



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E
INDUSTRIAL**

**CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES E
INFORMÁTICOS**

Tema:

**DESARROLLO DE UN REPOSITORIO DIGITAL PARA EL MANEJO DE
DOCUMENTACIÓN DE UNIDADES FISCALES CON UNA APLICACIÓN WEB
PROGRESIVA EN LA UNIDAD DISTRITAL DE TECNOLOGÍAS DE
INFORMACIÓN Y COMUNICACIONES DEL DISTRITO 18D01-AMBATO 1
EDUCACIÓN.**

Trabajo de Titulación Modalidad: Proyecto de Investigación, presentado
previo a la obtención del título de Ingeniero en Sistemas Computacionales e
Informáticos.

ÁREA: Software

LÍNEA DE INVESTIGACIÓN: Desarrollo de Software

AUTOR: Jonathan Fernando Tisalema Lascano

TUTOR: Ing. Buenaño Valencia Edwin Hernando, Mg.

Ambato – Ecuador

marzo – 2022

APROBACIÓN DEL TUTOR

En calidad de tutor del Trabajo de Titulación con el tema: DESARROLLO DE UN REPOSITORIO DIGITAL PARA EL MANEJO DE DOCUMENTACIÓN DE UNIDADES FISCALES CON UNA APLICACIÓN WEB PROGRESIVA EN LA UNIDAD DISTRITAL DE TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIONES DEL DISTRITO 18D01-AMBATO 1 EDUCACIÓN, desarrollado bajo la modalidad Proyecto de Investigación por el señor Jonathan Fernando Tisalema Lascano, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 15 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y el numeral 7.4 del respectivo instructivo.

Ambato, marzo 2022

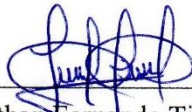
Ing. Buenaño Valencia Edwin Hernando, Mg.

TUTOR

AUTORIA

El presente Proyecto de Investigación titulado: DESARROLLO DE UN REPOSITORIO DIGITAL PARA EL MANEJO DE DOCUMENTACIÓN DE UNIDADES FISCALES CON UNA APLICACIÓN WEB PROGRESIVA EN LA UNIDAD DISTRITAL DE TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIONES DEL DISTRITO 18D01-AMBATO 1 EDUCACIÓN es absolutamente original, autentico y personal. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, marzo 2022.



Jonathan Fernando Tisalema Lascano

CC:1804393724

AUTOR

APROBACIÓN TRIBUNAL DE GRADO

En calidad de par calificador del Informe Final del Trabajo de Titulación presentado por el señor Jonathan Fernando Tisalema Lascano, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado DESARROLLO DE UN REPOSITORIO DIGITAL PARA EL MANEJO DE DOCUMENTACIÓN DE UNIDADES FISCALES CON UNA APLICACIÓN WEB PROGRESIVA EN LA UNIDAD DISTRITAL DE TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIONES DEL DISTRITO 18D01-AMBATO 1 EDUCACIÓN, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 17 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y al numeral 7.6 del respectivo instructivo. Para cuya constancia suscribimos, conjuntamente con la señora Presidenta del Tribunal.

Ambato, marzo 2022

Ing. Pilar Urrutia, Mg.

PRESIDENTA DEL TRIBUNAL

Ing. Carlos Núñez, Mg

PROFESOR CALIFICADOR

Ing. Sandra Carrillo R., Mg

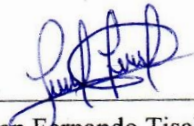
PROFESOR CALIFICADOR

DERECHOS DE AUTOR

Autorizo la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Titulación en favor de la Universidad Técnica de Ambato, con fines de difusión pública. Además, autorizo su reproducción total y parcial dentro de las regulaciones de la institución.

Ambato, marzo 2022.



Jonathan Fernando Tisalema Lascano

CC:1804393724

AUTOR

DEDICATORIA

Doy gracias en primer lugar a Dios por permitirme seguir viviendo y llenando mi vida de bendiciones a mí y a mi familia.

Dedicar este nuevo paso en mi vida a mis padres Fernando Tisalema y Wilma Lascano ya que han sido los pilares fundamentales en mi formación tanto personal como académico, guiándome y apoyándome a través de mi vida como estudiante.

A mi hermana Celeste Tisalema que en los tiempos de incertidumbre se ha encontrado ahí para apoyarme y hacerme compañía. A mis abuelitos que desde pequeño me mostraron su cariño y son como unos padres para mí.

De igual manera a mis tíos que me apoyaban en los momentos que más los necesitaba e inculcaban valores para llegar a ser una buena persona para la sociedad.

AGRADECIMIENTO

Agradecer a dios por la salud y la vida que ha vertido sobre mí y mi familia.

Agradecer totalmente al Ing. Buenaño Valencia Edwin Hernando, por todos los conocimientos impartidos dentro y fuera de clases, además por guiarme en el desarrollo de mi proyecto.

A la Universidad Técnica de Ambato, a la Facultad, ya que se convirtió en un segundo hogar que junto a mis compañeros pasamos momentos alegres, tristes y difíciles pero que llevaron a convertirnos en las personas que a día de hoy somos.

A la Unidad Distrital de Tecnologías de Información y Comunicación del distrito 18D01 de Ambato, por brindarme sus conocimientos y facilidades para poder realizar mi proyecto.

INDICE GENERAL DE CONTENIDOS

APROBACIÓN DEL TUTOR	ii
AUTORIA	iii
APROBACIÓN TRIBUNAL DE GRADO.....	iv
DERECHOS DE AUTOR	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
RESUMEN EJECUTIVO.....	xvii
ABSTRACT.....	xviii
INTRODUCCIÓN.....	xix
CAPITULO I: MARCO TEÓRICO	1
1.1. Tema de Investigación	1
1.2. Antecedentes Investigativos.....	1
1.3. Contextualización del problema.....	3
1.4. Delimitación.....	4
1.4.1. Contenidos.....	4
1.4.2. Espacial.....	4
1.4.3. Temporal.....	4
1.4.4. Justificación.....	5
1.5. Fundamentación Teórica.....	6
1.5.1. Sistemas de Información.....	6
1.5.2. Desarrollo de Software.....	6
1.5.3. Sistemas de Archivos	7
1.5.4. Aplicaciones Web Progresivas (PWA)	7
1.5.5. Framework de Desarrollo.....	8
1.5.6. JavaScript.....	9
1.5.7. Angular.....	9
1.5.8. Ionic.....	9
1.5.9. PHP (Hypertext Preprocessor)	10
1.5.10. Laravel.....	11
1.5.11. MySQL.....	12
1.5.12. Repositorios Virtuales	12
1.5.13. Metodologías Tradicionales	13

1.5.14. Metodologías Agiles	14
1.5.15. Comparativa Metodologías Tradicionales y Metodologías Agiles	15
1.6. Objetivos	16
1.6.1. Objetivo General	16
1.6.2. Objetivos Específicos	17
CAPITULO II: METODOLOGÍA	17
2.1. Materiales	18
2.2 Métodos	21
2.2.1 Modalidad básica de la investigación	21
2.2.2 Población y Muestra	22
2.2.3. Recolección de información	23
2.2.4. Procesamiento y análisis de datos.	33
CAPITULO III: RESULTADOS Y DISCUSIÓN	34
3.1. Análisis general de resultados	34
3.1.1. Métodos de recolección de información	34
3.1.2. Procedimiento para la solicitud de informatizada a las unidades educativas.35	
3.2.2. Métodos de almacenamiento de información	36
3.2.2.1 Comparativa métodos de almacenamiento de información	39
3.2.3. Herramienta para el desarrollo de la aplicación web progresiva Front-end. 39	
3.2.4. Herramienta para el desarrollo de la aplicación Back-End	41
3.2.4. Herramientas para el desarrollo de la aplicación web progresiva Base de Datos.	42
3.2.4. Tecnologías para el desarrollo de la aplicación web progresiva.....	44
3.2.4.1. Base de Datos	45
3.2.4.2. Desarrollo de Servicios Web	45
3.2.4.3. Ionic	46
3.2.5. Metodología para el desarrollo de la aplicación web progresiva	47
3.2.6. Desarrollo de la metodología	49
3.2.6.1. Planificación	49
3.2.6.2. Definición de roles	49
3.2.6.3. Levantamiento de la Información	50
3.2.6.4. Historias de Usuario	51
3.2.6.5. Tareas	61
3.2.6.6. Evaluación de las historias de usuario	77
3.2.6.7. Plan de Entrega	79

3.2.6.8. Plan de iteraciones.....	80
3.2.6.9. Diseño	81
3.2.6.9.1. Metáfora	81
3.2.6.9.2. Tarjetas CRC	82
3.2.6.9.3. Diseño de Interfaces	93
3.2.6.10. Codificación	107
3.2.6.10.1. Codificación Back-End	107
3.2.6.10.2. Codificación Front-End.....	131
3.2.6.11. Pruebas	136
CAPITULO IV: CONCLUSIONES Y RECOMENDACIONES	145
4.1. Conclusiones	145
4.2. Recomendaciones.....	146
Bibliografía	148
ANEXOS Y APÉNDICES	151
ANEXO A	151
ANEXO B	152
B.1. Manual de Usuario	152
B.2. Ingreso a la aplicación	152
B.3. Menú y pantalla Principal.....	154
B.4. Unidades Educativas	155
B.5. Buscar Archivos	167
B.6. Gestión de usuarios.....	169
B.7. Reportes.....	172
B.8. Noticias.....	173
B.9. Salir del Sistema.....	175

ÍNDICE DE TABLAS

Tabla 1. Ventajas y Desventajas de una PWA.....	7
Tabla 2. Tipos de Repositorios Virtuales.....	13
Tabla 3. Características Principales de XP	14
Tabla 4. Comparativa de Metodologías	16
Tabla 5. Matriz de Observación.....	21
Tabla 6. Población	22
Tabla 7. Matriz de Observación.....	30
Tabla 8. Recolección de Información-Matriz de Observación	31
Tabla 9. Métodos o procesos de Recolección de Información	34
Tabla 10. Comparativa métodos de almacenamiento	39
Tabla 11. Comparativa framework de desarrollo	40
Tabla 12. Comparativa herramientas Back-End	42
Tabla 13. Comparativa herramientas Base de Datos	42
Tabla 14. Tecnologías para el desarrollo de la aplicación web progresiva	44
Tabla 15. Comparativa Metodologías.....	47
Tabla 16. Características Metodología XP	48
Tabla 17. Definición de Roles	50
Tabla 18. Modelo Historia de usuario	51
Tabla 19. Descripción del modelo de historia de usuario.....	52
Tabla 20. Historia de usuario 01-Roles de Usuario	52
Tabla 21. Historia de usuario 02-Diseño Base de Datos	53
Tabla 22. Historia de usuario 03-Arquitectura del proyecto	53
Tabla 23. Historia de usuario 04-Pantalla de presentación para administradores distritales.....	54
Tabla 24. Historia de usuario 05-Pantalla menú unidad educativa.....	54
Tabla 25. Historia de usuario 06-Pantalla para periodos/años lectivos	55
Tabla 26. Historia de usuario 07-Pantalla niveles/cursos	55
Tabla 27. Historia de usuario 08-Pantalla paralelos	56
Tabla 28. Historia de usuario 09-Pantalla contenido paralelo.....	56
Tabla 29. Historia de usuario 10-Pantalla documentación institucional.....	57
Tabla 30. Historia de usuario 11-Pantalla archivos documentación institucional.....	57
Tabla 31. Historia de usuario 12-Pantalla presentación administradores institucionales	58
Tabla 32. Historia de usuario 13-Buscador de Archivos	58
Tabla 33. Historia de usuario 14-Gestión de Usuarios	59
Tabla 34. Historia de usuario 15-Reportes	59
Tabla 35. Historia de usuario 16-Noticias/Notificaciones.....	60
Tabla 36. Historia de usuario 17-Pantalla inicio.....	60
Tabla 37. Historia de usuario 18-Acceso a la aplicación.....	61
Tabla 38. Tarea 1- Historia 1	61
Tabla 39. Tarea 1- Historia 2	62
Tabla 40. Tarea 2- Historia 2	62
Tabla 41. Tarea 1- Historia 3	62

Tabla 42. Tarea 1- Historia 4.....	63
Tabla 43. Tarea 1- Historia 5.....	63
Tabla 44. Tarea 1- Historia 6.....	64
Tabla 45. Tarea 2- Historia 6.....	64
Tabla 46. Tarea 1- Historia 7.....	65
Tabla 47. Tarea 2- Historia 7.....	65
Tabla 48. Tarea 1- Historia 8.....	66
Tabla 49. Tarea 2- Historia 8.....	66
Tabla 50. Tarea 1- Historia 9.....	67
Tabla 51. Tarea 2- Historia 9.....	67
Tabla 52. Tarea 3- Historia 9.....	68
Tabla 53. Tarea 1- Historia 10.....	68
Tabla 54. Tarea 2- Historia 10.....	69
Tabla 55. Tarea 1- Historia 11.....	69
Tabla 56. Tarea 2- Historia 11.....	70
Tabla 57. Tarea 3- Historia 11.....	70
Tabla 58. Tarea 1- Historia 12.....	71
Tabla 59. Tarea 1- Historia 13.....	71
Tabla 60. Tarea 2- Historia 13.....	72
Tabla 61. Tarea 1- Historia 14.....	72
Tabla 62. Tarea 2- Historia 14.....	73
Tabla 63. Tarea 1- Historia 15.....	73
Tabla 64. Tarea 2- Historia 15.....	74
Tabla 65. Tarea 1- Historia 16.....	74
Tabla 66. Tarea 2- Historia 16.....	75
Tabla 67. Tarea 1- Historia 17.....	75
Tabla 68. Tarea 1- Historia 18.....	76
Tabla 69. Tarea 2- Historia 18.....	76
Tabla 70. Tarea 3- Historia 18.....	77
Tabla 71. Evaluación Historias de usuario.....	78
Tabla 72. Plan de Entrega.....	79
Tabla 73. Plan de Iteraciones.....	80
Tabla 74. Tarjeta CRC- Ingreso nueva Unidad Educativa.....	82
Tabla 75. Tarjeta CRC- Presentación Unidades Educativas.....	83
Tabla 76. Tarjeta CRC- Gestión de Unidades Educativas.....	83
Tabla 77. Tarjeta CRC- Ingreso nuevo Periodo Lectivo.....	84
Tabla 78. Tarjeta CRC- Presentación Periodos Lectivos.....	84
Tabla 79. Tarjeta CRC- Gestión de periodos lectivos.....	85
Tabla 80. Tarjeta CRC- Ingreso nueva categoría Documentos Institucionales.....	85
Tabla 81. Tarjeta CRC- Presentación Categorías de Documentación.....	85
Tabla 82. Tarjeta CRC- Gestión de categorías.....	86
Tabla 83. Tarjeta CRC- Ingreso nuevo Curso o Nivel.....	86
Tabla 84. Tarjeta CRC- Presentación Cursos o Niveles.....	86
Tabla 85. Tarjeta CRC- Gestión de Cursos o Niveles.....	87
Tabla 86. Tarjeta CRC- Ingreso nuevo Paralelo.....	87
Tabla 87. Tarjeta CRC- Presentación Paralelos.....	87

Tabla 88. Tarjeta CRC- Gestión de paralelos	88
Tabla 89. Tarjeta CRC- Ingreso Archivos Estudiantes.....	88
Tabla 90. Tarjeta CRC- Presentación Archivos.....	88
Tabla 91. Tarjeta CRC- Gestión de archivos	89
Tabla 92. Tarjeta CRC- Presentación Opciones	89
Tabla 93. Tarjeta CRC- Descarga y visualización de archivos	89
Tabla 94. Tarjeta CRC- Presentación Opciones	90
Tabla 95. Tarjeta CRC- Descarga y visualización de reporte.....	90
Tabla 96. Tarjeta CRC- Ingreso de Noticias/eventos	90
Tabla 97. Tarjeta CRC- Presentación Noticias/eventos.....	91
Tabla 98. Tarjeta CRC- Gestión de noticias	91
Tabla 99. Tarjeta CRC- Ingreso de Usuarios.....	91
Tabla 100. Tarjeta CRC- Presentación del listado de usuarios.....	92
Tabla 101. Tarjeta CRC- Gestión de usuarios	92
Tabla 102. Tarjeta CRC- Validación de Usuarios	92
Tabla 103. Tarjeta CRC- Enrutamiento de Roles	93
Tabla 104. Prueba de Aceptación Historia 1	137
Tabla 105. Prueba de Aceptación Historia 4	137
Tabla 106. Prueba de Aceptación Historia 5	138
Tabla 107. Prueba de Aceptación Historia 6	138
Tabla 108. Prueba de Aceptación Historia 7	139
Tabla 109. Prueba de Aceptación Historia 8	139
Tabla 110. Prueba de Aceptación Historia 9	140
Tabla 111. Prueba de Aceptación Historia 10	140
Tabla 112. Prueba de Aceptación Historia 11	141
Tabla 113. Prueba de Aceptación Historia 12	141
Tabla 114. Prueba de Aceptación Historia 13	142
Tabla 115. Prueba de Aceptación Historia 14	142
Tabla 116. Prueba de Aceptación Historia 15	143
Tabla 117. Prueba de Aceptación Historia 16	143
Tabla 118. Prueba de Aceptación Historia 17	144
Tabla 119. Prueba de Aceptación Historia 18	144

ÍNDICE DE ILUSTRACIONES

Ilustración 1 Herramientas Ionic.....	10
Ilustración 2 Modelo Vista Controlador	11
Ilustración 3 Fases RUP.....	14
Ilustración 4 Resultado Encuesta Primera parte Pregunta 1	23
Ilustración 5 Resultado Encuesta Primera parte Pregunta 2	24
Ilustración 6 Resultado Encuesta Primera parte Pregunta 3	24
Ilustración 7 Resultado Encuesta Primera parte Pregunta 4	25
Ilustración 8 Resultado Encuesta Primera parte Pregunta 5	26
Ilustración 9 Resultado Encuesta Segunda parte Pregunta 2	27
Ilustración 10 Resultado Encuesta Segunda parte Pregunta 3	27
Ilustración 11 Resultado Encuesta Segunda parte Pregunta 4	28
Ilustración 12 Resultado Encuesta Segunda parte Pregunta 1	28
Ilustración 13 Resultado Encuesta Segunda parte Pregunta 2	29
Ilustración 14 Resultado Encuesta Segunda parte Pregunta 3	30
Ilustración 15 Resultado Encuesta Segunda parte Pregunta 4	30
Ilustración 16 Procedimiento para la solicitud informatizada.	36
Ilustración 17 Almacenamiento en la nube.....	38
Ilustración 18 Estructura Pages Ionic	46
Ilustración 19 Fases XP	49
Ilustración 20 Módulos Definidos	51
Ilustración 21 Menú para Administradores distritales	94
Ilustración 22 Menú para administradores institucionales	94
Ilustración 23 Menú para Rectores	94
Ilustración 24 Pantalla inicio unidades educativas para administradores distritales .	95
Ilustración 25 Pantalla para el ingreso de nueva unidad educativa	95
Ilustración 26 Menú de la unidad educativa	97
Ilustración 27 Menú Periodos Lectivos	97
Ilustración 28 Ingreso nuevo periodo lectivo	97
Ilustración 29 Menú niveles.....	98
Ilustración 30 Ingreso nuevo nivel según la jornada seleccionada.....	98
Ilustración 31 Menú Paralelos.	99
Ilustración 32 Ingreso nuevo Paralelo	100
Ilustración 33 Menú contenido del paralelo.	100
Ilustración 34 Ingreso archivos al paralelo.	101
Ilustración 35 Menú Categorías documentos institucionales.	101
Ilustración 36 Ingreso Archivos institucionales.....	102
Ilustración 37 Buscar documentos estudiantes	102
Ilustración 38 Buscar documentos institucionales.....	103
Ilustración 39 Gestión de Usuarios.....	103
Ilustración 40 Agregar nuevo usuario.....	104
Ilustración 41 Reporte de archivos de estudiantes.....	104
Ilustración 42 Reporte documentos institucionales.	105
Ilustración 43 Gestión de Noticias.....	105

