de mensaje	Mensaje enviado

Nota: Detalles prueba de caja negra módulo chat grupal.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Tabla 19 muestra los resultados del módulo chat personal.

Tabla 19. Prueba de caja negra módulo chat personal

N	Módulo	Acción	Resultado
1	CHAT PERSONAL	Seleccionar usuario para chateo	Listado vacío no hay usuarios conectados
2		Seleccionar usuario para chateo	Listado de usuarios conectados
3		Envío de mensaje	Error de conexión al repositorio de datos
4		Envío de mensaje	Error: Insertar datos
5		Envío de mensaje	Mensaje enviado

Nota: Detalles prueba de caja negra módulo chat personal.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Tabla 20 muestra los resultados del módulo tareas.

Tabla 20. Prueba de caja negra módulo tareas

N	Módulo	Acción	Resultado
1	TAREAS	Lista materias	Error de conexión al repositorio de datos
2		Lista materias	Listado de materias
3		Crea tarea	Error de conexión al repositorio de datos
4		Crea tarea	Error: Insertar datos
5		Crea tarea	Tarea creada
6		Consulta tareas por materia	Error de conexión al repositorio de datos
7		Consulta tareas por materia	Tareas por materia
8		Sube tarea	Error de conexión al repositorio de datos
9		Sube tarea	Error: Insertar datos
10		Sube tarea	Tarea subida
11		Consulta nota tarea	Error de conexión al repositorio de datos
12		Consulta nota tarea	Nota tarea

Nota: Detalles prueba de caja negra módulo tareas.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Tabla 21 muestra los resultados del módulo upload.

Tabla 21. Prueba de caja negra módulo upload

N	Módulo	Acción	Resultado
1	UPLOAD	Seleccionar archivo para subir al servidor	Listado vacío, error de lectura
2		Seleccionar archivo para subir al servidor	Listado de archivos locales
3		Upload archivo	Error al subir el archivo, credenciales no válidas
4		Upload archivo	El archivo ha subido correctamente

Nota: Detalles prueba de caja negra módulo upload.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

En la Tabla 22 muestra los resultados del módulo download.

Tabla 22. Prueba de caja negra módulo download

N	Módulo	Acción	Resultado
1	DOWNLOAD	Seleccionar archivo para descargar del servidor	Listado vacío, error de lectura
2		Seleccionar archivo para descargar del servidor	Listado de archivos remotos
3		Download archivo	Error al descargar el archivo
4		Download archivo	Archivo descargado correctamente

Nota: Detalles prueba de caja negra módulo download.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

5.2. Pruebas de Aceptación

En este ítem se detalla las pruebas realizadas conjuntamente con los usuarios finales que en este caso se validó con dos estudiante con discapacidad visual de la tiflobiblioteca. Los resultados de la prueba de aceptación se han evaluado de acuerdo a la encuesta que se encuentran en el Anexo 5.

Tabla 23. Prueba de aceptación login

Prueba de Aceptación Login	
Código	AVIDA-001
Requerimiento evaluado	Autenticación tradicional y mediante huella digital.
Descripción	El usuario podrá autenticarse mediante credenciales o huella digital.
Resultado esperado	Los usuarios podrán acceder al sistema mediante credenciales o huella digital.
Resultado obtenido	Resultado esperado
Evaluación	Revisado por: <ul style="list-style-type: none"> • Christian Castañeda • Mauricio Suarez

Nota: Detalle prueba de aceptación login.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

Tabla 24. Prueba de aceptación chat

Prueba de Aceptación Chat	
Código	AVIDA-002
Requerimiento evaluado	Chat
Descripción	El usuario podrá enviar mensajes mediante el reconocimiento de voz y comandos del teclado.
Resultado esperado	Los usuarios podrán hacer uso del reconocimiento de voz para el envío de mensajes.
Resultado obtenido	Resultado esperado
Evaluación	Revisado por: <ul style="list-style-type: none"> • Christian Castañeda • Mauricio Suarez

Nota: Detalle prueba de aceptación chat.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

Tabla 25. Prueba de aceptación tareas

Prueba de Aceptación Tareas	
Código	AVIDA-003
Requerimiento evaluado	Tareas
Descripción	El usuario podrá enviar la tarea y podrá escuchar su nota.
Resultado esperado	Los usuarios podrán hacer uso del envío de tareas y cuál es su calificación.
Resultado obtenido	Resultado esperado
Evaluación	Revisado por: <ul style="list-style-type: none"> • Christian Castañeda • Mauricio Suarez

Nota: Detalle prueba de aceptación tareas.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

Tabla 26. Prueba de aceptación manejador de archivos

Prueba de Aceptación Manejador de archivos	
Código	AVIDA-004
Requerimiento evaluado	Manejado de archivos
Descripción	El usuario podrá subir y descargar cualquier tipo de archivo mediante lector de pantalla.
Resultado esperado	Los usuarios podrán hacer uso del repositorio.
Resultado obtenido	Resultado esperado
Evaluación	Revisado por: <ul style="list-style-type: none"> • Christian Castañeda • Mauricio Suarez

Nota: Detalle prueba de aceptación manejador de archivos.