Ilustración 44 Agregar nueva Noticia.	105
Ilustración 45 Pantalla de Inicio.	106
Ilustración 46 Ingreso al Sistema.	106
Ilustración 47 Diagrama de Base de Datos.	107
Ilustración 48 Estructura Laravel.	108
Ilustración 49 Configuración MySQL-Laravel.	108
Ilustración 50 Conexión a la base de datos.	108
Ilustración 51 Modelo conexión tablas base de datos.	109
Ilustración 52 Modelos para la gestión de los datos.	109
Ilustración 53 Controladores.	110
Ilustración 54 Rutas para consulta API.	111
Ilustración 55 Rutas para consulta API.	111
Ilustración 56 Ingresar Unidad Educativa.	112
Ilustración 57 Eliminar Unidad Educativa.	112
Ilustración 58 Actualizar unidad Educativa.	112
Ilustración 59 Obtener listado de Unidades Educativas.	113
Ilustración 60 Obtener Unidad Educativa por Id.	114
Ilustración 61 Registrar usuario.	114
Ilustración 62 Actualizar Usuario.	115
Ilustración 63 Eliminar Usuario.	115
Ilustración 64 Obtener lista de usuarios y roles.	116
Ilustración 65 Login de usuario.	116
Ilustración 66 LogOut.	116
Ilustración 67 Ingresar nuevo Periodo Lectivo.	117
Ilustración 68 Eliminar Periodo Lectivo.	117
Ilustración 69 Actualizar periodo lectivo.	118
Ilustración 70 Obtener periodos lectivos por unidad educativa.	118
Ilustración 71 Ingresar Nueva Categoría.	119
Ilustración 72 Actualizar categoría.	119
Ilustración 73 Eliminar categoría.	120
Ilustración 74 Subida de Archivos a documentación.	120
Ilustración 75 Listar los archivos dentro de la categoría de documentación.	121
Ilustración 76 Obtener el Archivo.	121
Ilustración 77 Eliminar Archivo.	122
Ilustración 78 Ingresar nivel.	122
Ilustración 79 Eliminar Nivel.	123
Ilustración 80 Actualizar Nivel.	123
Ilustración 81 Obtener listado de niveles por periodo lectivo.	124
Ilustración 82 Ingresar Paralelo.	124
Ilustración 83 Eliminar Paralelo.	125
Ilustración 84 Actualizar Paralelo.	125
Ilustración 85 Subir archivos al paralelo.	126
Ilustración 86 Obtener los archivos del paralelo.	126
Ilustración 87 Obtener el archivo.	127
Ilustración 88 Eliminar Archivo.	127
Ilustración 89 Ingresar Noticia.	128

Ilustración 90 Obtener Noticias.	128
Ilustración 91 Eliminar Noticia.....	129
Ilustración 92 Reporte de documentos institucionales.	129
Ilustración 93 Reporte de archivos estudiantes por nivel.	129
Ilustración 94 Reporte archivos de estudiantes por paralelo.	130
Ilustración 95 Pages para el Front-End.....	130
Ilustración 96 Peticiones para Unidad Educativa.	131
Ilustración 97 Peticiones para periodos lectivos.....	131
Ilustración 98 Peticiones para niveles.....	131
Ilustración 99 Peticiones paralelos.	132
Ilustración 100 Peticiones para los archivos del paralelo.	132
Ilustración 101 Peticiones para la documentación institucional.	132
Ilustración 102 Peticiones para los archivos de la documentación.....	133
Ilustración 103 Peticiones para reportes.	133
Ilustración 104 Método para el ingreso de usuario.....	133
Ilustración 105 Método de navegación según el rol de usuario.....	134
Ilustración 106 Guards para el control de navegación.....	134
Ilustración 107 Rutas de Navegación.	134
Ilustración 108 Método para subir Archivos.	135
Ilustración 109 Método para descargar el Archivo.....	135
Ilustración 110 Método para Visualizar el Archivo.....	135
Ilustración 111 Método para eliminar un archivo.....	136
Ilustración 112 Métodos para el cifrado de información.....	136

RESUMEN EJECUTIVO

En la actualidad, por la emergencia sanitaria la tecnología web y móvil se ha convertido en la prioridad para muchos desarrolladores que buscan satisfacer todas las necesidades de los usuarios, pero con el excesivo tiempo que conlleva desarrollar una aplicación móvil y otra web surgen como respuesta las aplicaciones web progresivas que permiten el doble funcionamiento para dichas necesidades de una manera muy versátil ya que integra un solo código que puede ser interpretado tanto a nivel de ordenador como en dispositivos móviles.

El presente proyecto de investigación tiene como objetivo principal el desarrollo de un repositorio virtual con una aplicación web progresiva que permita manejar los documentos de estudiantes y documentación institucional de las unidades educativas fiscales de Ambato. Como apoyo al personal de la Unidad Distrital de Tecnologías de Información y Comunicaciones del Distrito 18D01-Ambato 1 Educación, ya que no cuentan con una herramienta para la gestión de documentación de todas las unidades educativas fiscales que se encuentran a su cargo. Además de las constantes consultas y peticiones de la documentación por parte de los rectores.

El desarrollo de la aplicación con la tecnología IONIC, permite reducir de gran manera el tiempo para la creación de un aplicativo que se pueda visualizar en un ordenador y en un dispositivo móvil. La metodología para el desarrollo del proyecto es XP (Xtreme Programming), metodología ágil que se enfoca en reglas y principios para mejorar las fases de pruebas y los tiempos de entrega. Además, el framework Laravel para el desarrollo del Back-End ya que integra seguridad en su modelo de arquitectura.

Palabras clave: Angular, ionic, repositorio, laravel, archivos.

ABSTRACT

At present, due to the health emergency, web and mobile technology has become the priority for many developers who are looking to meet all the needs of users, but with the excessive time involved in developing a mobile app and a web app, progressive web applications are emerging as a response that allow dual operation for these needs in a very versatile way, since it integrates a single code that can be interpreted both on a computer and on mobile devices. mobile.

The main objective of this research project is the development of a virtual repository with a progressive web application that allows the management of student documents and institutional documentation of the fiscal education units of Ambato. To support the staff of the District Information Technology and Communications Unit of District 18D01-Ambato 1 Education, since they do not have a tool for the management of documentation of all the tax education units under their charge. In addition to the constant consultations and requests for documentation by the rectors.

The development of the application with IONIC technology allows to greatly reduce the time needed to create an application that can be viewed on a computer and on a mobile device. The methodology for the development of the project is XP (Xtreme Programming), an agile methodology that focuses on rules and principles to improve testing phases and delivery times. In addition, the Laravel framework for the development of the Back-End as it integrates security into its architecture model.

Keywords: Angular, Ionic, repository, Laravel, files.

INTRODUCCIÓN

El proyecto que se presenta a continuación lleva el título: “DESARROLLO DE UN REPOSITORIO DIGITAL PARA EL MANEJO DE DOCUMENTACIÓN DE UNIDADES FISCALES CON UNA APLICACIÓN WEB PROGRESIVA EN LA UNIDAD DISTRITAL DE TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIONES DEL DISTRITO 18D01-AMBATO 1 EDUCACIÓN” se encuentra dividido en los siguientes capítulos:

CAPÍTULO I: “MARCO TEÓRICO”, se describe la situación problemática que presenta la Unidad Distrital de Tecnologías de Información y Comunicaciones del distrito, justificando las razones por las que se realiza la investigación para poder establecer y definir objetivos específicos que serán la base del desarrollo investigativo.

CAPÍTULO II: “METODOLOGÍA”, se define las técnicas y herramientas que se van a utilizar para el desarrollo del proyecto, se describe los métodos para la recolección de información, la modalidad para la investigación y las etapas que conlleven su desarrollo.

CAPÍTULO III: “RESULTADOS Y DISCUSIÓN”, se detalla de manera puntual y clara el desarrollo del proyecto de investigación, las metodologías existentes que se pueden utilizar y vez las herramientas necesarias para el correcto funcionamiento del proyecto.

CAPÍTULO IV: “CONCLUSIONES Y RECOMENDACIONES”, sección enfocada a las conclusiones a las que el investigador considero en el transcurso de la elaboración del proyecto además de las recomendaciones que el investigador puede realizar para tomar en cuenta en la elaboración de proyectos posteriores.

CAPITULO I: MARCO TEÓRICO

1.1. Tema de Investigación

Desarrollo de un repositorio digital para el manejo de documentación de unidades fiscales con una aplicación web progresiva en la Unidad Distrital de Tecnologías de Información y Comunicaciones del Distrito 18D01-Ambato 1 Educación.

1.2. Antecedentes Investigativos

Tomando en cuenta antecedentes investigativos y proyectos se puede mencionar: Según Barrueco, José Manuel y García Testal, Cristina [1] en su estudio “REPOSITARIOS INSTITUCIONALES UNIVERSITARIOS: EVOLUCIÓN Y PERSPECTIVAS”.

El uso de las tecnologías para el almacenamiento de información institucional es de gran ayuda tanto para los encargados de la administración de los datos como para los que la buscan. En su investigación realizada a los diferentes repositorios institucionales, mencionan principalmente sus características, topología, volumen y crecimiento al paso de los años. Se menciona además la facilidad de acceso a la información necesaria, como una forma de fomentar la investigación científica, según la forma que se comparte sus proyectos y evitar la creación de repositorios institucionales con información redundante o innecesaria.

Además, se puede mencionar a Barahona Do Santos, Fernando [2] en su proyecto de Tesis “Diseño e implementación de un portal Web y un repositorio virtual de contenido multimedia académico para el área de Ingeniería Electrónica y Redes”. La información manejada desde un portal web, con facilidad de uso, búsqueda y acceso a la documentación que sea necesaria, es muy eficaz y es considerada como la mejor alternativa al momento de implementar un sitio. Además de

contar con la creación y diseño web desde cero lo provee de funcionalidades ilimitadas y que pueden convertir a las aplicaciones web en progresivas.

Se toma en cuenta de igual manera a Jessica Leottau Andrade, Mónica Patricia Pinzon Vives y Jose Miguel Torres Scaff [3] en su proyecto de grado “Diseño e implementación de una Biblioteca Virtual basada en el buscador para Universidades”.

Se concluye que, los esquemas en los que la información, documentación o libros son almacenados de una forma virtual es muy necesaria ya que pueden estar disponibles en cualquier momento y las unidades educativas no gastan mucho dinero en documentación impresa, promoviendo de esta manera también la ayuda al medio ambiente evitando la tala de árboles y generación de CD's para el almacenamiento de la información.

Cabe mencionar a MSc. Alexander Ochoa Agüero, Dr. Dalier Pérez Águila y Lic. Yali Bles Portú [4] en su artículo “Repositorio Institucional de contenido educativo de la Escuela Latinoamericana de Medicina”. Se detalla que, la implementación de un repositorio está sujeto a varias limitaciones como son el lenguaje en el que está desarrollado, los requerimientos necesarios para su instalación, por lo cual optaron por mejores alternativas y el desarrollo propio ya que satisfacían de mejor manera los requerimientos de su institución.

Se menciona a Bertha Alejandra Paredes Calderón [5] en su proyecto de tesis “Diseño y Aplicación de una biblioteca virtual de tesis digitales de la escuela de ingeniería industrial”.

Concluye que, se debe realizar un estudio preliminar para poder definir las características tanto básicas como complejas que se puede presentar en el sistema, además poder contar

con los beneficiarios del sistema los cuáles pueden aportar con su conocimiento para poder definir de una mayor manera las características y funcionalidades finales a implementar.

Además, hay que considerar a Rafael Hurtado y Maby Hernández [6] en su proyecto “Diseño e implementación de la Biblioteca Virtual especializada sobre los derivados de la caña de azúcar”. El uso y manejo de nuevas tecnologías dedicadas al almacenamiento de información es de gran importancia y más aún si es almacenada de manera digital y es accesible desde cualquier lugar, ya que evita contratiempos como movilidad o papeleos para poder acceder a la información. Aumentando de forma considerable la productividad de los usuarios que manejen la información.

Por último, se toma en cuenta a Whitney Oswaldo Tapia Ochoa [7] en su proyecto de titulación “Repositorio virtual de la biblioteca Uniandes”.

Concluye que se debe llevar de una forma clara y responsable la información que se va a guardar en un repositorio ya que puede ser vista por un gran número de personas esto llevara a un impacto favorable a los usuarios y directivos. Además, que los encargados del manejo de la información deben ser personas responsables y experimentadas para garantizar la legitimidad de la información y su totalidad.

1.3. Contextualización del problema

En la actualidad gran parte de las unidades educativas tanto fiscales como privadas cuentan con sistemas para facilitar el control de procesos destinados al almacenamiento y manejo de su información, pero se debe tomar en cuenta los niveles superiores, donde se encuentran las unidades distritales que deben contar con toda la información que se genera en los sistemas de las unidades educativas. Para ello el uso de las tecnologías de información y comunicación mediante aplicaciones que permitan el acceso a los servicios que brindan las unidades distritales deben estar siempre disponibles para que se pueda tener una visión clara y transparente del trabajo de la administración pública [8].

Por lo que se opta por el manejo de repositorios de datos a los que tanto directivos de unidades educativas, como personal distrital pueden acceder de una manera fácil y rápida debido a que están destinados al manejo de información concreta [9]. En el país se maneja un sistema de registros tanto de calificaciones como documentación a nivel general de las instituciones educativas para la educación virtual [10] “Carmenta”, pero como es un sistema que se utiliza en todo el país, su tiempo en línea puede variar a lo largo del día, y muchas veces llega a estar desconectado por varias horas. Por lo que, si en algún momento se necesita alguna información académica, se debe esperar a que el sistema pueda estar disponible nuevamente. A su vez esta información es manejada de forma distrital por diferentes centros de administración pública de la educación en cada provincia del país.

En el distrito 18D01-Ambato 1 Educación manejan toda la información que proviene de más de 40 unidades educativas en Tungurahua, por lo que el manejo de la información debe ser rápida y segura. Pero en la unidad de tecnologías de información encargada de la documentación no cuenta con el personal y equipos necesarios. Por lo que el almacenamiento de la documentación es simple y está expuesta a pérdidas, ya que no se cuenta con sistema de almacenamiento o búsqueda de información.

1.4. Delimitación

1.4.1. Contenidos

Línea de Investigación: Tecnología de la Información y Sistemas de Control.

1.4.2. Espacial

Unidad Distrital de Tecnologías de Información y Comunicaciones del Distrito 18D01-Ambato 1 Educación.

1.4.3. Temporal

La presente investigación se desarrolló en el periodo comprendido entre octubre del 2021 y marzo del 2022.

1.4.4. Justificación

El uso de las aplicaciones web progresivas en la actualidad tiene una gran acogida, tanto en desarrolladores como en usuarios finales ya que este tipo de aplicaciones permite su ejecución en cualquier tipo de dispositivo ya sea de escritorio o móvil, debido a sus importantes características tecnológicas que brindan una buena experiencia de uso en base a sus distintas funcionalidades.

Con el manejo de un repositorio web para el almacenamiento de la documentación de las unidades educativas fiscales se pretende dar a conocer en primer lugar las nuevas tecnologías para la creación de aplicaciones como lo son las aplicaciones web progresivas que permiten de una forma rápida y segura poder ser utilizadas en cualquier tipo de dispositivo como aplicaciones nativas, de esta forma brindar a los directores de las unidades educativas el acceso a la documentación que requieran en cualquier lugar, de igual manera a los empleados de la unidad permitir designar encargados en las unidades educativas, la tarea de subir la información pertinente al repositorio, para tener un control y seguimiento de la información generada en los diferentes periodos académicos.

Es factible y novedoso ya que permite tener la información disponible en cualquier momento que se requiera, además de evitar la pérdida de información al contar con un orden de almacenamiento, evitando los métodos de almacenamiento tradicionales que se maneja en la unidad distrital. Con la información presentada se puede determinar que, el proyecto es de gran importancia y utilidad para la Unidad Distrital de Tecnologías de Información y Comunicaciones del Distrito 18D01-Ambato 1 Educación, además que se cuenta con las fuentes de información pertinentes y recursos para el cumplimiento de los objetivos a desarrollar.

Factibilidad Técnica. – El investigador cuenta los conocimientos necesarios para el desarrollo de la aplicación a su vez los requerimientos y procesos necesarios para su elaboración.

Factibilidad Operativa. – Se cuenta con el apoyo del personal de la unidad de trabajo a su vez con los departamentos que manejan la información, y el equipo proporcionado por la unidad de tecnologías y comunicaciones del distrito.

Factibilidad Económica. - es económicamente factible ya que el tiempo, análisis, investigación y desarrollo son asumidos por el investigador.

1.5.Fundamentación Teórica

1.5.1. Sistemas de Información

Los sistemas de información se definen como un conjunto de procesos formales que operan sobre colecciones de datos estructurados, de tal manera que ayudan a cubrir necesidades en una empresa la cual recopila y distribuyen su información para poder operar con normalidad. Estos sistemas muchas veces cuentan con procesos y actividades para la toma de decisiones al momento de desempeñar las funciones de negocio que van acorde a sus estrategias [11]. En los últimos años tanto el entorno de desarrollo como el mercado para las tecnologías de información, se han incrementado, dando a conocer a profundidad la reducción del ciclo de vida de los productos que se desarrollan año tras año, avistando de manera clara las necesidades de las empresas de la implementación y manejo de nuevas estrategias de información, que pueda garantizar su supervivencia y crecimiento empresarial [12].

1.5.2. Desarrollo de Software

El desarrollo de software es una serie de pasos estructurados utilizados para la creación o elaboración de un sistema para cumplir o solucionar un problema o necesidad. En estos procesos a su vez intervienen diferentes entidades como son un cliente o usuario, un analista que se encarga de proveer los requerimientos a los desarrolladores, los cuales son los encargados de diseñar y codificar el sistema. El resultado final del desarrollo de software es dar un documento que de forma entendible explique el proyecto realizado a los usuarios que lo vayan a implementar y utilizar [13].

1.5.3. Sistemas de Archivos

Los sistemas de archivos también conocidos como sistemas de almacenamiento son procesos que permiten la organización, búsqueda, almacenamiento, eliminación de datos o archivos de manera concreta y específica. El objetivo principal de estos sistemas es brindar a un usuario un lugar donde pueda acceder a esta información sin lugar a error de manera eficaz y lo más rápido posible. En la actualidad los sistemas de archivos implementan un modelo digital en su totalidad ya que no cuentan con una ubicación física o material de su alojamiento, ya que se busca disminuir el gasto de recursos materiales. Pero a su vez se debe contar con encargados para la gestión del ciclo de vida de los documentos, además de contribuir con la recuperación y consulta de información necesaria [14].

1.5.4. Aplicaciones Web Progresivas (PWA)

Una PWA es una aplicación web que maneja las tecnologías más actuales de los navegadores, mediante las cuales ofrecen a los dispositivos móviles una interacción similar a una aplicación nativa, la cual fomenta los tiempos de respuesta, rendimiento o que la interfaz no contenga distorsión o fallos. Las aplicaciones web progresivas ofrecen una evolución natural a las aplicaciones web ya que cuentan con un enfoque integral con aspectos de software, para que a su vez la tecnología de los dispositivos móviles no acorte su duración de vida ante la acelerada producción de estos [15]. Cabe tomar en cuenta el gran impacto de la tecnología móvil en la actualidad, los dispositivos móviles ocupan un papel importante en la vida de las personas los cuales combinados con el acceso a internet conforman la principal manera de acceder a la información. La mayoría de los usuarios buscan en la red la información que sea de su utilidad, pero gran parte de la información aún se presenta en el modelo tradicional por lo que en un dispositivo móvil presenta fallos y distorsión, conllevando a la creación de aplicaciones nativas exclusivas para un sistema operativo, multiplicando el esfuerzo de desarrollo y mantenimiento [16].