Elaborado por: Estefani Herrera y Ricardo Figueroa.

5.3. Resultados pruebas de aceptación

El resultado se muestra en la Figura 56 encuesta realizada conjuntamente con los estudiantes no videntes dio como resultado un total de tres que en la escala de puntaje es buena.

Resultados

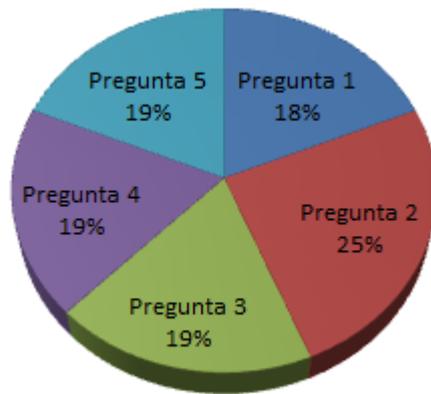


Figura 56. Resultados prueba con los usuarios.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

Prueba con los estudiantes no videntes



Figura 57. Prueba con los estudiantes no videntes.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

Prueba con los estudiantes no videntes



Figura 58. Prueba con los estudiantes no videntes.
Elaborado por: Estefani Herrera y Ricardo Figueroa.

CONCLUSIONES

- Las plataformas virtuales tradicionales son poco amigables con las personas que poseen discapacidad visual, carecen de herramienta de apoyo para interactuar con ellas, necesariamente requieren de un tercero como ayuda para utilizarla. Esta situación aparte de no ser óptima frustra a la persona porque siente que no es independiente, con este proyecto se logró construir un prototipo de ambiente virtual cuya funcionalidad de ingreso tiene la capacidad de utilizar la huella digital como opción biométrica para el logeo. Además se ha implementado en gran parte de los módulos de la aplicación la lectura y el reconocimiento de voz de esta manera apoyamos a la interacción del usuario con módulos desarrollados.
- La aplicación contiene un módulo manejador de archivos que permite a los estudiantes con discapacidad visual, subir archivos al servidor y descargar el contenido del repositorio.
- Al implementar herramientas de apoyo en la comunicación se pudo facilitar el envío de mensajes utilizando reconocimiento de voz para convertir el audio en texto y de esta manera permitir interactuar docente-estudiante y entre estudiantes.
- Al emplear la metodología SCRUM estamos utilizando un método ágil que nos ayuda a dividir la complejidad del proyecto en un sin número de pasos por lo que no es necesario esperar hasta finalizar el proyecto para tener un entregable si no que cada interacción o paso en si ya es un entregable.
- El desarrollar una solución que ayude a estudiantes con discapacidad visual es un tema muy complejo ya que no podemos sentir, imaginar con exactitud las necesidades de cada uno.

RECOMENDACIONES

- Se recomienda que para una versión posterior de la aplicación se realice un estudio más profundo de la problemática que implica la interacción de los estudiantes con discapacidad visual con respecto a las herramientas que utilizan en el proceso de aprendizaje.
- Es recomendable utilizar el navegador web Google Chrome porque integra Speech Recognition en el mismo y tiene un alto nivel de madurez.
- Es aconsejable leer el manual de usuario antes de la utilización del aplicativo, esto permitirá que el usuario se sienta seguro en su uso.
- Luego de las pruebas con estudiantes no videntes se sugiere usar comandos de teclado como método principal de interacción con las páginas web, dejando el mouse como periférico de acceso secundario.

GLOSARIO DE TÉRMINOS

UPS: Universidad Politécnica Salesiana.

AVAC: Plataforma virtual de la Universidad Politécnica Salesiana.

AVIDA: Ambiente virtual de accesibilidad.

Ceguera Total: Pérdida total de visión.

Discapacidad Visual: Término que engloba ceguera total.

SSH: Secure Shell, órdenes seguro.

SCP: Standard Control Procedure y pertenece a la categoría Base de Datos.

SFTP: Protocolo seguro para transferencia de archivos.

API: Application Programming Interface.

GUI: Graphical User Interface.

SDK: Software Development Kit.

USB: Universal Serial Bus, Bus Universal en Serie.

Lector de Pantalla: Es una herramienta que está diseñada para leer o reproducir el contenido que está en una página web.

Ceguera Nocturna: Personas con baja visión en ambientes con poca iluminación.

Tiflo: Término griego que hace referencia al entorno de las personas no videntes.