Tabla 1. Ventajas y Desventajas de una PWA

Ventajas	Desventajas
El costo y tiempo de desarrollo es menor ya que se enfoca en la elaboración de un solo proyecto.	Depende de la conexión a internet
La velocidad para el acceso y manejo este sujeto solo a la velocidad de internet del usuario	Necesita compatibilidad con el navegador web
Los requerimientos del sistema como la memoria son menores	Experiencia de usuario debido a la interfaz
Funcionalidad completa en dispositivos Android	Funcionalidades reducidas en iOS

Elaborado por: El Investigador.

1.5.5. Framework de Desarrollo

Se conoce como framework de desarrollo a una colección de librerías, bibliotecas, normas y herramientas que los desarrolladores de aplicaciones utilizan para crear diferentes tipos de aplicación según las necesidades de un usuario final. Un framework trabaja a su vez con una serie de segmentos o componentes que se relacionan entre sí para generar diferentes funcionalidades. Un framework ayuda de gran medida a los desarrolladores ya que permite la generación de un proyecto el cual se lo puede modificar en cualquier momento. Una característica esencial de los frameworks es que permite la reutilización del código, implementando un modelo incremental ya que de esta manera promueve la evolución e las aplicaciones y mejorando su ciclo de vida. La gran parte de los framework de desarrollo son de código abierto, ya que cuentan con funcionalidades y características que cualquier persona puede acceder, además que tienen grandes comunidades de desarrollo promoviendo la reutilización del código, de igual manera manejan estructuras y modelos para la creación de los proyectos permitiendo de esta manera una creación optima de las aplicaciones [17].

1.5.6. JavaScript

JavaScript es el lenguaje de programación o a su vez un lenguaje de comandos y secuencias que permiten el manejo de funciones para la creación de páginas web dinámicas, permitiendo la integración de tablas, efectos de texto, animaciones y demás acciones al momento de ingresar alguna información o manejar botones. Se conoce también como lenguaje interpretado ya que no necesita que el programa compile para que su código pueda ejecutarse, su código puede ejecutarse directamente en el navegador y ver su funcionamiento [18].

Las características más esenciales de JavaScript son su facilidad de uso ya que fácilmente se lo puede crear y manejar en un editor de texto. Es bastante potente ya que puede permitir el manejo de objetos.

1.5.7. Angular

Angular es un framework de código abierto basado en JavaScript para el desarrollo de aplicaciones web, es uno de los frameworks más utilizados por los desarrolladores ya que se adapta a la mayoría de necesidades que se pretende cubrir, además de contar con un sin número de interacciones y funcionalidades con otros frameworks como Bootstrap o Ionic ya que funciona como código base para generar aplicaciones más atractivas o a su vez funcionar como aplicaciones para dispositivos móviles y en otros casos como una aplicación web progresiva (PWA) [19].

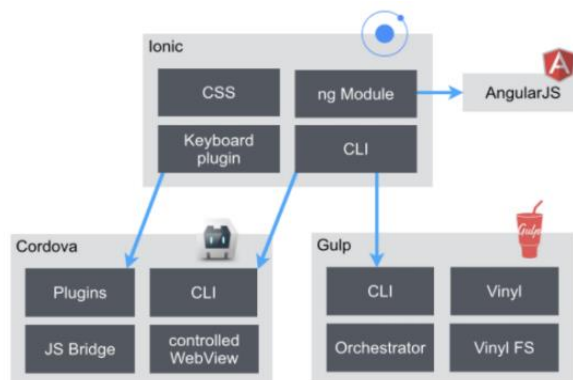
1.5.8. Ionic

Ionic es un framework de desarrollo de aplicaciones gratuito y de código abierto, mayormente usado para aplicaciones móviles basadas en JavaScript, HTML y CSS. Desde su creación los desarrolladores solo lo utilizaban para crear aplicaciones nativas, pero en la actualidad es utilizado para el desarrollo tanto de aplicaciones móviles como para aplicaciones web. Debido a su adaptabilidad también pueden llegar a comprender el desarrollo de aplicaciones web progresivas, con su facilidad de implementación y utilización ya que los desarrolladores solo se enfocan en un proyecto [20]. Ionic cuenta con 4 partes principales en su estructura:

Las hojas de estilo en el cual se modela y diseña la apariencia tanto para dispositivos móviles o web.

- El módulo principal de AngularJS que maneja las directivas a implementar y las rutas de navegación.
- Las herramientas Córdova o Capacitor y Gulp que permiten gestionar la línea de comandos para la comunicación con las diferentes librerías.
- El plugin del keyboard el cual ayuda a saber el estado e información de nuestra aplicación.

Ilustración 1 Herramientas Ionic



Elaborado por: Fuente [20]

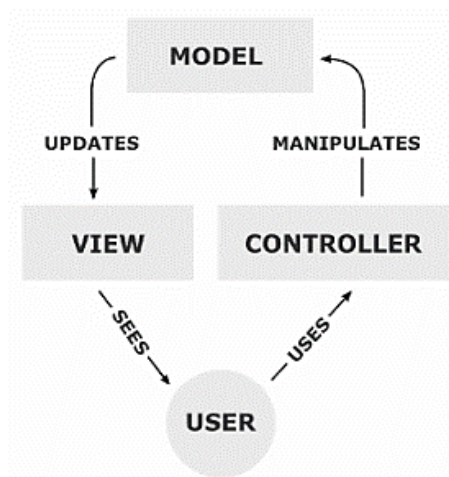
1.5.9. PHP (Hypertext Preprocessor)

PHP o también conocido por su nombre original Personal Home Page es un lenguaje interpretado de código abierto, originalmente fue usado para el desarrollo de aplicaciones que actúen en el lado del servidor, brindando la posibilidad de generar contenido dinámico para las aplicaciones web. Es uno de los lenguajes de programación más conocidos ya que ayuda a la inserción y manejo de archivos y documentos HTML ayudando al procesamiento de datos con herramientas web. Este lenguaje cuenta con una variedad de características y funcionalidades, siendo capaz de poder instalar PHP en gran variedad de sistemas operativos de manera gratuita. Las características principales de PHP son su velocidad de procesamiento y robustez, además de ser una potente herramienta al poder manejar objetos y es bastante portable ya que su código puede ser llevado e implementado en cualquier proyecto o plataforma [21].

1.5.10. Laravel

Laravel es un framework de PHP para el desarrollo de aplicaciones robustas que sigue una modelo vista controlador (MVC) lo cual brinda un método más ágil para creación tanto de páginas web como para aplicaciones destinadas a la comunicación con el lado del servidor. Laravel maneja una forma distinta de comunicación al momento de responder solicitudes HTTP ya que propone una interacción mediante funciones anónimas que se asocian a una ruta específica para acceder a la información, reduciendo de gran manera la cantidad de código, más aún cuando solo se desea agregar una sola funcionalidad [22].

Ilustración 2 Modelo Vista Controlador



Elaborado por: Fuente [22]

Una de las características más llamativas de Laravel es que cuenta con una herramienta llamada Eloquent ORM (Object Relational Mapper), la cual permite tratar la persistencia de datos, como si se quisiera acceder los métodos o características de un objeto o clase en PHP. El ORM permite mapear y enlazar los objetos de PHP a las tablas que contenga la base de datos, permitiendo de esta manera tener el control sobre dichas tablas y realizar procesos como Insert, Update, Delete de la información directamente con métodos del framework, incrementando el nivel de seguridad por parte del servidor [22].

1.5.11. MySQL

MySQL es un sistema de código abierto para la gestión de base de datos relacionales que maneja un modelo cliente servidor. En la actualidad el manejo de MySQL se ve enfocado en el ámbito empresarial pese a ser gratuito tiene bastante aceptación en los desarrolladores profesionales, debido a su facilidad de uso, implementación y todas las características que ofrece. Una de sus características esenciales de MySQL es que permite almacenar datos con múltiples motores de almacenamiento como InnoDB, CSV y NDB. A su vez es capaz de generar replica de datos y la partición de tablas para aumentar su velocidad de acceso a los datos, durabilidad y el rendimiento. La gran ventaja de MySQL es que ocupa los comandos básicos de SQL para poderlo manejar [23].

1.5.12. Repositorios Virtuales

La necesidad de almacenar la documentación de las instituciones educativas ha llevado a los desarrolladores de software a implementar sistemas de archivos o técnicas que puedan almacenar grandes cantidades de documentación ya sea en una base de datos o equipos físicos. Pero la gran mayoría busca la facilidad al momento de acceder a los documentos almacenados por lo que se ha ido mejorando los motores de búsqueda de información en las interfaces que trabajan con los usuarios de estos almacenes de información conocidos como repositorios o bibliotecas virtuales, los mismos que componen un conjunto de servicios para almacenar, catalogar y mostrar la documentación. Actualmente existen y se manejan repositorios virtuales en el ámbito educativo e investigativo ya que las instituciones buscan almacenar su documentación y ponerlas a disposición de las personas que pretendan buscar ayuda en sus investigaciones [24].

Tabla 2. Tipos de Repositorios Virtuales

Nombre	Descripción
Repositorio Académico	Guarda documentos resultantes de la investigación de los docentes y su actividad académica
Repositorio de Investigación	Almacena la documentación resultante de investigaciones por parte de los miembros de la institución como tesis, artículos, congresos, entre otros.
Repositorio Institucional	Guarda los documentos de la institución y los estudiantes, como informes, boletas, discursos, entrevistas.
Repositorio Multimedia	Almacena los archivos digitales y multimedia que se genere en la institución.
Repositorios Temáticos	Almacenan información relacionada a un tema o área específica.

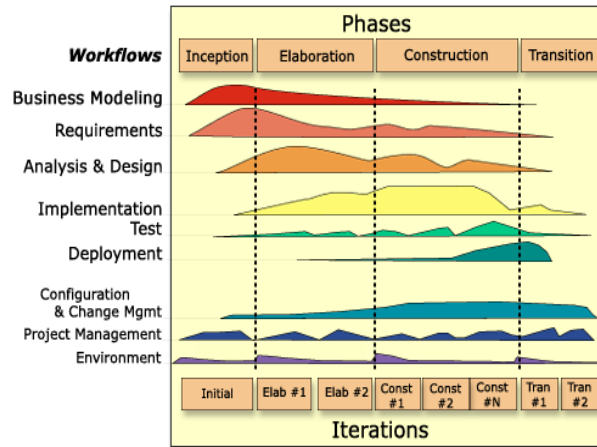
Elaborado por: El Investigador.

1.5.13. Metodologías Tradicionales

En un inicio el desarrollo de software conllevaba procesos singulares y artesanales, con una gran necesidad de generar proyectos que cumplan la meta deseada en su totalidad se acoge fundamentos y metodologías presentes en otras áreas de trabajo y se las adapta al desarrollo de software, con esto el individuos inmersos en el desarrollo centran su trabajo en etapas que manejan una secuencia de lo que se debe hacer y cumplir lo cual mejoraba lentamente la necesidad de la implementación del software. Las principales representantes de las metodologías tradicionales es RUP(Rational Unified Process), que busca la manera de generar proyectos con documentación extensa con el fin de cumplir con la planificación del proyecto definiendo todo sin dejar margen a la suposición y dejando definida en un inicio todas las fases del proyecto, RUP es un proceso de acercamiento disciplinado que permite asignar tareas y sus responsables dentro de un grupo de desarrollo, permitiendo de esta manera generar software de gran calidad con el fin de satisfacer los requerimientos del usuario cumpliendo de gran manera el cronograma y presupuesto establecido al inicio del proceso. Utiliza el modelado y lenguaje UML para

la notación de sus procesos, consta de 4 fases o ciclo de vida para su desarrollo como se muestra en la Ilustración 3 [25].

Ilustración 3 Fases RUP



Elaborado por: Fuente [25].

1.5.14. Metodologías Ágiles

Las metodologías ágiles surgen como respuesta a problemas con aspectos puntuales como el tiempo en la toma de decisiones y la planificación, potenciando el desarrollo a gran escala de nuevos proyectos de software. Se destaca en el manifiesto ágil la interacción que puede llegar a tener las herramientas de los procesos de los usuarios con los individuos inmersos en el desarrollo, además tomar en cuenta que uno de los objetivos es generar un producto entendible y flexible a un producto con una documentación excesiva.

Una de las principales representantes de la metodología ágil es XP (Extreme Programming), la cual se basa y prioriza en la adaptabilidad que puede llegar a tener un proyecto que, en la previsibilidad, la gran parte de usuarios de XP consideran que uno de los requisitos que no se debe tomar a la ligera son los cambios de requisitos, ya que es una parte inevitable en el desarrollo de los nuevos proyectos [25].

Tabla 3. Características Principales de XP

Desarrollo iterativo e incremental	Los cambios y las mejoras se realizan sucesivamente una tras otra.
---	--

Pruebas unitarias continuas	Las pruebas son más repetitivas y automatizadas, además se toma en cuenta el análisis código escrito antes de su codificación.
Programación por parejas	Las tareas de desarrollo deben realizarse en pares, priorizando la revisión del código y discutirlo para evitar pérdida de productividad.
Frecuente interacción	La comunicación del usuario con el equipo de programación es esencial.
Corrección	Se debe verificar errores en las funcionalidades antes de agregar nuevas.
Refactorización	La legibilidad y mantenibilidad del código puede ser replicado en otros proyectos, por lo que se debe reescribir y comprobar su funcionamiento.
Propiedad del código compartida	Los integrantes o grupos de trabajo pueden aportar al trabajo de los demás, con el fin de encontrar posibles errores o extender partes del proyecto.
Simplicidad en el código	XP busca la simplicidad y funcionalidad para poder agregar funcionalidades si fuera necesario a futuro a algo complejo y que no se pueda extender.

Elaborado por: El investigador.

1.5.15. Comparativa Metodologías Tradicionales y Metodologías Ágiles

Es de gran importancia conocer las diferencias entre las metodologías tradicionales y ágiles dado que permite tomar en cuenta la que más se adapte al tipo de proyecto que se va a realizar.

Tabla 4. Comparativa de Metodologías

Metodologías Tradicionales	Metodologías Agiles
Ventajas	Ventajas
Los cambios de objetivos se pueden dar en cada fase	La retroalimentación de los usuarios es primordial.
Los proyectos de innovación se adaptan de gran manera.	Define los objetivos a alcanzar después de cada iteración.
Es sencillo y se guía por la intuición para desarrollar software	El trabajo se distribuye equitativamente en todas las fases del proyecto.
Se detalla la documentación de cada fase	Se adapta mejor a entornos volátiles.
Desventajas	Desventajas
La evaluación para los riesgos se los debe realizar de una forma compleja.	El alcance del proyecto es delimitado por el cliente mas no por los desarrolladores.
La flexibilidad para la mayoría de los proyectos debe ser excesiva.	El desarrollo de documentación queda en segundo plano.
El cliente debe describir y detallar el alcance del proyecto a acordado lo que puede ser muy incómodo para él.	

Elaborado por: El investigador.

1.6. Objetivos

1.6.1. Objetivo General

Desarrollar un Repositorio Digital para el manejo de documentación de Unidades Fiscales con una Aplicación Web Progresiva en la Unidad Distrital de Tecnologías de Información y Comunicaciones del Distrito 18D01-Ambato 1 Educación.

1.6.2. Objetivos Específicos

- Analizar los métodos de recolección de información en las Unidades Educativas Fiscales de la Unidad Distrital.
- Definir los métodos existentes para el almacenamiento de información e identificar sus ventajas y desventajas.
- Analizar las herramientas para el desarrollo e implementación de las aplicaciones web progresivas.
- Definir las herramientas y equipos necesarios para el desarrollo e implementación de la aplicación.
- Desarrollar el repositorio y la aplicación web requeridas por la Unidad Distrital

CAPITULO II: METODOLOGÍA

2.1. Materiales

Para la recolección de información en el presente proyecto se aplicó una encuesta mediante un cuestionario al personal de la unidad de tecnologías de información del distrito con la finalidad de conocer la manera en que se maneja la información y la factibilidad que tiene al momento de implementar la aplicación. De igual manera se aplicó una encuesta a los rectores de las unidades educativas para conocer la factibilidad que tiene implementar la aplicación.

Además, se realizó una matriz de observación para analizar los procesos tanto de recolección, tratamiento, almacenamiento y manipulación de la información de cada unidad educativa fiscal en el escenario en este caso la unidad de tecnologías de información del distrito.

ENCUESTA

Primera parte: Personal del Distrito – 9 Personas

Objetivo: conocer la manera en que se maneja la información de las unidades educativas fiscales y la factibilidad que tiene implementar la aplicación.

Pregunta N.1: ¿Considera que el personal encargado de manejar la información de todas las unidades educativas fiscales de Ambato es muy reducido?

Si

No

Pregunta N.2: La información obtenida de las unidades educativas las almacenan en:

Servidor de Archivos

Computador de Trabajo

Disco Duro Externo

Memoria USB

Pregunta N.3: ¿Con que frecuencia los rectores de las instituciones educativas solicitan su documentación al departamento de tecnologías?

- Nunca
- A veces
- Casi Siempre
- Siempre

Pregunta N.4: ¿Al momento de solicitar la información de una unidad educativa, el responsable se encarga de subir la información en su totalidad con el informe que lo respalde?

- Nunca
- A veces
- Casi Siempre
- Siempre

Pregunta N.5: ¿Consideraría manejar una herramienta tecnológica que permita acceder a la información de una unidad educativa en cualquier lugar que se encuentre?

- Si
- No

Pregunta N.6: Para acceder a la información de una unidad educativa le gustaría realizarla por:

- Celular
- Computador

Pregunta N.7: ¿Le gustaría poder encontrar la información de una unidad educativa de manera ordenada para posteriormente poder visualizarla o descargarla?

- Si
- No

Pregunta N.8: ¿Le gustaría conocer información, comunicados y eventos que conciernan a las unidades educativas fiscales en general y al Distrito de Educación de Ambato?

Si

No

Segunda parte: Rectores – 40 Personas

Objetivo: conocer factibilidad y requerimientos que se necesiten al implementar la aplicación.

Pregunta N.1: ¿Consideraría manejar una herramienta tecnológica que permita acceder a la información de una unidad educativa en cualquier lugar que se encuentre?

Si

No

Pregunta N.2: Para acceder a la información de una unidad educativa le gustaría realizarla por:

Celular

Computador

Pregunta N.3: ¿Le gustaría poder encontrar la información de una unidad educativa de manera ordenada para posteriormente poder visualizarla o descargarla?

Si

No

Pregunta N.4: ¿Le gustaría conocer información, comunicados y eventos que conciernan a las unidades educativas fiscales en general y al Distrito de Educación de Ambato?

Si

No

MATRIZ DE OBSERVACIÓN

Lugar: Unidad Distrital de Tecnologías de Información y Comunicaciones

Objetivo: Analizar los procesos y procedimientos de recolección, tratamiento, almacenamiento y manipulación de la información de las unidades educativas fiscales para determinar funcionalidades y requerimientos.

Tabla 5. Matriz de Observación

N°	Departamento	Criterio de Evaluación	Descripción
1	Departamento de Apoyo Seguimiento y Regulación (ASRE)	Recolección de Información Académica	Frecuencia que se solicita la documentación académica (boletas, promociones y concentrados) a las Unidades Educativas.
			Forma en que se solicita la documentación.
			Quien solicita la documentación.
			Como se recibe la documentación.
2		Tratamiento de la Información Académica	Verificar la documentación recibida.
			Clasificar la documentación.
3	Unidad Distrital de Tecnologías de Información y Comunicaciones	Almacenamiento de la Información Académica	Verificar la documentación recibida.
			Almacenar la información académica.
4		Manipulación de la Información Académica	Quien solicita la documentación y como.
			Cuál es la documentación que más se solicita.
			Como se emite la documentación solicitada.

Elaborado por: El Investigador.

2.2 Métodos

2.2.1 Modalidad básica de la investigación

Investigación bibliográfica

Para el análisis y desarrollo de la aplicación se reforzó los conocimientos con el manejo de documentación técnica presente en libros, artículos científicos y proyectos similares.

Investigación de campo

Se contó con la experiencia de los trabajadores de la unidad para definir y detallar las características fundamentales del problema para poder tener un enfoque claro de la solución.

Investigación experimental

Se pudo analizar las causas del problema y proponer la mejor técnica para el manejo de la documentación.

2.2.2 Población y Muestra

La población de estudio fueron 49 personas que manejan la documentación de las unidades educativas fiscales. En base a las personas con las que cuenta la población no se considera una cantidad aceptable para tener una muestra.

N°	Tipo de Población	Descripción	Número	Porcentaje
1	Administradores de TI	Unidad Distrital de Tecnologías de Información y Comunicaciones	3	6,12%

2	Analistas	Departamento de Apoyo Seguimiento y Regulación (ASRE)	6	12,24 %
3	Rectores	Unidades Educativas Fiscales	40	81,64%
		Total	49	100%

Tabla 6. Población

Elaborado por: El Investigador.

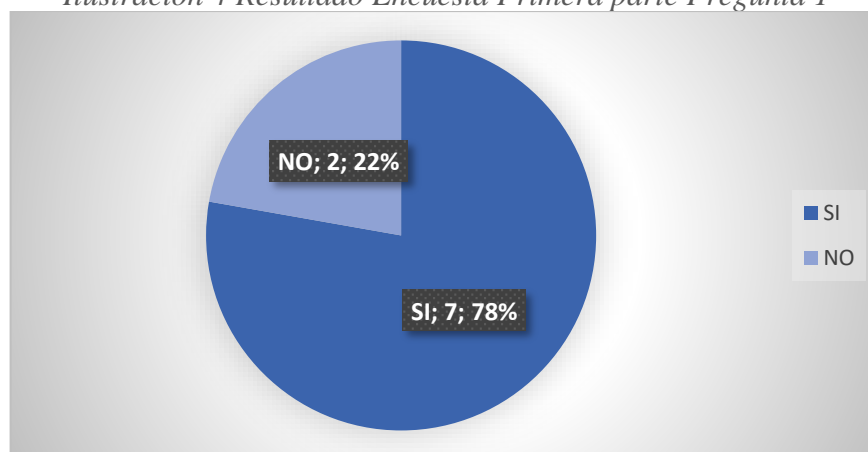
2.2.3. Recolección de información

Una vez realizada la encuesta a los analistas del departamento de apoyo seguimiento y regulación ASRE y a los administradores de TI de la unidad distrital se pudo obtener los siguientes resultados:

Primera parte: Personal del Distrito – 9 Personas

Pregunta N.1: ¿Considera que el personal encargado de manejar la información de todas las unidades educativas fiscales de Ambato es muy reducido?

Ilustración 4 Resultado Encuesta Primera parte Pregunta 1



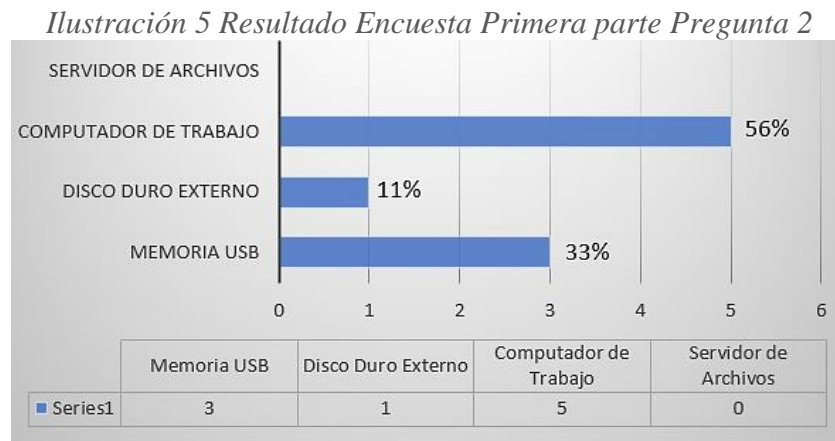
Elaborado por: El Investigador.

Análisis e Interpretación

El 78% del personal del departamento de Tecnologías de Información del distrito considera que el número de personas encargadas tanto del manejo y procesamiento de información es muy reducido, debido a esto buscan una manera de agilizar los procesos

tanto de recolección como de control de la información de todas las instituciones educativas que están a su cargo.

Pregunta N.2: La información obtenida de las unidades educativas las almacenan en:



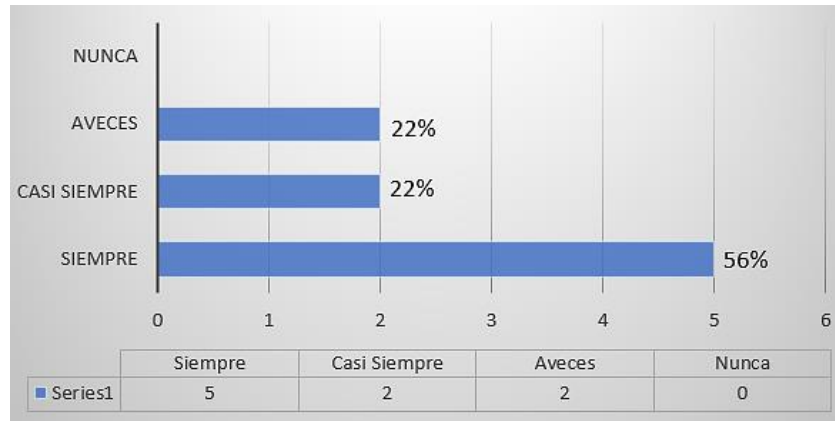
Elaborado por: El Investigador.

Análisis e Interpretación

La mayor parte representada por el 56% del personal encuestado manifiesta que la información es guardada en su computador de trabajo, seguido por el 33% que guarda en un disco duro externo y un 11% lo hacen en una memoria USB, lo que provoca que la información este muy dividida y haga muy difícil tener un control, además de dificultar el proceso de búsqueda. Por otro lado, la información no está segura y puede estar expuesta a perdida.

Pregunta N.3: ¿Con que frecuencia los rectores de las instituciones educativas solicitan su documentación al departamento de tecnologías?

Ilustración 6 Resultado Encuesta Primera parte Pregunta 3



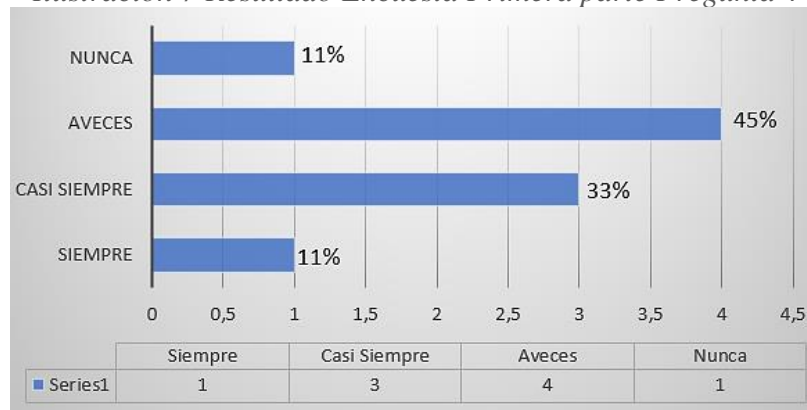
Elaborado por: El Investigador.

Análisis e Interpretación

El 56% del personal del departamento considera que la documentación que ellos manejan es solicitada por parte de los rectores de las unidades educativas de manera frecuente, mientras que un 22% solicita a veces y casi siempre. De estos resultados hay que tomar en cuenta que los rectores no guardan la información de su institución por lo que la solicitan al departamento del distrito, haciendo que el personal tome parte de su tiempo en buscar la información retrasando otros procesos.

Pregunta N.4: ¿Al momento de solicitar la información de una unidad educativa, el responsable se encarga de subir la información en su totalidad con el informe que lo respalde?

Ilustración 7 Resultado Encuesta Primera parte Pregunta 4



Elaborado por: El Investigador.

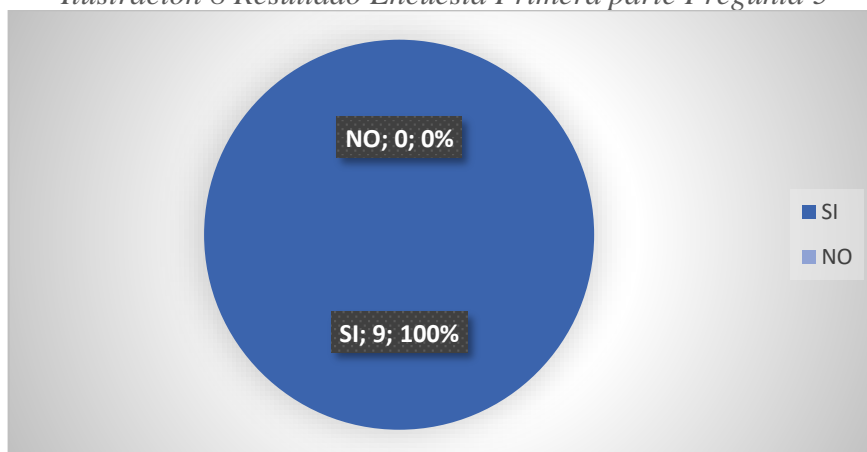
Análisis e Interpretación

El 45% del personal respondió que solo a veces la información es respaldada con un informe que detalle los archivos entregados a la unidad de tecnologías del distrito, por otro lado, un 33% dijo que la mayoría de las veces lo respaldan, mientras que un 11% respondió que nunca y un 11% que siempre lo realizan.

Hay que tomar en cuenta que la información de las instituciones educativas fiscales que maneja el departamento del distrito es muy extensa por lo que siempre debe tener un documento que respalde a detalle toda la información que se envía, para poder tener constancia y facilitar búsquedas y solicitudes de la información en un futuro.

Pregunta N.5: ¿Consideraría manejar una herramienta tecnológica que permita acceder a la información de una unidad educativa en cualquier lugar que se encuentre?

Ilustración 8 Resultado Encuesta Primera parte Pregunta 5



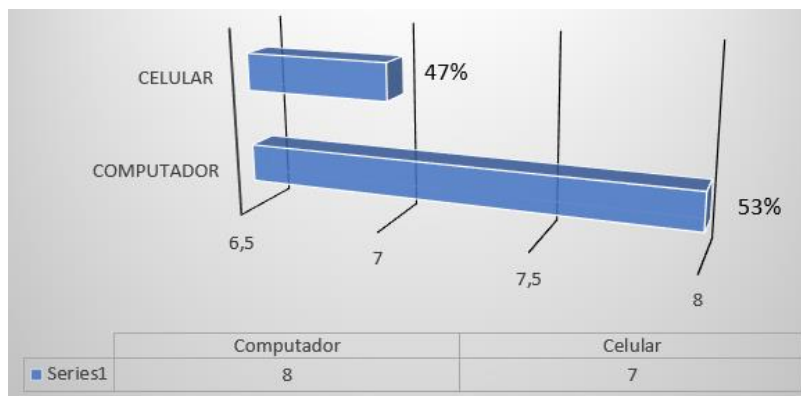
Elaborado por: El Investigador.

Análisis e Interpretación

El 100% de los encuestados consideran que el manejo de una herramienta para poder acceder a la información de una unidad educativa sería muy factible ya que agilizaría los procesos tanto del departamento del distrito como de los rectores.

Pregunta N.6: Para acceder a la información de una unidad educativa le gustaría realizarla por:

Ilustración 9 Resultado Encuesta Segunda parte Pregunta 2



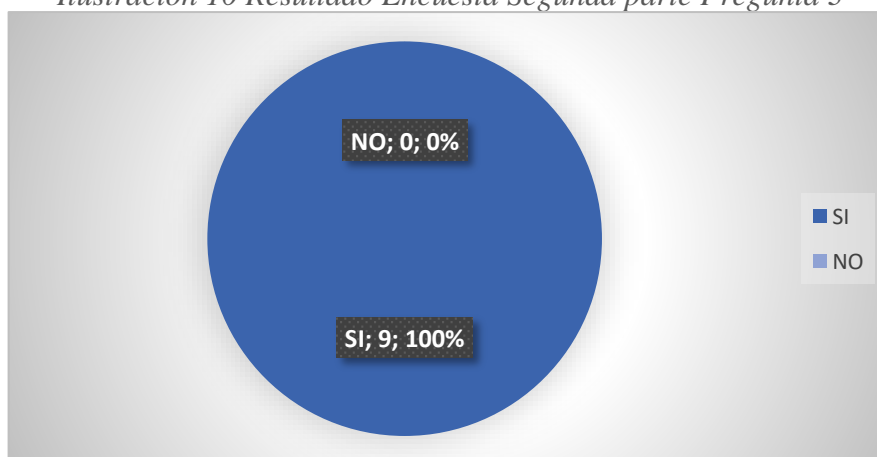
Elaborado por: El Investigador.

Análisis e Interpretación

El 53% de los encuestados consideran que para acceder a la información de una unidad educativa prefieren usar el computador, mientras que un 47% respondieron que les gustaría hacerlo por el celular, por lo que el desarrollo de una aplicación web progresiva es factible ya que puede ser utilizada y visualizada tanto en dispositivos móviles como teléfonos celulares y de igual manera en un computador.

Pregunta N.7: ¿Le gustaría poder encontrar la información de una unidad educativa de manera ordenada para posteriormente poder visualizarla o descargarla?

Ilustración 10 Resultado Encuesta Segunda parte Pregunta 3



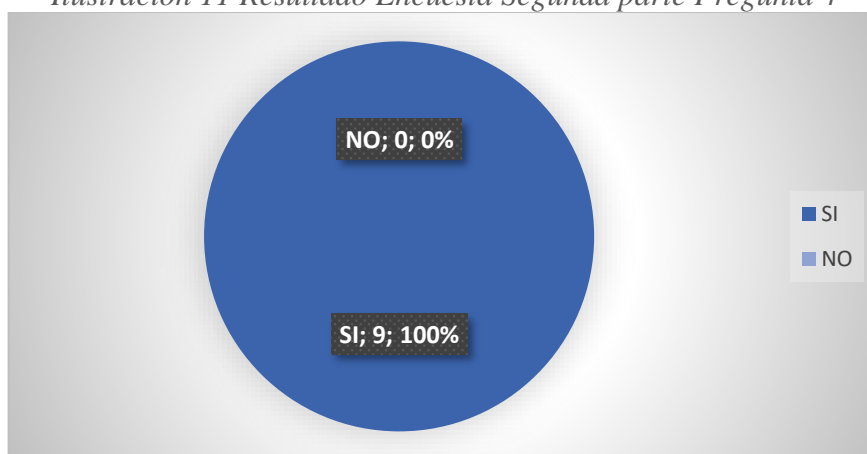
Elaborado por: El investigador.

Análisis e Interpretación

El 100% de los encuestados considera que una herramienta que permita visualizar la información de manera ordenada para posteriormente poder descargarla según la necesiten sería de gran ayuda en sus puestos de trabajo.

Pregunta N.8: ¿Le gustaría conocer información, comunicados y eventos que conciernan a las unidades educativas fiscales en general y al Distrito de Educación de Ambato?

Ilustración 11 Resultado Encuesta Segunda parte Pregunta 4



Elaborado por: El Investigador.

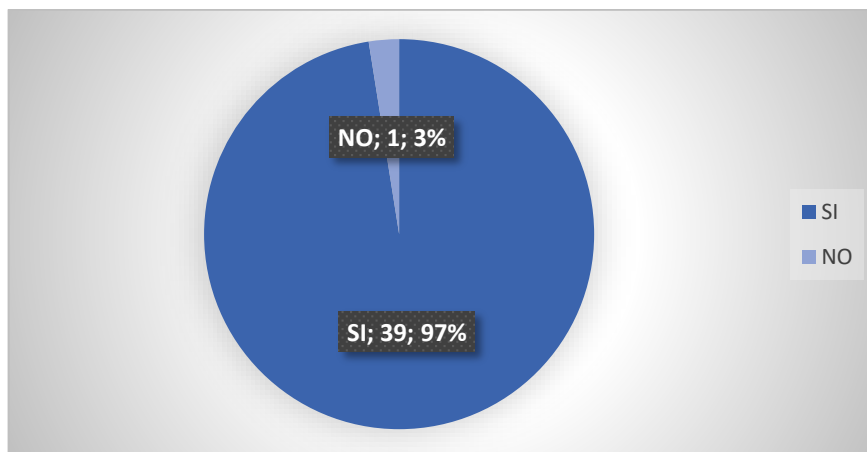
Análisis e Interpretación

El 100% de los encuestados respondió que les gustaría conocer información, comunicados y eventos que conciernan a las unidades educativas fiscales en general y al Distrito de Educación de Ambato.

Segunda parte: Rectores de las Unidades Educativas Fiscales – 40 Personas

Pregunta N.1: ¿Consideraría manejar una herramienta tecnológica que permita acceder a la información de una unidad educativa en cualquier lugar que se encuentre?

Ilustración 12 Resultado Encuesta Segunda parte Pregunta 1

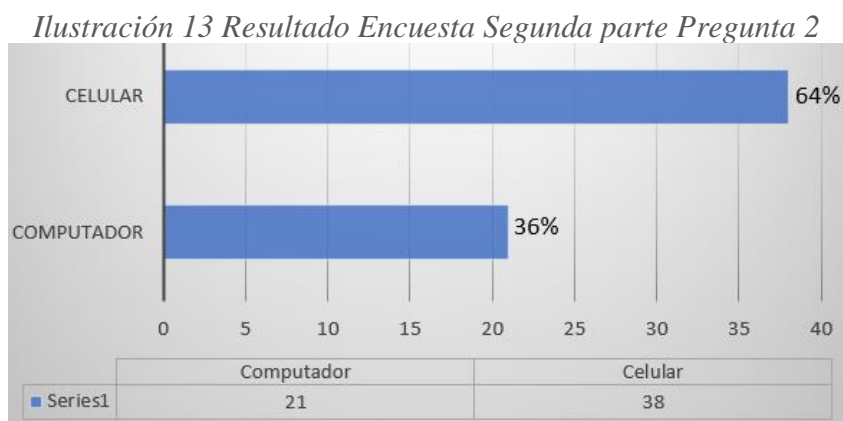


Elaborado por: El Investigador.

Análisis e Interpretación

El 97% de los encuestados consideran que el manejo de una herramienta para poder acceder a la información de una unidad educativa sería muy factible ya que agilizaría los procesos tanto del departamento del distrito como de los rectores.

Pregunta N.2: Para acceder a la información de una unidad educativa le gustaría realizarla por:



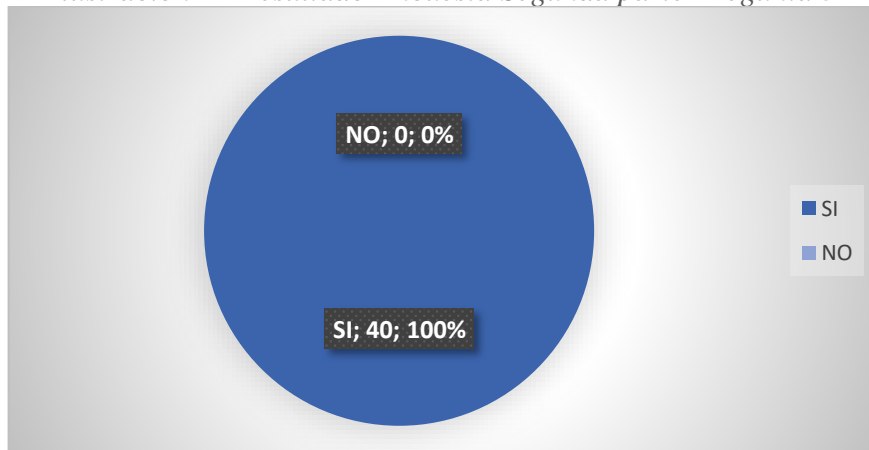
Elaborado por: El Investigador.