REFERENCIAS BIBLIOGRÁFICAS

- Incluyeme.com. (2013-2019). ¿Qué es la accesibilidad universal? Obtenido de <https://www.incluyeme.com/que-es-la-accesibilidad-universal-2/>
- Consejo Nacional para la Igualdad de Discapacidades. (02 de 06 de 2019). Estadísticas de Discapacidad. Recuperado el 09 de 07 de 2019, de Estadísticas de Discapacidad: <https://www.consejodiscapacidades.gob.ec/estadisticas-de-discapacidad/>
- Universitat de Barcelona. (2019). Las 5 etapas en los “Sprints” de un desarrollo Scrum. Recuperado el 12 de 07 de 2019, de <https://www.obs-edu.com/int/blog-investigacion/project-management/las-5-etapas-en-los-sprints-de-un-desarrollo-scrum>
- Universidad de Alicante. (2019). Accesibilidad Web. Recuperado el 12 de 07 de 2019, de <http://accesibilidadweb.dlsi.ua.es/?menu=quees>
- NetBeans. (s.f). Información NetBeans . Recuperado el 12 de 07 de 2019, de https://netbeans.org/community/releases/61/index_es.html
- Java. (s.f). ¿Qué es la tecnología Java y para qué la necesito? . Recuperado el 12 de 07 de 2019, de https://www.java.com/es/download/faq/whatis_java.xml
- MDN web docs. (2019). ¿Qué es JavaScript? Recuperado el 12 de 07 de 2019, de https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/Qu%C3%A9_es_JavaScript
- Luis Goncalves. (25 de 01 de 2019). ¿Qué es la metodología Scrum, todo lo que necesitas saber? Recuperado el 12 de 07 de 2019, de <https://luis-goncalves.com/es/que-es-la-metodologia-scrum/>
- Victor Ramos. (s.f). EL JDK . Recuperado el 12 de 07 de 2019, de <http://www.usmp.edu.pe/publicaciones/boletin/fia/info49/too.html>
- IBM Knoeledge Center. (s.f). Tecnología JSP . Recuperado el 12 de 07 de 2019, de https://www.ibm.com/support/knowledgecenter/es/SS5JSH_9.1.1/org.eclipse.wst.webtools.doc.user/topics/cpdjsps.html
- Laura Chuburu. (2018). ¿Qué es JQuery y cómo implementarlo? Recuperado el 12 de 07 de 2019, de <https://www.laurachuburu.com.ar/tutoriales/que-es-jquery-y-como-implementarlo.php>
- David Miller. (s.f). Start Bootstrap. Recuperado el 12 de 07 de 2019, de <https://startbootstrap.com/>
- Culturacion. (s.f). ¿Qué es Apache? Recuperado el 12 de 07 de 2019, de <http://culturacion.com/que-es-apache/>
- OpenSSH. (18 de 04 de 2019). OpenSSH. Recuperado el 12 de 07 de 2019, de <https://www.openssh.com/>
- TechTarget. (2019). SQL. Recuperado el 12 de 07 de 2019, de <https://searchsqlserver.techtarget.com/definicion/SQL>
- MariaDB. (2019). SQLyog: Community Edition. Recuperado el 12 de 07 de 2019, de <https://mariadb.com/kb/en/library/sqlyog-community-edition/>
- Margaret Rouse. (2015). MySQL. Recuperado el 12 de 07 de 2019, de <https://searchdatacenter.techtarget.com/es/definicion/MySQL>
- UBUNLOG. (s.f). MySQL Workbench, herramienta visual para el diseño de bases de datos. Recuperado el 12 de 07 de 2019, de <https://ubunlog.com/mysql-workbench-bases-datos/>
- Kai Wedekind. (26 de 07 de 2018). API de reconocimiento de voz. Recuperado el 12 de 07 de 2019, de <https://codeburst.io/html5-speech-recognition-api-670846a50e92?gi=b75e75e472ad>

- SIASA. (2011). Tecnología de Seguridad. Recuperado el 12 de 07 de 2019, de <https://www.siasa.com/producto.php?prod=0100041>
- JARC BLOG. (14 de 11 de 2011). Programacion de 4500 Fingerprint Reader Digital Persona en Java con Windows 7. Recuperado el 12 de 07 de 2019, de <http://blogjarc.blogspot.com/2011/12/programacion-de-4500-fingerprint-reader.html?m=1>
- Foros del web. (29 de 03 de 2013). ¿Cómo podre hacer un chat en jsp? Recuperado el 11 de 04 de 2018, de <http://www.forosdelweb.com/f45/como-podre-hacer-chat-jsp-1044355/>

ANEXOS

- **Anexo 1. Diccionario de datos:** Muestra el detalle de las tablas que conforman la base de datos de la aplicación.
- **Anexo 2. Instalación OpenSSH:** Describe los pasos necesarios para la instalación del servidor OpenSSH.
- **Anexo 3. Manual de usuario:** Guía que describe como el usuario debe interactuar con la aplicación.
- **Anexo 4. Arquetipo de la aplicación:** Muestra las ilustraciones del bosquejo de los módulos que componen la aplicación.
- **Anexo 5. Resultado pruebas de aceptación:** Muestra el resultado obtenido al ejecutar la prueba con el usuario final.