Análisis e Interpretación

El 64% de los encuestados consideran que para acceder a la información de una unidad educativa prefieren usar el teléfono celular, mientras que un 36% respondieron que les gustaría hacerlo por el computador, por lo que el desarrollo de una aplicación web progresiva es factible ya que puede ser utilizada y visualizada tanto en dispositivos móviles como teléfonos celulares y de igual manera en un computador.

Pregunta N.3: ¿Le gustaría poder encontrar la información de su unidad educativa de manera ordenada para posteriormente poder visualizarla o descargarla?

Ilustración 14 Resultado Encuesta Segunda parte Pregunta 3



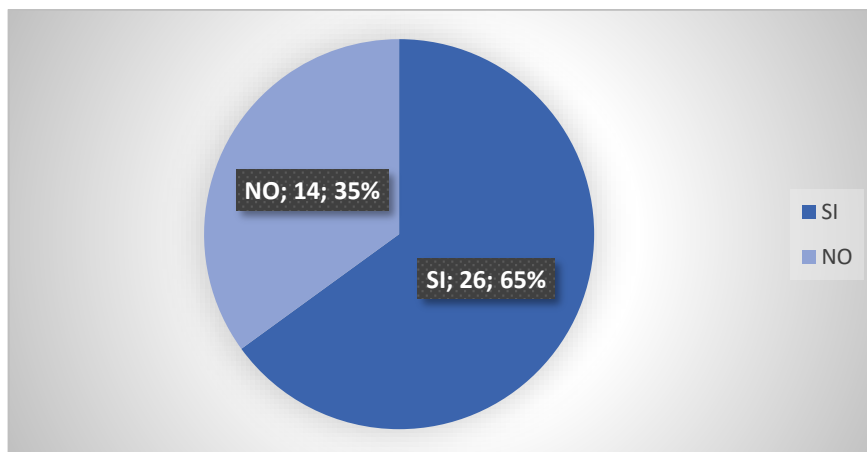
Elaborado por: El investigador.

Análisis e Interpretación

El 100% de los encuestados considera que una herramienta que permita visualizar la información de manera ordenada para posteriormente poder descargarla según la necesiten sería de gran ayuda en sus puestos de trabajo.

Pregunta N.4: ¿Le gustaría conocer información, comunicados y eventos que conciernan a las unidades educativas fiscales en general y al Distrito de Educación de Ambato?

Ilustración 15 Resultado Encuesta Segunda parte Pregunta 4



Elaborado por: El Investigador.

Análisis e Interpretación

El 65% de los encuestados respondió que les gustaría conocer información, comunicados y eventos que conciernan a las unidades educativas fiscales en general y al Distrito de Educación de Ambato, mientras un 35% respondieron que no.

Una vez realizada la matriz de observación en los dos departamentos del distrito encargados de la documentación de las unidades educativas se pudo obtener las siguientes observaciones y conclusiones.

Tabla 7. Matriz de Observación

N°	DEPARTAMENTO	CRITERIO DE EVALUACIÓN	DESCRIPCIÓN	OBSERVACIÓN	CONCLUSIÓN
1	Departamento de Apoyo Seguimiento y Regulación (ASRE)	Recolección de Información Académica	Frecuencia que se solicita la documentación académica (boletas, promociones y concentrados) a las Unidades Educativas.	1 vez por quimestre se solicita el concentrado de los paralelos de las unidades educativas. Al final del periodo lectivo se solicita las boletas y promociones de los estudiantes.	Se concluye que el personal del Departamento de Apoyo Seguimiento y Regulación del día a día es el encargado de llevar a parte preliminar el proceso de recolección de información ya que ellos se encargan de solicitar y verificar la información que proviene de diferentes unidades educativas, pero debido a que el número de unidades educativas es muy grande, el proceso de clasificación de documentos puede llevar mucho tiempo.
			Forma en que se solicita la documentación.	Se emite un comunicado mediante un informe a los rectores de las unidades educativas fiscales solicitando la documentación pertinente.	
			Quien solicita la documentación.	Un analista es el encargado de solicitar la documentación de las unidades educativas.	
			Como se recibe la documentación.	La información se la recibe en un archivo .rar denominado “empaquetado” el cual consta con los concentrados, boletas y promociones.	
2		Tratamiento de la Información Académica	Verificar la documentación recibida.	Los analistas revisan si el número de boletas, promociones o concentrados que reciben están completos y correctamente emitidos.	
			Clasificar la documentación.	Una vez comprobada la documentación, proceden a clasificarla según la unidad	

				educativa, periodo lectivo, nivel o curso, jornada y paralelo según corresponda. Posteriormente guardan una copia de seguridad y se envía a la unidad distrital de tecnologías de información.	
3	Unidad Distrital de Tecnologías de Información y Comunicaciones	Almacenamiento de la Información Académica	Verificar la documentación recibida.	Se revisa si la documentación esta ordenada correctamente.	Se concluye que la unidad distrital de tecnologías de información es encargada del almacenamiento de la documentación y posteriormente revisada y clasificada. Percuentan con un espacio específico para almacenar la información y acceder a ella cuando la solicita
			Almacenar la información académica.	Se procede a almacenar la información en un computador o disco duro extraíble, registrando en un documento de Excel su ubicación.	
4	Unidad Distrital de Tecnologías de Información y Comunicaciones	Manipulación de la Información Académica	Quien solicita la documentación y como.	Los rectores de las unidades educativas mediante una llamada telefónica o correo solicitan boletas, promociones de algún estudiante al departamento de regulación.	Se concluye que la unidad distrital de tecnologías de información es encargada del almacenamiento de la documentación y posteriormente revisada y clasificada. Percuentan con un espacio específico para almacenar la información y acceder a ella cuando la solicita
			Cuál es la documentación que más se solicita.	Boletas y Promociones de estudiantes.	

Elaborado por: El Investigador.

Tabla 8. Recolección de Información-Matriz de Observación

Proceso a Realizar	Descripción	Observación
--------------------	-------------	-------------

Clasificación de la información.	El departamento de seguimiento y regulación no cuenta con una aplicación que le permita clasificar y almacenar la documentación de las unidades educativas de manera rápida y segura. La clasificación se realiza de forma manual por lo que toman demasiado tiempo debido al número de unidades educativas que tienen a su cargo.	La clasificación de la documentación se lo realiza según la unidad educativa, el periodo lectivo, el nivel o curso, la jornada (matutina o vespertina), el paralelo y por último la documentación del estudiante boletas, promociones y concentrado.
Manejo del listado de Unidades Educativas	Para el inicio del proceso de clasificación de la información es necesario conocer la unidad educativa con la que se va a trabajar, para poder evitar errores al momento de asignar la información.	Para evitar errores al momento de seleccionar la unidad educativa por temas de duplicidad de nombres, cada unidad educativa debe contar con el nombre, la dirección y su logotipo para de esta manera diferenciarlas.
Manejo de periodos lectivos	Posteriormente a la selección de la unidad educativa se debe manejar el periodo lectivo al que se va a acceder, por lo que se debe tomar en cuenta que no todas las unidades educativas pueden contar con los mismos periodos lectivos.	En los periodos lectivos se asigna a la unidad educativa seleccionada y posteriormente debe llevar un nombre que lo identifique.
Manejo de niveles o cursos.	Una vez seleccionado un periodo lectivo en una unidad educativa se toma en cuenta el listado de cursos o niveles con los que cuenta la unidad educativa.	Según la jornada que se seleccione se presenta el listado de niveles o cursos diferenciados por su nombre.

	Tomando en cuenta que las unidades educativas trabajan en dos jornadas matutina y vespertina.	
Manejo de paralelos	Al seleccionar un nivel o curso se presenta los paralelos con los que cuenta dicho nivel.	Los paralelos se ven diferenciados por su nombre.
Manejo de documentación estudiantil	Al acceder a un paralelo seleccionado se presenta la información de los estudiantes boletas, promociones y concentrado.	La información se clasifica según los 3 tipos de documentos que se almacena boletas y promociones que cuentan con el número de cedula del estudiante y el concentrado que almacena el listado de estudiantes y calificaciones. La documentación se podrá visualizar o descargar.
Manejo de documentación institucional	Las unidades educativas pueden enviar información que solicite el departamento distrital para revisarla, almacenarla y utilizarla cuando se la requiera.	La documentación debe contar con una categoría a la que pertenecer (Oficios, informes, actas, entre otros). La documentación se podrá visualizar o descargar.
Almacenamiento de la información	El departamento de tecnologías de la información una vez recibe el empaquetado de documentos por parte del departamento de regulación procede revisar si la clasificación es correcta y a almacenar la información.	La documentación se la guarda en una computadora central que manejan los administradores o en discos duros extraíbles.

Gestión de Usuarios	Para el manejo de información se toma en cuenta los principales actores como son los rectores, delegados institucionales encargados de proveer la documentación al departamento distrital y sus administradores y analistas del departamento.	Se toma en cuenta la gestión de usuarios mediante un rol de acceso a la información académica como visualizar, descargar o ingresar y suprimir la documentación. Para ello el usuario debe contar con su nombre, el correo que será su usuario, la contraseña, la unidad educativa a la que esta asignado y el rol que cumple dentro del sistema.
Reportes de informes de constancia	Cuando se recibe el empaquetado de información de una unidad educativa se recibe un informe con el listado de toda la documentación enviada. Este documento sirve como constancia de que se provee la información.	Se debe tomar en cuenta que el documento se debe generar cada vez que se registra nueva documentación.

Elaborado por: El Investigador.

2.2.4. Procesamiento y análisis de datos.

En base a la información obtenida por parte del personal del distrito, se define que llevar el control de la información de todas las unidades educativas fiscales en la ciudad de Ambato es un trabajo muy grande y se necesita la ayuda de un sistema que permita agilizar tanto los procesos de recolección, almacenamiento y búsqueda de la información.

Por otro lado, la segunda parte de la encuesta define la aceptación sobre la aplicación de un sistema que permita solventar las necesidades de acceder a la información de las unidades educativas y la forma en la que quieren acceder a ella.

Con la matriz de observación realizada en los dos departamentos del distrito que manejan la documentación de las unidades educativas se pudo conocer como es el proceso de recolección de información, y se pudo determinar características y requerimientos como la manera para clasificar la información de manera sucesiva, el nivel de acceso que deben tener los usuarios y el proceso posterior al momento de registrar la documentación.

CAPITULO III: RESULTADOS Y DISCUSIÓN

3.1. Análisis general de resultados

3.1.1. Métodos de recolección de información

En los departamentos y unidades del distrito cuentan con 2 procesos de solicitud informatizada para la recolección de información con los que obtienen la documentación de las unidades educativas:

Tabla 9. Métodos o procesos de Recolección de Información

Nombre del Método Característica	Solicitud Informatizada (Quimestral/Parcial)	Solicitud Informatizada (al final del periodo lectivo)
¿Quién lo realiza?	Departamento de Apoyo Seguimiento y Regulación (ASRE)	Departamento de Apoyo Seguimiento y Regulación (ASRE)
¿Cuántas veces?	2 por periodo lectivo	1 vez por periodo lectivo
¿Cuándo lo realiza?	Cuando se emiten las calificaciones correspondientes al quimestre o parcial que haya terminado en el periodo lectivo.	Cuando el periodo lectivo está por terminar, se solicitan las boletas, promociones y concentrado de los estudiantes.
Periodo para la recolección de información.	8 días a partir de la solicitud.	15 días a partir de la solicitud.
¿Cómo se solicita la información?	Notificación al Rector o responsable de la unidad Educativa.	Notificación al Rector o responsable de la unidad Educativa.
¿Cómo se recepta la información?	Los rectores o los encargados de proveer el empaquetado académico	Los rectores o los encargados de proveer el empaquetado académico deben enviar un

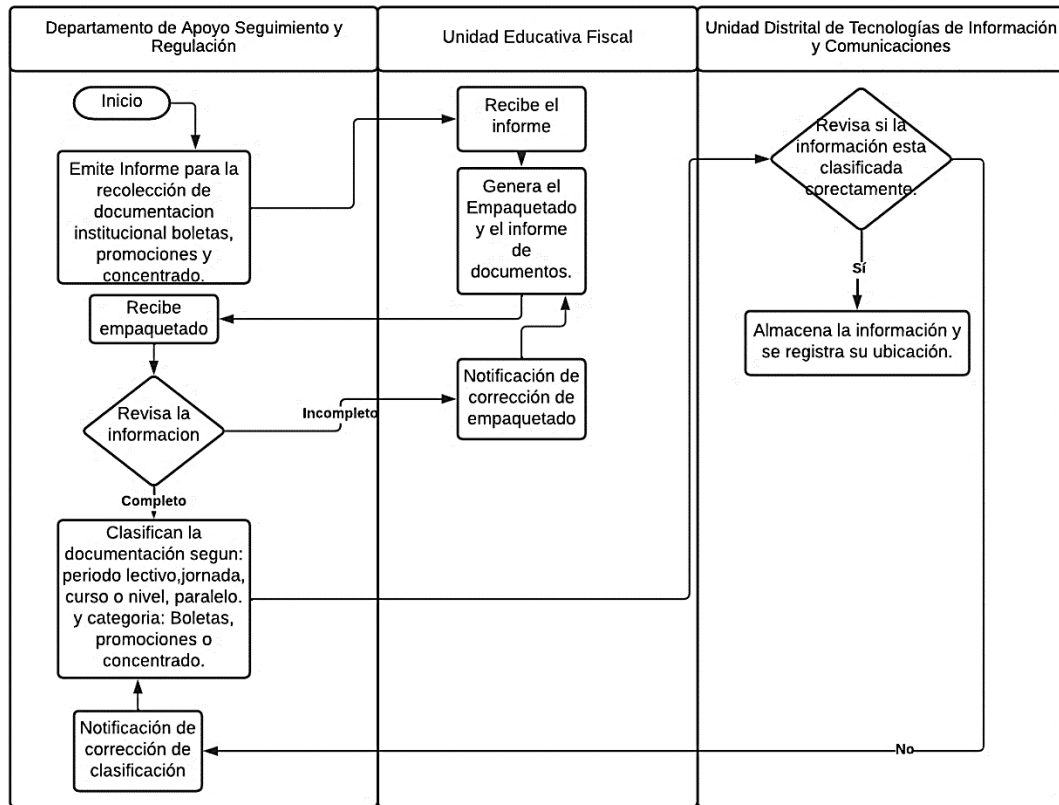
	deben enviar un informe con el listado de toda la documentación que se envía al departamento.	informe con el listado de toda la documentación que se envía al departamento.
¿Quién almacena la información?	Departamento de Apoyo Seguimiento y Regulación (ASRE) / Unidad Distrital de Tecnologías de Información y Comunicaciones	Departamento de Apoyo Seguimiento y Regulación (ASRE) / Unidad Distrital de Tecnologías de Información y Comunicaciones
¿Cómo se almacena la información?	Computador, USB, Discos Duro, Almacenamiento en la nube	Computador, USB, Discos Duro, Almacenamiento en la nube

Elaborado por: Fuente [26]

3.1.2. Procedimiento para la solicitud de informatizada a las unidades educativas.

En el proceso para la solicitud de documentación están envueltos tanto el departamento de regulación como la unidad de tecnologías de información del distrito por lo cual se detalló el proceso a seguir mediante el cual se recolecta la documentación de las diferentes unidades educativas.

Ilustración 16 Procedimiento para la solicitud informatizada.



Elaborado por: Fuente [26].

Una vez analizado el proceso mediante el cual pasa la documentación de las unidades educativas en el departamento distrital, se toma en cuenta que todo el proceso se lo realiza de manera manual tanto el de clasificación como el de almacenamiento de la información. Debido a esto el proceso para todas las unidades educativas de las que está a cargo el departamento puede llevar horas, incluso días en ser terminado, ralentizando de esta manera el avance de los procesos o proyectos alternos.

3.2.2. Métodos de almacenamiento de información

Tomando en cuenta los métodos de recolección de información se analizó las diferentes formas de almacenar la información de las unidades educativas para poder definir la mejor opción para la solución de la aplicación. Además, se tomó en cuenta que en un repositorio de almacenamiento de archivos el principal problema a tomar en cuenta es el lugar en donde se va a almacenar los datos, ya que presentan cuestiones como:

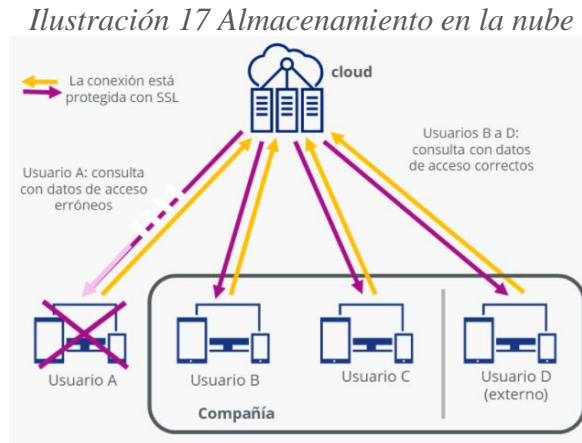
- Seguridad de la Información.
- Velocidad de acceso.
- Capacidad de almacenamiento requerido.
- Manera/forma de almacenar los archivos.
- Costo.

En el proceso de registro de una unidad educativa en los departamentos del distrito se maneja una aplicación para el registro de la documentación proveniente de la nueva unidad educativa a cargo, esta aplicación almacena la documentación en una base de datos con archivos tipo Blob. Una gran alternativa si se desea almacenar archivos e información en un solo lugar sería el almacenamiento en una base de datos como lo es MySQL, el motor de base de datos integra la posibilidad de almacenar archivos en la base de datos mediante el formato de archivos tipo Blob (Binary Large Object) que nos permite guardar la información de algún elemento en un archivo binario. Pero esta opción cuenta con la desventaja que, al almacenar archivos multimedia de gran tamaño, tiende a sobrecargar de información la base de datos por lo que disminuye el rendimiento y velocidad de respuesta al momento de realizar una petición.

Para el proceso de solicitud informatizada tanto parcial como al final del periodo educativo los analistas del departamento de regulación solicitan la información la clasifican y proceden a enviar la documentación al departamento de tecnologías de información, pero ellos deben tener una copia de seguridad de los empaquetados recibidos. Pero muchos de ellos no tienen un equipo físico principal para guardar ese respaldo por lo que muchos de ellos guardan su información en memorias USB, en disco duros o lo almacenan en la nube.

El almacenamiento en la nube viene a ser un modelo de almacenamiento de datos que ocupa las redes de computadoras, en el que la información se encuentra alojados en diferentes espacios de almacenamiento de manera virtualizada. El conjunto de memorias RAM, discos duros y procesadores que conforman el servidor permiten que la

información pueda ser solicitada en cualquier momento mientras se tenga acceso mediante un usuario y contraseña. La mayoría de servicios de almacenamiento en la nube cuentan con programas o interfaces desde las cuales se pueden acceder a su información [27].



Elaborado por: Fuente [27].

Otra alternativa es el almacenamiento de archivos en un servidor de aplicaciones, de esta manera los archivos se semejan más al almacenamiento en la nube ya que una vez los usuarios desean guardar los archivos, se genera una copia del archivo en su ordenador y lo almacena en el servidor. Los archivos se encuentran almacenados con sus extensiones originales y se pueden acceder de manera directa conociendo la ruta en donde se encuentran alojados.

3.2.2.1 Comparativa métodos de almacenamiento de información

Tabla 10. Comparativa métodos de almacenamiento

	Almacenamiento en la Nube	Almacenamiento en Base de Datos	Almacenamiento en Servidor
Seguridad	Muy Buena	Buena	Buena
Velocidad de acceso	Muy Buena	Buena	Muy Buena
Capacidad de Almacenamiento	Según el plan solicitado	Según lo permita la base de datos	Según la capacidad del servidor
Forma de Almacenar	Extensión original	Comprimido BLOB	Extensión original
Costo	Alto	Alto	Medio

Elaborado por: El Investigador.

3.2.3. Herramienta para el desarrollo de la aplicación web progresiva Front-end

Anteriormente en el apartado del Capítulo II en el punto 2.2.7 Fundamentación teórica se dio a conocer la herramienta Ionic para la creación de aplicaciones web progresivas, a continuación, se realizó la comparativa con dos framework que nos permite realizar el mismo tipo de aplicaciones de tal manera que se analizó y selecciono la herramienta a utilizar.

Tabla 11. Comparativa framework de desarrollo

	Vue	React	Ionic
Respaldo/Autor	No está respaldado por empresas	Sostenido por Facebook	Sostenido por Google y la comunidad
Lenguaje	Basado en JavaScript	Biblioteca de JavaScript	Manejado en JavaScript o TypeScript
Representación gráfica DOM	Utiliza DOM virtual	Utiliza DOM virtual	Manejo de DOM Complejo
Uso	Adecuado para aplicaciones ligeras	Adecuado para aplicaciones ligeras	Adecuado para aplicaciones medianas
Curva de Aprendizaje	Curva de aprendizaje ligera	Curva de aprendizaje ligera	Curva de aprendizaje pesada, pero con un lenguaje más amigable
Flexibilidad	No es muy flexible	Flexible	Mediamente flexible
Vinculación de datos	Unidireccional, siendo los datos inmutables.	Bidireccional, donde los datos son mutables.	Bidireccional, donde los datos son mutables.
Directivas	Sencillo si se tiene conocimiento de JS y CSS.	Sencillo si se tiene conocimiento de JS.	Incomprensible si no se tiene conocimiento de Angular.
Arquitectura	Mediante Componentes	Modo Vista	Modo Vista Controlador
Creación de aplicaciones	Desarrollo de aplicaciones Web.	Tanto para web y apps móviles. Para apps de una sola página y múltiples páginas	Desarrollo de aplicaciones móviles híbridas, aplicaciones de escritorio y PWA.

Elaborado por: El investigador.

El manejo de la herramienta Ionic para la creación de la aplicación web progresiva permitió de gran manera desarrollar un sistema que permita el manejo de la información de las unidades educativas fiscales de manera rápida y ordenada. Con métodos de acceso rápido ya que puede desplegarse en un dispositivo móvil o en un computador. A partir de esto tanto el personal del departamento del distrito y los rectores pueden manejar la aplicación y visualizar o descargar la información requerida. Con esto se pretende iniciar la inclusión de este Framework para el desarrollo de futuras aplicaciones en el departamento de tecnologías del distrito ya que cuenta con una gran variedad de funcionalidades muy fáciles de utilizar e implementar.

3.2.4. Herramienta para el desarrollo de la aplicación Back-End

Para el desarrollo del Back-End del proyecto se analizó herramientas tanto que puedan proveer seguridad, facilidad de implementación, análisis y sobre todo la documentación para poder crear un sistema que pueda ser utilizado de manera eficaz y precisa. Por lo que se analizó 3 herramientas que nos permiten el desarrollo en el apartado de servicios.

Tabla 12. Comparativa herramientas Back-End

	Laravel	ASP.NET	DJango
Lenguaje de Programación	PHP	C#	Python
Reutilización	Funciona en varias plataformas	Funciona en varias plataformas	Se debe modificar para que funcione en otras plataformas
Código Abierto	Si	No	Si
Seguridad	Facilita las tareas comunes utilizadas en la mayoría de los proyectos web, como la autenticación	Se debe agregar extensiones para que la seguridad funcione correctamente	Toma en serio la seguridad y ayuda a los desarrolladores a evitar muchos errores de seguridad comunes
Soporte de plataformas	Si	No, se debe modificar el código para que funcione en otras plataformas	Si
Documentación	Gran cantidad de documentación y ayuda de la comunidad	Gran cantidad de documentación y ayuda de la comunidad	Gran cantidad de documentación y ayuda de la comunidad
Extensiones	Si	Si	Si

Elaborado por: El investigador.

3.2.4. Herramientas para el desarrollo de la aplicación web progresiva Base de Datos.

Para el desarrollo de la base de datos se tomó en cuenta que debía utilizarse una base de datos relacional ya que en los requerimientos cuenta con datos y categorías de objetos enlazados entre sí. Para ello se analizó 3 bases de datos con la que se podía trabajar:

Tabla 13. Comparativa herramientas Base de Datos

	MySQL	PostgreSQL	MongoDB
Código Abierto	Si	Si	Si
Lenguaje	SQL	SQL	No SQL
Velocidad	Alta	Media	Alta
Rendimiento	Alto	Alto	Alto
Alcance	Proyectos Pequeños y Medianos	Proyectos grandes	Proyectos medianos y Grandes
Concurrencia	Media	Alta	Alta
Principios ACID	Atomicidad, consistencia y durabilidad de la información	Atomicidad, consistencia, integridad y durabilidad de la información	Atomicidad, consistencia, integridad y durabilidad de la información
Facilidad	Alta	Media	Media
Soporte de Archivos	Limitados tipos de archivos	Soporte Nativo para varios tipos de archivos	Soporte para varios tipos de archivos

Elaborado por: El investigador

3.2.4. Tecnologías para el desarrollo de la aplicación web progresiva

Tabla 14. Tecnologías para el desarrollo de la aplicación web progresiva

	Nombre	Funcionalidad	Descripción	Software
	MySQL	Base de datos	Los estándares, facilidad de manejo y la disponibilidad para su uso ya que, al ser de código abierto, y las normativas para las necesidades del sistema, convierten a MySQL en una herramienta muy eficaz para el manejo de la base de datos.	Libre
	Laravel	API REST	Para la comunicación entre la base de datos y el front-end con la composición y forma para el manejo de datos e información mediante el mapeo de objetos y enlace a la base de datos provee a la aplicación un nivel de seguridad que impide a atacantes la posibilidad de acceder a la información.	Libre
	Visual Studio Code	IDE	Herramienta de código abierto que permite trabajar con un editor de código en cualquier lenguaje de programación.	Libre
	Ionic	Framework de desarrollo	Framework basado en Angular permite la programación entre modelos estructurados para la vista y control bajo JavaScript, CSS y HTML para modelar la vista en diferentes dispositivos.	Libre

Elaborado por: El investigador.

3.2.4.1. Base de Datos

En la unidad de tecnologías de información y comunicación del distrito consideran y manejan MySQL como una herramienta eficaz y de gran utilidad en sus proyectos debido a su flexibilidad y facilidad al momento de crear un modelo relacional de base de datos. Por lo que el manejo y la implementación de herramientas de software libre son de gran ayuda para la creación de proyectos de investigación. Ya que se va a manejar información para la gestión de los archivos y el manejo de la aplicación se optó por MySQL para la gestión de la base de datos.

MySQL es un sistema gestor de base de datos relacional, lo que permite el manejo tanto de información individual como información que debe tener enlaces para poder acceder a otra información. La información a manejar presenta una estructura escalonada o de árbol ya que las vistas de acceso se manejan de forma sucesiva según las opciones que se seleccione.

3.2.4.2. Desarrollo de Servicios Web

Los servicios web para la aplicación web progresiva se los desarrolló mediante servicios REST bajo Laravel, ya que es una manera muy versátil de obtener la información que se solicita de la base de datos, además de poderla manipular de la manera que se necesite modelando las interfaces de presentación en archivos JSON o XML.

Se toma en cuenta de igual manera la estructura que maneja Laravel, ya que con su ORM (Object Relational Mapper), permite implementar seguridad a nivel de servidor ya que maneja métodos de acceso a los datos de una forma más segura.

Laravel cuenta de igual manera con bibliotecas para manejar el almacenamiento de archivos en su servidor, Storage permitió la manipulación de los archivos dentro del servidor. Con esta herramienta se desarrolló el modelo para la generación, manejo y gestión de los archivos que se van a guardar. También nos permite generar rutas para poder acceder a los archivos.

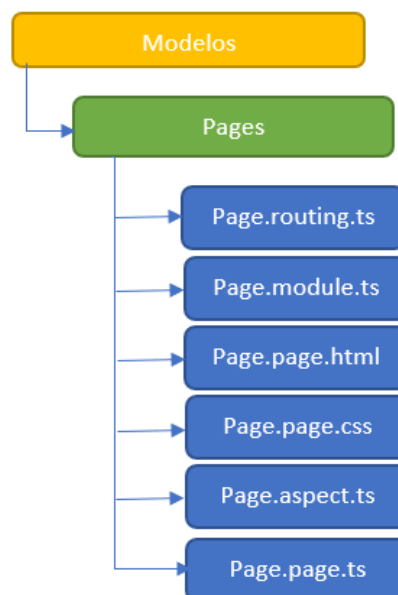
3.2.4.3. Ionic

Para el desarrollo de la aplicación web progresiva se optó por el IDE de Visual Studio Code ya que permite manejar Angular además de Ionic para el desarrollo del proyecto. Se decidió el framework de Ionic ya que su estructura y modelado para la interfaz permite generar vistas tanto para ordenador como para dispositivos móviles de una manera más rápido y flexibles para el usuario. Además, hay que tomar en cuenta que el paso de una aplicación generada en Ionic a una aplicación web progresiva se lo realiza de una manera rápida ya que solo basta con instalar y configurar los componentes para que pueda presentarse como tal.

Estructura

Para la generación del proyecto Ionic maneja una estructura de modelos y pages mediante las cuales se puede acceder tanto a las interfaces de programación como se muestra en la Ilustración 18.

Ilustración 18 Estructura Pages Ionic



Elaborado por: El Investigador.

En la Ilustración 18 se muestra `page.routing.ts` que se encarga del direccionamiento de las páginas, las rutas y enlaces para poder acceder, el `page.module.ts` que maneja tanto los

módulos como las librerías que utilizaran las páginas para su correcto funcionamiento, el `page.page.html` que se encarga de la vista y componentes que tendrá la página, el `page.page.css` que maneja los estilos que se implementen, el `page.aspect.ts` maneja la compilación de la página al momento de su ejecución y por último el `page.page.ts` que maneja toda la parte posterior de la página, en esta parte se incorpora todos los métodos, variables y validaciones que pueda tener la página.

3.2.5. Metodología para el desarrollo de la aplicación web progresiva

Tabla 15. Comparativa Metodologías

	Scrum	Kanban	XP
Tiempos	Sprint de 2 a 4 semanas	Ciclo corto y eficiente	Iteraciones basadas en estimaciones de una a tres semanas
Materiales	Entregables al final de cada sprint	Limite estricto de artículos en el flujo	Entregables al final de cada iteración
Trabajo	Divide el trabajo en bloque	Flujo de trabajo para dividir las tareas en artículos más pequeños	Retroalimentación continua
Flexibilidad	Mejora continua	No se empieza una tarea sin terminar dentro del flujo	Basada en flexibilidad, acepta cambios continuos
Equipo	Equipos autoorganizados	Tarjetas visuales	Programación en pares
Actores	<ul style="list-style-type: none"> • Producto Owner • Scrum Master • Equipo de desarrollo 	<ul style="list-style-type: none"> • Liderazgo en todos los niveles, evita definición de roles 	<ul style="list-style-type: none"> • Cliente • Programador • Coach • Tester

Elaborado por: El Investigador.

Para el desarrollo de la aplicación web progresiva se optó por el manejo de la metodología XP ya que se enfoca en el desarrollo de proyectos mediante fases, por lo que se adapta de mejor manera a las fases de planificación, diseño, codificación y pruebas que se van a realizar ya que es una metodología muy versátil en momentos que se necesita realizar un cambio o una nueva integración [28]. Además, hay que tomar en cuenta que XP maneja

un flujo constante de comunicación entre el usuario y el programador incluso en fases ya posteriores a las completadas.

Tabla 16. Características Metodología XP

Característica	Descripción
Comunicación	Se comenta el código para mejorar su entendimiento y lograr que su reutilización sea sencilla
Retroalimentación	El usuario debe estar inmerso en gran parte del proyecto ya que su opinión es clave.
Respeto	No se debe realizar cambios que interfieran en el progreso de otros procesos que puedan causar demora.
Simplicidad	Se busca simplificar todo lo que se pueda para agilizar los procesos.

Elaborado por: El Investigador.

Fases de XP

1. Planificación

En esta fase el objetivo principal es llevar a cabo la relación con el cliente y el equipo de trabajo, aquí se plantean los requerimientos definiendo los puntos clave para las historias de usuario, procesos a realizarse y las iteraciones estimadas [28].

2. Diseño

La segunda fase se enfoca en buscar el método más sencillo para acoplar las historias de usuario a los requerimientos, una de las alternativas más comunes y efectivas es el manejo de tarjetas CRC y prototipos de las interfaces para tener una visión del funcionamiento del proyecto.

3. Codificación

Aquí se definen y emplean los métodos de programación para tener una integración continua del código y realizar pruebas de funcionamiento constantes, con el fin de que los programadores puedan realizar cambios si se requiera y evitar tanto problemas como retrasos.

4. Pruebas

En esta etapa se prioriza las pruebas de funcionalidad para evitar errores o encontrar soluciones más óptimas y rápidas. Aquí se define el correcto funcionamiento de los proyectos además se estima el tiempo para las iteraciones y se analiza las historias de usuario.



Elaborado por: Fuente [28].

3.2.6. Desarrollo de la metodología

3.2.6.1. Planificación

La Unidad Distrital de Tecnologías de Información y Comunicaciones del Distrito 18D01-Ambato 1 Educación con el fin de obtener un repositorio digital para el manejo de documentación de unidades fiscales con una aplicación web progresiva, brindó todas las facilidades para realizar un levantamiento información y de esta manera obtener los requerimientos.

3.2.6.2. Definición de roles

Mediante la definición de roles se estableció las responsabilidades para el desarrollo del proyecto, tomando en cuenta los roles más prioritarios para la tecnología XP se designó las siguientes responsabilidades.

Tabla 17. Definición de Roles

Rol XP	Responsable	Descripción
Usuario	Analistas y Administradores de TI del distrito Rectores de Unidades Educativas Fiscales	Define requerimientos, especificaciones y provee información
Programador	Jonathan Tisalema	Convertir requerimientos en funcionalidades para el desarrollo del proyecto.
Tester	Ing. Bolívar Alcocer	Realizar las pruebas y verificar el proyecto.

Elaborado por: El Investigador.

3.2.6.3. Levantamiento de la Información

Conociendo que se cuenta con los recursos necesarios para el desarrollo de la aplicación web progresiva y determinando que el proyecto es factible, mediante la recolección de información que se realizó en base a una encuesta al departamento de TI del distrito el cual se enfoca a las formas con las que cuentan para recolectar la información y los requerimientos para flexibilidad de la aplicación. Y en base a la encuesta a los rectores de las unidades educativas fiscales que de igual forma se enfoca a la manera en la que accederán a la información con la aplicación. Para el desarrollo del proyecto se definió los siguientes módulos a desarrollar:

Ilustración 20 Módulos Definidos



Elaborado por: El Investigador.

3.2.6.4. Historias de Usuario

Para la representación de los requerimientos se manejó las historias de usuario para poder analizar de mejor manera el levantamiento de la información, estas historias de usuario se dividieron en actividades para establecer su nivel de complejidad.

Se utilizó el modelo de la Tabla 10 para el desarrollo de cada una de las historias de usuario.

Tabla 18. Modelo Historia de usuario

HISTORIA DE USUARIO	
Número:	Usuario:
Nombre historia:	
Prioridad en negocio:	Riesgo en el desarrollo:
Puntos estimados:	Iteración estimada:
Responsable:	
Descripción:	
Observación:	

Elaborado por: El Investigador.

Tabla 19. Descripción del modelo de historia de usuario

HISTORIA DE USUARIO	
Número: numeración para identificar la historia de usuario.	Usuario: persona a la que va dirigida la historia de usuario
Nombre historia: título	
Prioridad en negocio: se asigna según la necesidad con valores de Alto, Medio y Bajo.	Riesgo en el desarrollo: se asigna según el riesgo al momento de desarrollar la historia de usuario con valores Alto, Medio y Bajo.
Puntos estimados: se asigna el número de días para desarrollar la historia de usuario.	Iteración estimada: número de iteraciones para implementar la historia.
Responsable: programador responsable de cada historia.	
Descripción: se detalla la información obtenida por parte del cliente.	
Observación: se aclara dudas o información adicional a la historia de usuario.	

Elaborado por: El Investigador.

A continuación, se detalló el funcionamiento de la aplicación web progresiva, mediante las historias de usuario.

Tabla 20. Historia de usuario 01-Roles de Usuario

HISTORIA DE USUARIO	
Número: 01	Usuario: Todos
Nombre historia: Roles de Usuario	
Prioridad en negocio: Alto	Riesgo en el desarrollo: Medio
Puntos estimados: 1	Iteración estimada: 1
Responsable: Jonathan Tisalema	
Descripción: con el levantamiento de información se definió el nivel de acceso a la documentación que tendrán tanto administradores distritales, administradores educativos y rectores de las instituciones educativas.	
Observación: Los administradores distritales como los administradores educativos cuentan con permiso de visualizar y editar la información. Los rectores cuentan con permisos para solo visualizar la información.	

Elaborado por: El investigador

Tabla 21. Historia de usuario 02-Diseño Base de Datos

HISTORIA DE USUARIO	
Número: 02	Usuario: Desarrollador
Nombre historia: Diseño Base de Datos	
Prioridad en negocio: Alto	Riesgo en el desarrollo: Alto
Puntos estimados: 2	Iteración estimada: 1
Responsable: Jonathan Tisalema	
Descripción: un diseño tanto optimo como sencillo es de gran eficacia al modelado de la base de datos acorde a los requerimientos obtenidos.	
Observación:	

Elaborado por: El investigador.

Tabla 22. Historia de usuario 03-Arquitectura del proyecto

HISTORIA DE USUARIO	
Número: 03	Usuario: Desarrollador
Nombre historia: Arquitectura del proyecto	
Prioridad en negocio: Alto	Riesgo en el desarrollo: Medio
Puntos estimados: 1	Iteración estimada: 1
Responsable: Jonathan Tisalema	
Descripción: un buen modelo o arquitectura fomenta los puntos clave a realizarse o tomar en cuenta durante el desarrollo del proyecto.	
Observación:	

Elaborado por: El investigador.

Tabla 23. Historia de usuario 04-Pantalla de presentación para administradores distritales

HISTORIA DE USUARIO	
Número: 04	Usuario: Administradores distritales
Nombre historia: Pantalla de presentación para administradores distritales	
Prioridad en negocio: Alto	Riesgo en el desarrollo: Alto
Puntos estimados: 2	Iteración estimada: 1
Responsable: Jonathan Tisalema	
Descripción: pantalla exclusiva para los administradores distritales, se muestra el listado de las unidades educativas fiscales registradas en el sistema. Permite al administrador insertar, editar, eliminar unidades educativas si cumple con los requerimientos para eliminar (no contar con categorías o archivos en su interior).	
Observación: Puede acceder al posterior menú de selección de información al seleccionar una unidad educativa.	

Elaborado por: El investigador

Tabla 24. Historia de usuario 05-Pantalla menú unidad educativa

HISTORIA DE USUARIO	
Número: 05	Usuario: Todos
Nombre historia: Pantalla menú unidad educativa	
Prioridad en negocio: Medio	Riesgo en el desarrollo: Bajo
Puntos estimados: 1	Iteración estimada: 1
Responsable: Jonathan Tisalema	
Descripción: pantalla de presentación para las categorías a seleccionar para la documentación de la unidad educativa.	
Observación: Cuenta con las tres opciones disponibles para la documentación de la unidad educativa Boletas, Promociones o Documentación Institucional.	

Elaborado por: El investigador.

Tabla 25. Historia de usuario 06-Pantalla para periodos/años lectivos

HISTORIA DE USUARIO	
Número: 06	Usuario: Todos
Nombre historia: Pantalla para periodos/años lectivos	
Prioridad en negocio: Alto	Riesgo en el desarrollo: Alto
Puntos estimados: 2	Iteración estimada: 2
Responsable: Jonathan Tisalema	
Descripción: en la pantalla se visualiza los periodos lectivos registrados dentro de la unidad educativa. Los administradores pueden ingresar, editar o eliminar periodos lectivos que cumplan con los requerimientos para eliminar (no contar con niveles/cursos en su interior).	
Observación: Puede acceder al posterior menú de selección de información al seleccionar un periodo lectivo.	

Elaborado por: El investigador.

Tabla 26. Historia de usuario 07-Pantalla niveles/cursos

HISTORIA DE USUARIO	
Número: 07	Usuario: Todos
Nombre historia: Pantalla niveles/cursos	
Prioridad en negocio: Alto	Riesgo en el desarrollo: Alto
Puntos estimados: 2	Iteración estimada: 2
Responsable: Jonathan Tisalema	
Descripción: en la pantalla se visualiza los niveles con los que cuenta el periodo lectivo anteriormente seleccionado. Se debe tomar en cuenta que existen dos jornadas matutina y vespertina. Los administradores pueden ingresar, editar y eliminar los cursos que cumplan con los requerimientos para eliminar (no contar con paralelos en su interior).	
Observación: Puede acceder al posterior menú de selección de información al seleccionar un curso.	

Elaborado por: El investigador.

Tabla 27. Historia de usuario 08-Pantalla paralelos

HISTORIA DE USUARIO	
Número: 08	Usuario: Todos
Nombre historia: Pantalla paralelos	
Prioridad en negocio: Alto	Riesgo en el desarrollo: Alto
Puntos estimados: 2	Iteración estimada: 2
Responsable: Jonathan Tisalema	
Descripción: en la pantalla se visualiza los paralelos con los que cuenta el nivel o curso anteriormente seleccionado. Los administradores pueden ingresar, editar y eliminar los paralelos que cumplan con los requerimientos para eliminar (no contar con archivos en su interior).	
Observación: Puede acceder al posterior menú de selección de información al seleccionar un paralelo.	

Elaborado por: El investigador.

Tabla 28. Historia de usuario 09-Pantalla contenido paralelo

HISTORIA DE USUARIO	
Número: 09	Usuario: Todos
Nombre historia: Pantalla contenido paralelo.	
Prioridad en negocio: Alto	Riesgo en el desarrollo: Alto
Puntos estimados: 4	Iteración estimada: 2
Responsable: Jonathan Tisalema	
Descripción: en la pantalla se visualiza la documentación de los estudiantes pertenecientes al paralelo seleccionado. Los administradores pueden ingresar, eliminar, visualizar, descargar los archivos. Los rectores pueden solo visualizar y descargar la información que seleccionen.	
Observación: Tomar en cuenta que los archivos pueden ser Boletas, Promociones y Concentrado (Anexo A).	

Elaborado por: El investigador.

Tabla 29. Historia de usuario 10-Pantalla documentación institucional

HISTORIA DE USUARIO	
Número: 10	Usuario: Todos
Nombre historia: Pantalla documentación institucional.	
Prioridad en negocio: Alto	Riesgo en el desarrollo: Alto
Puntos estimados: 2	Iteración estimada: 2
Responsable: Jonathan Tisalema	
Descripción: en la pantalla se visualiza las categorías para la distribución de la documentación institucional. Los administradores pueden ingresar, editar y eliminar categorías que cumplan con los requerimientos para eliminar (no contar con archivos en su interior).	
Observación:	

Elaborado por: El investigador.

Tabla 30. Historia de usuario 11-Pantalla archivos documentación institucional

HISTORIA DE USUARIO	
Número: 11	Usuario: Todos
Nombre historia: Pantalla archivos documentación institucional.	
Prioridad en negocio: Alto	Riesgo en el desarrollo: Alto
Puntos estimados: 2	Iteración estimada: 3
Responsable: Jonathan Tisalema	
Descripción: en la pantalla se visualiza los archivos pertenecientes a la categoría seleccionada.	
Observación: Los administradores pueden ingresar, eliminar, visualizar, descargar los archivos. Los rectores pueden solo visualizar y descargar la información que seleccionen.	

Elaborado por: El investigador.

Tabla 31. Historia de usuario 12-Pantalla presentación administradores institucionales

HISTORIA DE USUARIO	
Número: 12	Usuario: Todos
Nombre historia: Pantalla presentación administradores institucionales.	
Prioridad en negocio: Medio	Riesgo en el desarrollo: Medio
Puntos estimados: 1	Iteración estimada: 3
Responsable: Jonathan Tisalema	
Descripción: en la pantalla se visualiza el logotipo de la unidad educativa, nombre y dirección.	
Observación: Puede acceder al posterior menú de selección de información al seleccionar la unidad educativa.	

Elaborado por: El investigador.

Tabla 32. Historia de usuario 13-Buscador de Archivos

HISTORIA DE USUARIO	
Número: 13	Usuario: Todos
Nombre historia: Buscador de Archivos.	
Prioridad en negocio: Alto	Riesgo en el desarrollo: Alto
Puntos estimados: 3	Iteración estimada: 3
Responsable: Jonathan Tisalema	
Descripción: en la pantalla puede seleccionar las opciones para realizar la búsqueda rápida del archivo que se necesite a través de: el tipo de archivo que se busque, Boletas, Promociones, Concentrado o Documentos Institucionales, la unidad educativa, periodo lectivo, jornada, nivel o curso, paralelo, estudiante.	
Observación: Tomar en cuenta que los administradores institucionales y los rectores solo pueden tener acceso a la información de su unidad educativa.	

Elaborado por: El investigador.

Tabla 33. Historia de usuario 14-Gestión de Usuarios

HISTORIA DE USUARIO	
Número: 14	Usuario: Administradores distritales
Nombre historia: Gestión de Usuarios.	
Prioridad en negocio: Alto	Riesgo en el desarrollo: Alto
Puntos estimados: 2	Iteración estimada: 3
Responsable: Jonathan Tisalema	
Descripción: en la pantalla se visualiza la información de los usuarios registrados	
Observación: se puede ingresar nuevos usuarios con un rol específico para el acceso a la información. Editar y eliminar usuarios.	

Elaborado por: El investigador.

Tabla 34. Historia de usuario 15-Reportes

HISTORIA DE USUARIO	
Número: 15	Usuario: Todos
Nombre historia: Reportes.	
Prioridad en negocio: Alto	Riesgo en el desarrollo: Alto
Puntos estimados: 2	Iteración estimada: 4
Responsable: Jonathan Tisalema	
Descripción: en la pantalla se puede seleccionar las opciones para obtener el reporte de los archivos almacenados según la categoría reporte de archivos de estudiantes en un nivel, paralelo o de los documentos institucionales almacenados.	
Observación: Tomar en cuenta que los administradores institucionales y los rectores solo pueden tener acceso a la información de su unidad educativa.	

Elaborado por: El investigador.

Tabla 35. Historia de usuario 16-Noticias/Notificaciones

HISTORIA DE USUARIO	
Número: 16	Usuario: Administradores distritales
Nombre historia: Noticias/Notificaciones.	
Prioridad en negocio: Alto	Riesgo en el desarrollo: Alto
Puntos estimados: 2	Iteración estimada: 4
Responsable: Jonathan Tisalema	
Descripción: en la pantalla se visualiza el listado de noticias que se van a presentar en la pantalla de inicio al sistema. Se puede ingresar nuevas noticias y también eliminar.	
Observación: consta de un título general, una descripción y una imagen.	

Elaborado por: El investigador.

Tabla 36. Historia de usuario 17-Pantalla inicio

HISTORIA DE USUARIO	
Número: 17	Usuario: Todos
Nombre historia: Pantalla inicio.	
Prioridad en negocio: Medio	Riesgo en el desarrollo: Medio
Puntos estimados: 1	Iteración estimada: 4
Responsable: Jonathan Tisalema	
Descripción: en la pantalla se visualiza las noticias o notificaciones referentes a los eventos o información pertinente que sea de interés a las unidades educativas.	
Observación: se visualiza de manera lateral las noticias o eventos registrados.	

Elaborado por: El investigador.

Tabla 37. Historia de usuario 18- Acceso a la aplicación

HISTORIA DE USUARIO	
Número: 18	Usuario: Todos
Nombre historia: Acceso a la aplicación	
Prioridad en negocio: Alto	Riesgo en el desarrollo: Medio
Puntos estimados: 4	Iteración estimada: 4
Responsable: Jonathan Tisalema	
Descripción: pantalla para el ingreso a la aplicación.	
Observación: se validará el ingreso de los usuarios mediante el usuario (correo electrónico) y su contraseña, además del acceso a la información según el rol de usuario.	

Elaborado por: El investigador.

3.2.6.5. Tareas

Historia: Grupos de usuarios

Tabla 38. Tarea 1- Historia 1

TAREA	
Número: 1	Número historia: 1
Nombre tarea: Definición de los roles de usuario	
Tipo de tarea: Análisis	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
<p>Descripción: Presentar los roles que pueden tener los usuarios y el nivel de acceso que tienen a la documentación:</p> <ul style="list-style-type: none"> ✓ Administrador: rol de acceso para el personal de la unidad del distrito, tiene acceso a todo el sistema con la posibilidad de ingresar, editar y eliminar información. ✓ Administrador_UE: rol de acceso para los encargados en las diferentes unidades educativas, tienen acceso a toda la información referente a su unidad educativa con la posibilidad de ingresar, editar y eliminar información. ✓ Rector: rol de acceso para los rectores de las unidades educativas, tiene acceso a toda la información referente a su unidad educativa para poder visualizarla y descargarla. 	

Elaborado por: El investigador.

Historia: Diseño Base de Datos

Tabla 39. Tarea 1- Historia 2

TAREA	
Número: 2	Número historia: 2
Nombre tarea: Diseño Base de Datos	
Tipo de tarea: Análisis y diseño	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: plantear el modelado relacional para la base de datos según los requerimientos para su posterior revisión y aprobación.	

Elaborado por: El investigador.

Tabla 40. Tarea 2- Historia 2

TAREA	
Número: 3	Número historia: 2
Nombre tarea: Conexión Back-End	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: establecer la conexión con Laravel, se construye los modelos y se prueba la conexión a la base de datos.	

Elaborado por: El investigador.

Historia: Arquitectura del proyecto

Tabla 41. Tarea 1- Historia 3

TAREA	
Número: 4	Número historia: 3
Nombre tarea: Análisis y definición de la estructura del proyecto	
Tipo de tarea: Análisis	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: realizar un análisis de las herramientas tanto físicas como digitales disponibles para la elaboración del proyecto. Además de tomar en cuenta el tipo de proyecto que se va a generar.	

Elaborado por: El investigador.

Historia: Pantalla de presentación para administradores distritales

Tabla 42. Tarea 1- Historia 4

TAREA	
Número: 5	Número historia: 4
Nombre tarea: Ingreso de información unidades educativas	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none">1. Construcción de los servicios en el Back-End para poder insertar, editar y eliminar la información de las unidades educativas y poder consumir mediante un api en el Front-End.2. Diseño de la interfaz para el ingreso de la información de las unidades educativas.3. Diseño de la interfaz para la selección de unidades educativas.4. Validaciones para el ingreso de información, edición y eliminar.	

Elaborado por: El investigador.

Historia: Pantalla menú unidad educativa

Tabla 43. Tarea 1- Historia 5

TAREA	
Número: 6	Número historia: 5
Nombre tarea: Diseño interfaz para el menú de las unidades educativas	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: Diseño de la interfaz para la selección de los diferentes tipos de búsqueda de información ya sean Boletas, Promociones o Documentación Institucional.	

Elaborado por: El investigador.

Historia: Pantalla para periodos/años lectivos

Tabla 44. Tarea 1- Historia 6

TAREA	
Número: 7	Número historia: 6
Nombre tarea: Diseño interfaz para el ingreso de periodos lectivos.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none">1. Construcción de los servicios en el Back-End para poder insertar, editar y eliminar la información de los periodos lectivos y poder consumir mediante un api en el Front-End.2. Diseño de la interfaz para el ingreso de periodos lectivos de las unidades educativas.	

Elaborado por: El investigador.

Tabla 45. Tarea 2- Historia 6

TAREA	
Número: 8	Número historia:6
Nombre tarea: Diseño interfaz para la selección del periodo lectivo.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none">1. Diseño de la interfaz para la selección del periodo lectivo.2. Validaciones para el ingreso de información, edición y eliminar.	

Elaborado por: El investigador.

Historia: Pantalla niveles/cursos

Tabla 46. Tarea 1- Historia 7

TAREA	
Número: 9	Número historia: 7
Nombre tarea: Diseño interfaz para el ingreso de niveles/cursos.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none">1. Construcción de los servicios en el Back-End para poder insertar, editar y eliminar la información de los niveles/cursos y poder consumir mediante un api en el Front-End.2. Diseño de la interfaz para el ingreso de niveles/cursos de los periodos en las unidades educativas.	

Elaborado por: El investigador.

Tabla 47. Tarea 2- Historia 7

TAREA	
Número: 10	Número historia:7
Nombre tarea: Diseño interfaz para la selección del curso/periodo.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none">1. Diseño de la interfaz para la selección del nivel/curso.2. Validaciones para el ingreso de información, edición y eliminar.	

Elaborado por: El investigador.

Historia: Pantalla paralelos

Tabla 48. Tarea 1- Historia 8

TAREA	
Número: 11	Número historia: 8
Nombre tarea: Diseño interfaz para el ingreso de paralelos.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none">1. Construcción de los servicios en el Back-End para poder insertar, editar y eliminar la información de los paralelos y poder consumir mediante un api en el Front-End.2. Diseño de la interfaz para el ingreso de paralelos en los cursos de las unidades educativas.	

Elaborado por: El investigador.

Tabla 49. Tarea 2- Historia 8

TAREA	
Número: 12	Número historia:8
Nombre tarea: Diseño interfaz para la selección del curso/periodo.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none">1. Diseño de la interfaz para la selección del paralelo según la jornada (matutina o vespertina).2. Validaciones para el ingreso de información, edición y eliminar.	

Elaborado por: El investigador.

Historia: Pantalla contenido paralelo

Tabla 50. Tarea 1- Historia 9

TAREA	
Número: 13	Número historia: 9
Nombre tarea: Diseño interfaz para el ingreso de archivos del paralelo según el tipo de documento (boleta, promoción, concentrado).	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none">1. Construcción de los servicios en el Back-End para poder insertar y eliminar los archivos de los paralelos y poder consumir mediante un api en el Front-End.2. Diseño de la interfaz para el ingreso de archivos en los paralelos de las unidades educativas.	

Elaborado por: El investigador.

Tabla 51. Tarea 2- Historia 9

TAREA	
Número: 14	Número historia:9
Nombre tarea: Desarrollo métodos subir, descargar archivos.	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none">1. Desarrollo del método para subir los archivos al servidor.2. Desarrollo del método para visualizar los archivos del servidor.3. Desarrollo del método para descargar los archivos del servidor.	

Elaborado por: El investigador.

Tabla 52. Tarea 3- Historia 9

TAREA	
Número: 15	Número historia:9
Nombre tarea: Diseño interfaz para la selección del archivo.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción:	
<ol style="list-style-type: none"> 1. Diseño de la interfaz para la selección del archivo referente al contenido del paralelo. 2. Validaciones para el ingreso de archivos, eliminar, visualizar y descargar. 	

Elaborado por: El investigador.

Historia: Pantalla documentación institucional.

Tabla 53. Tarea 1- Historia 10

TAREA	
Número: 16	Número historia: 8
Nombre tarea: Diseño interfaz para el ingreso de categorías de archivos institucionales.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción:	
<ol style="list-style-type: none"> 1. Construcción de los servicios en el Back-End para poder insertar, editar y eliminar la información de las categorías y poder consumir mediante un api en el Front-End. 2. Diseño de la interfaz para el ingreso de categorías en la documentación de las unidades educativas. 	

Elaborado por: El investigador.

Tabla 54. Tarea 2- Historia 10

TAREA	
Número: 17	Número historia:8
Nombre tarea: Diseño interfaz para la selección de las categorías de archivos institucionales.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción:	
<ol style="list-style-type: none"> 1. Diseño de la interfaz para la selección de la categoría para acceder a los archivos de la unidad educativa. 2. Validaciones para el ingreso de información, edición y eliminar. 	

Elaborado por: El investigador.

Historia: Pantalla archivos documentación institucional.

Tabla 55. Tarea 1- Historia 11

TAREA	
Número: 18	Número historia: 11
Nombre tarea: Diseño interfaz para el ingreso de archivos en la documentación institucional.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción:	
<ol style="list-style-type: none"> 1. Construcción de los servicios en el Back-End para poder insertar y eliminar los archivos de la documentación y poder consumir mediante un api en el Front-End. 2. Diseño de la interfaz para el ingreso de archivos en las categorías de la documentación de las unidades educativas. 	

Elaborado por: El investigador.

Tabla 56. Tarea 2- Historia 11

TAREA	
Número: 19	Número historia:11
Nombre tarea: Desarrollo métodos subir, descargar archivos.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none">1. Desarrollo del método para subir los archivos al servidor.2. Desarrollo del método para visualizar los archivos del servidor.3. Desarrollo del método para descargar los archivos del servidor.	

Elaborado por: El investigador.

Tabla 57. Tarea 3- Historia 11

TAREA	
Número: 20	Número historia:11
Nombre tarea: Diseño interfaz para la selección del archivo.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none">1. Diseño de la interfaz para la selección del archivo referente al contenido de la categoría.2. Validaciones para el ingreso de archivos, eliminar, visualizar y descargar.	

Elaborado por: El investigador.

Historia: Pantalla presentación administradores institucionales

Tabla 58. Tarea 1- Historia 12

TAREA	
Número: 21	Número historia: 12
Nombre tarea: Diseño interfaz de presentación para los administradores institucionales	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: Diseño de la interfaz de presentación para los administradores institucionales, al presionar el logotipo accede al menú de la documentación de la unidad educativa.	

Elaborado por: El investigador.

Historia: Buscador de Archivos

Tabla 59. Tarea 1- Historia 13

TAREA	
Número: 22	Número historia: 13
Nombre tarea: Diseño de la interfaz con las opciones para la búsqueda de archivos.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none">1. Definir el modelo de búsqueda para la interfaz de documentos de estudiantes: documento a elegir, institución, año lectivo, jornada, grado nivel, paralelo, estudiantes.2. Definir el modelo de búsqueda para la interfaz de documentos institucionales: documento a elegir, institución, año lectivo, tipo de documento, archivo.3. Desarrollo de la interfaz.	

Elaborado por: El investigador.

Tabla 60. Tarea 2- Historia 13

TAREA	
Número: 23	Número historia:13
Nombre tarea: Desarrollo servicios para la búsqueda de información y validación.	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Programador responsable: Jonathan Tisalema	
Descripción:	
<ol style="list-style-type: none"> 1. Desarrollo de los servicios para el filtrado de información para la búsqueda de documentos de estudiantes. 2. Desarrollo de los servicios para el filtrado de información para la búsqueda de documentos de estudiantes. 3. Validaciones en la interfaz. 	

Elaborado por: El investigador.

Historia: Gestión de Usuarios

Tabla 61. Tarea 1- Historia 14

TAREA	
Número: 24	Número historia: 14
Nombre tarea: Diseño de la interfaz para el ingreso de usuarios.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción:	
<ol style="list-style-type: none"> 1. Construcción de los servicios en el Back-End para poder insertar, editar y eliminar los usuarios y los roles para el acceso al sistema. 2. Diseño de la interfaz para el ingreso de usuarios. 	

Elaborado por: El investigador.

Tabla 62. Tarea 2- Historia 14

TAREA	
Número: 25	Número historia: 14
Nombre tarea: Desarrollo de la interfaz gestión de usuarios.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción:	
<ol style="list-style-type: none"> 1. Desarrollo de la interfaz para la gestión de los usuarios. 2. Validaciones para el ingreso de archivos, eliminar, visualizar y descargar. 	

Elaborado por: El investigador.

Historia: Reportes

Tabla 63. Tarea 1- Historia 15

TAREA	
Número: 26	Número historia: 15
Nombre tarea: Diseño de la interfaz con las opciones para los reportes sugeridos.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción:	
<ol style="list-style-type: none"> 1. Definir el modelo para el reporte de archivos de estudiantes según el nivel seleccionado: reporte a elegir, institución, año lectivo, jornada, grado o nivel. 2. Definir el modelo para el reporte de archivos de estudiantes según el paralelo seleccionado: reporte a elegir, institución, año lectivo, jornada, grado o nivel, paralelo. 3. Definir el modelo para el reporte de archivos institucionales: reporte a elegir, institución, año lectivo. 4. Diseño de la interfaz. 	

Elaborado por: El investigador.

Tabla 64. Tarea 2- Historia 15

TAREA	
Número: 28	Número historia: 15
Nombre tarea: Desarrollo servicios para la búsqueda de información y validación.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none"> 1. Desarrollo de los servicios para el filtrado de información para la generación del reporte de estudiantes según el nivel. 2. Desarrollo de los servicios para el filtrado de información para la generación del reporte de estudiantes según el paralelo. 3. Desarrollo de los servicios para el filtrado de información para la generación del reporte de archivos institucionales. 4. Validaciones en la interfaz. 	

Elaborado por: El investigador.

Historia: Noticias/Notificaciones.

Tabla 65. Tarea 1- Historia 16

TAREA	
Número: 29	Número historia: 16
Nombre tarea: Diseño interfaz para el ingreso de noticias/notificaciones.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none"> 1. Construcción de los servicios en el Back-End para poder insertar y eliminar las noticias que se quieran postear en el inicio de la aplicación. 2. Diseño de la interfaz para el ingreso noticias y eventos para la comunidad educativa. 	

Elaborado por: El investigador.

Tabla 66. Tarea 2- Historia 16

TAREA	
Número: 30	Número historia: 16
Nombre tarea: Diseño interfaz para la gestión de noticias.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción:	
<ol style="list-style-type: none"> 1. Diseño de la interfaz para la gestión de noticias. 2. Validaciones para el ingreso de información y eliminar. 	

Elaborado por: El investigador.

Historia: Pantalla Inicio.

Tabla 67. Tarea 1- Historia 17

TAREA	
Número: 31	Número historia: 17
Nombre tarea: Diseño de interfaz inicio y presentación de noticias.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: Diseño de la interfaz para el inicio del sistema y comunicar noticias, eventos a la comunidad educativa de manera horizontal.	

Elaborado por: El investigador.

Historia: Acceso a la aplicación.

Tabla 68. Tarea 1- Historia 18

TAREA	
Número: 32	Número historia: 18
Nombre tarea: Diseño de la interfaz para el ingreso al sistema y validación.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none">1. Diseño de la interfaz para el ingreso.2. Validación de las credenciales del usuario.3. Almacenamiento del rol de usuario y su token de acceso.	

Elaborado por: El investigador.

Tabla 69. Tarea 2- Historia 18

TAREA	
Número: 33	Número historia: 18
Nombre tarea: Diseño de los menús de navegación según el rol de usuario.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none">1. Diseño del modelo de navegación para rectores, administradores institucionales y administradores distritales.	

Elaborado por: El investigador.

Tabla 70. Tarea 3- Historia 18

TAREA	
Número: 34	Número historia: 18
Nombre tarea: Enrutamiento y control de acceso a la aplicación según el rol de usuario.	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Programador responsable: Jonathan Tisalema	
Descripción: <ol style="list-style-type: none">1. Definir el nivel de acceso que tienen los rectores a la información y la navegación en el sistema.2. Definir el nivel de acceso que tienen los Administradores institucionales a la información y la navegación en el sistema.3. Definir el nivel de acceso que tienen los Administradores distritales a la información y la navegación en el sistema.	

Elaborado por: El investigador.

3.2.6.6. Evaluación de las historias de usuario

Al haber analizado las historias de usuario con sus tiempos estimados en efectuar las actividades propuestas, a continuación, se muestra la estimación tentativa para los tiempos de resolución de cada actividad y su prioridad al realizarla. Tomando en cuenta que un día laboral son 6 horas de trabajo.

Tabla 71. Evaluación Historias de usuario

N°.	Nombre de la Historia de Usuario	Prioridad	Días	Horas
01	Roles de Usuario	Alta	1	6
02	Diseño Base de Datos	Alta	2	12
03	Arquitectura del Proyecto	Alta	1	6
04	Pantalla presentación para administradores distritales	Alta	2	12
05	Pantalla menú unidad educativa	Media	1	6
06	Pantalla para periodos/años lectivos	Alta	2	12
07	Pantalla niveles/cursos	Alta	2	12
08	Pantalla paralelos	Alta	2	12
09	Pantalla contenido paralelo	Alta	4	24
10	Pantalla documentación institucional	Alta	2	12
11	Pantalla archivos documentación institucional	Alta	2	12
12	Pantalla presentación administradores institucionales	Media	1	6
13	Buscador de Archivos	Alto	3	18
14	Gestión de Usuarios	Alto	2	12
15	Reportes	Alto	2	12
16	Noticias/Notificaciones	Alto	2	12
17	Pantalla Inicio	Medio	1	6

18	Acceso a la aplicación	Alto	4	14
		Total	36	216

Elaborado por: El Investigador.

3.2.6.7. Plan de Entrega

Tabla 72. Plan de Entrega

N°	Historia de usuario	Tiempo estimado		Iteración asignada				Entrega asignada			
		Días	Horas	1	2	3	4	1	2	3	4
01	Roles de Usuario	1	6	x				x			
02	Diseño Base de Datos	2	12	x				x			
03	Arquitectura del Proyecto	1	6	x				x			
04	Pantalla presentación para administradores distritales	2	12	x				x			
05	Pantalla menú unidad educativa	1	6	x				x			
06	Pantalla para periodos/años lectivos	2	12		x				x		
07	Pantalla niveles/cursos	2	12		x				x		
08	Pantalla paralelos	2	12		x				x		
09	Pantalla contenido paralelo	4	24		x				x		
10	Pantalla documentación institucional	2	12		x				x		
11	Pantalla archivos documentación institucional	2	12			x				x	
12	Pantalla presentación administradores institucionales	1	6			x				x	
13	Buscador de Archivos	3	18			x				x	

14	Gestión de Usuarios	2	12			x				x	
15	Reportes	2	12				x				x
16	Noticias/Notificaciones	2	12				x				x
17	Pantalla Inicio	1	6				x				x
18	Acceso a la aplicación	4	14				x				x

Elaborado por: El Investigador.

3.2.6.8. Plan de iteraciones

A continuación, se muestra el estado de desarrollo de las historias de usuario, con su conclusión en la fase de pruebas de aceptación:

Tabla 73. Plan de Iteraciones

Iteración	N°.	Nombre de la Historia de Usuario	Prioridad	Estado de Desarrollo	Pruebas de Aceptación
1	01	Roles de Usuario	Alta	Completado	Aprobado
	02	Diseño Base de Datos	Alta	Completado	Aprobado
	03	Arquitectura del Proyecto	Alta	Completado	Aprobado
	04	Pantalla presentación para administradores distritales	Alta	Completado	Aprobado
	05	Pantalla menú unidad educativa	Media	Completado	Aprobado
2	06	Pantalla para periodos/años lectivos	Alta	Completado	Aprobado
	07	Pantalla niveles/cursos	Alta	Completado	Aprobado

	08	Pantalla paralelos	Alta	Completado	Aprobado
	09	Pantalla contenido paralelo	Alta	Completado	Aprobado
	10	Pantalla documentación institucional	Alta	Completado	Aprobado
3	11	Pantalla archivos documentación institucional	Alta	Completado	Aprobado
	12	Pantalla presentación administradores institucionales	Media	Completado	Aprobado
	13	Buscador de Archivos	Alto	Completado	Aprobado
	14	Gestión de Usuarios	Alto	Completado	Aprobado
4	15	Reportes	Alto	Completado	Aprobado
	16	Noticias/Notificaciones	Alto	Completado	Aprobado
	17	Pantalla Inicio	Medio	Completado	Aprobado
	18	Acceso a la aplicación	Alto	Completado	Aprobado

Elaborado por: El investigador.

3.2.6.9. Diseño

3.2.6.9.1. Metáfora

El repositorio digital permite tanto a los empleados del departamento del distrito como a los delegados de las unidades educativas, gestionar la documentación y archivos tanto de estudiantes como de la institución en cualquier lugar y cualquier momento. Además, permite a los rectores de las unidades educativas acceder a la información de su institución teniendo la posibilidad de visualizar y descargarla.

La aplicación se divide en módulos que manejan sus propias funciones y actividades para manejar el correcto funcionamiento de manera ordenada, a continuación, se presentan los módulos implementados:

1. **Modulo Unidades Educativas:** este módulo cuenta con todas las características para la gestión de información de unidades educativas y su información consecutiva para el acceso a los diferentes periodos lectivos, cursos, paralelos, estudiantes. Además, en el apartado de documentación institucional debe gestionar tanto el ingreso de categorías para la información y la documentación o archivos que se vayan a almacenar.
2. **Modulo Buscador:** este módulo cuenta con las opciones necesarias para poder realizar las búsquedas de información en las diferentes categorías que se ingresen en la aplicación como periodos, cursos, paralelos, documentación. Con la posibilidad de visualizar el documento seleccionado y poder descargarlo.
3. **Modulo Reportes:** este módulo cuenta con las opciones necesarias para poder obtener el reporte de los archivos almacenados en el repositorio realizando el filtrado ya sea para el reporte de archivos de estudiantes en un nivel o curso, en un paralelo, o de igual manera obtener el reporte de los archivos institucionales. Con la posibilidad de visualizar el documento seleccionado y poder descargarlo.
4. **Modulo Notificaciones:** este módulo permite la gestión de la información que concierna a la comunidad educativa ya sean eventos o noticias, para posteriormente visualizarla en la pantalla de inicio del sistema.
5. **Modulo Usuarios:** en este módulo se puede ingresar los usuarios que van a acceder al sistema mediante su correo electrónico se generara su usuario y se asignara una contraseña para su validación, además se asigna el rol de acceso con el que contara para visualizar la información. Además, valida todo el enrutamiento y acceso a los diferentes módulos.

3.2.6.9.2. Tarjetas CRC

- Modulo Unidades Educativas

Tabla 74. Tarjeta CRC- Ingreso nueva Unidad Educativa

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Obtener información ingresada <ul style="list-style-type: none"> ✓ Nombre ✓ Dirección ✓ Logotipo Validar Información Registro de la nueva unidad educativa Actualización listada de unidades educativas	Datos de las unidades educativas. Métodos de integración al Back-End.
Observaciones:	

Elaborado por: El investigador.

Tabla 75. Tarjeta CRC- Presentación Unidades Educativas

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Visualizar información Unidades Educativas Buscador para el filtrado de unidades educativas	Servicios de acceso a datos.
Observaciones:	

Elaborado por: El investigador.

Tabla 76. Tarjeta CRC- Gestión de Unidades Educativas

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores

Permitir la edición de la información de las unidades educativas ingresadas <ul style="list-style-type: none"> ✓ Nombre ✓ Dirección ✓ Logotipo Permitir la opción de eliminar una unidad educativa que no contenga categorías (periodos o documentación).	Servicios de acceso a datos. Métodos de integración al Back-End.
Observaciones:	

Elaborado por: El investigador.

Tabla 77. Tarjeta CRC- Ingreso nuevo Periodo Lectivo

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Obtener información ingresada <ul style="list-style-type: none"> ✓ Nombre ✓ Unidad Educativa a la que pertenece Validar Información Registro de un nuevo periodo según la unidad educativa seleccionada Actualización listada de periodos según la unidad educativa seleccionada	Información del periodo lectivo y la unidad educativa seleccionada. Métodos de integración al Back-End.
Observaciones:	

Elaborado por: El investigador.

Tabla 78. Tarjeta CRC- Presentación Periodos Lectivos

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores

Visualizar información de periodos lectivos. Buscador para el filtrado de periodos lectivos.	Servicios de acceso a datos.
Observaciones:	

Elaborado por: El investigador.

Tabla 79. Tarjeta CRC- Gestión de periodos lectivos

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Permitir la edición de la información de los periodos lectivos <ul style="list-style-type: none"> ✓ Nombre Permitir la opción de eliminar un periodo lectivo que no contenga categorías (cursos o niveles).	Servicios de acceso a datos. Métodos de integración al Back-End.
Observaciones:	

Elaborado por: El investigador.

Tabla 80. Tarjeta CRC- Ingreso nueva categoría Documentos Institucionales

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Obtener información ingresada <ul style="list-style-type: none"> ✓ Nombre Validar Información Registro de una nueva categoría para guardar documentación según la unidad educativa seleccionada. Actualización listada de categorías para documentación según la unidad educativa seleccionada	información de la categoría. Métodos de integración al Back-End.
Observaciones:	

Elaborado por: El investigador.

Tabla 81. Tarjeta CRC- Presentación Categorías de Documentación

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores

Visualizar información de las categorías. Buscador para el filtrado de categorías.	Servicios de acceso a datos.
Observaciones:	

Elaborado por: El investigador.

Tabla 82. Tarjeta CRC- Gestión de categorías

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Permitir la edición de la información de las categorías ✓ Nombre Permitir la opción de eliminar categoría que no contenga archivos.	Servicios de acceso a datos. Métodos de integración al Back-End.
Observaciones:	

Elaborado por: El investigador.

Tabla 83. Tarjeta CRC- Ingreso nuevo Curso o Nivel

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Obtener información ingresada ✓ Nombre ✓ Periodo lectivo al que Pertenece Validar Información Registro de un nuevo curso o nivel según el periodo seleccionado y la jornada a la que corresponde. Actualización listada de cursos o niveles según el periodo seleccionado y la jornada a la que corresponde.	Información del curso o nivel. Métodos de integración al Back-End.
Observaciones:	

Elaborado por: El investigador.

Tabla 84. Tarjeta CRC- Presentación Cursos o Niveles

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores

Visualizar información de cursos o niveles según la jornada. Buscador para el filtrado de cursos o niveles.	Servicios de acceso a datos.
Observaciones:	

Elaborado por: El investigador.

Tabla 85. Tarjeta CRC- Gestión de Cursos o Niveles

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Permitir la edición de la información de los cursos o niveles ingresados.	Servicios de acceso a datos.
Permitir la opción de eliminar curso o nivel que no contengan paralelos.	Métodos de integración al Back-End.
Observaciones:	

Elaborado por: El investigador.

Tabla 86. Tarjeta CRC- Ingreso nuevo Paralelo

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Obtener información ingresada <ul style="list-style-type: none"> ✓ Nombre ✓ Curso o nivel al que pertenece Validar Información Registro de un nuevo paralelo según el curso seleccionado. Actualización listada de paralelos según el curso que se ha seleccionado.	Información del paralelo. Métodos de integración al Back-End.
Observaciones:	

Elaborado por: El investigador.

Tabla 87. Tarjeta CRC- Presentación Paralelos

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores

Visualizar información de los paralelos. Buscador para el filtrado de paralelos.	Servicios de acceso a datos.
Observaciones:	

Elaborado por: El investigador.

Tabla 88. Tarjeta CRC- Gestión de paralelos

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Permitir la edición de la información de paralelos(nombre). Permitir la opción de eliminar paralelos que no contengan boletas, promociones o concentrado.	Servicios de acceso a datos. Métodos de integración al Back-End.
Observaciones:	

Elaborado por: El investigador.

Tabla 89. Tarjeta CRC- Ingreso Archivos Estudiantes

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Obtener los archivos ingresados (individual o múltiples) según la categoría seleccionada para el almacenamiento. <ul style="list-style-type: none"> ✓ Boletas ✓ Promociones ✓ Concentrado. Almacenar los archivos en el servidor y su información en la base de datos. Actualización listada de paralelos según el curso que se ha seleccionado.	Archivos de los estudiantes pertenecientes al paralelo (boletas, promociones y concentrado). Métodos de integración al Back-End.
Observaciones:	

Elaborado por: El investigador.

Tabla 90. Tarjeta CRC- Presentación Archivos

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores

Visualizar archivos de los estudiantes. Buscador para el filtrado de archivos.	Servicios de acceso a datos.
Observaciones:	

Elaborado por: El investigador.

Tabla 91. Tarjeta CRC- Gestión de archivos

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Permitir las opciones para visualización y descarga de los archivos.	Servicios de acceso a los archivos. Métodos de integración al Back-End.
Observaciones:	

Elaborado por: El investigador.

- Modulo Buscador

Tabla 92. Tarjeta CRC- Presentación Opciones

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Realizar los inputs para la selección de la información y realizar el filtrado (tipo de documento, unidad educativa, periodo lectivo, jornada, curso o nivel, paralelo y documento). Enlazar los inputs para cargar la información en cascada.	Servicios de acceso a datos.
Observaciones:	

Elaborado por: El investigador.

Tabla 93. Tarjeta CRC- Descarga y visualización de archivos

Ingreso nueva Unidad Educativa

Responsabilidades	Colaboradores
Desarrollo de la interfaz para la búsqueda de los archivos y presentar los botones para la visualización y descarga.	Servicios de acceso a datos.
Observaciones:	

Elaborado por: El investigador.

- Modulo Reportes

Tabla 94. Tarjeta CRC- Presentación Opciones

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Realizar los inputs para la selección de la información y realizar el filtrado (tipo de documento, unidad educativa, periodo lectivo, jornada, curso o nivel, paralel). Enlazar los inputs para cargar la información en cascada.	Servicios de acceso a datos.
Observaciones:	

Elaborado por: El investigador.

Tabla 95. Tarjeta CRC- Descarga y visualización de reporte

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Desarrollo de la interfaz para la generación del reporte y presentar los botones para la visualización y descarga.	Servicios de acceso a datos.
Observaciones:	

Elaborado por: El investigador.

- Modulo Notificaciones

Tabla 96. Tarjeta CRC- Ingreso de Noticias/eventos

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Obtener la información ingresada para presentar las noticias <ul style="list-style-type: none"> ✓ Título ✓ Descripción ✓ Imagen. Registro de la nueva noticia. Actualización listada de noticias.	Información de la noticia o evento. Métodos de integración al Back-End.
Observaciones:	

Elaborado por: El investigador.

Tabla 97. Tarjeta CRC- Presentación Noticias/eventos

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Desarrollo de la interfaz para la presentación de las noticias o eventos de forma horizontal.	Servicios de acceso a datos.
Observaciones:	

Elaborado por: El investigador.

Tabla 98. Tarjeta CRC- Gestión de noticias

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Visualizar archivos de las noticias. Eliminar una noticia seleccionada. Buscador para el filtrado de noticias.	Servicios de acceso a los archivos.
Observaciones:	

Elaborado por: El investigador.

- Modulo Usuarios

Tabla 99. Tarjeta CRC- Ingreso de Usuarios

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Obtener la información ingresada para los nuevos usuarios <ul style="list-style-type: none"> ✓ Nombre ✓ Correo ✓ Contraseña ✓ Rol de usuario ✓ Unidad educativa para administradores institucionales y rectores Registro del nuevo usuario. Actualización listada de usuarios.	Información del nuevo usuario. Métodos de integración al Back-End.
Observaciones:	

Elaborado por: El investigador.

Tabla 100. Tarjeta CRC- Presentación del listado de usuarios

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Visualizar los usuarios. Buscador para el filtrado de usuarios.	Servicios de acceso a datos.
Observaciones:	

Elaborado por: El investigador.

Tabla 101. Tarjeta CRC- Gestión de usuarios

Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Permitir las opciones para la actualización e información de los usuarios existentes en el sistema y también la posibilidad de eliminar.	Servicios de acceso a los archivos. Métodos de integración al Back-End.
Observaciones:	

Elaborado por: El investigador.

Tabla 102. Tarjeta CRC- Validación de Usuarios

Ingreso nueva Unidad Educativa

Responsabilidades	Colaboradores
Validar el correcto ingreso de los usuarios. Generar el token de acceso al sistema. Almacenar los datos de usuarios.	Servicios de acceso a los archivos.
Observaciones:	

Elaborado por: El investigador.

Tabla 103. Tarjeta CRC- Enrutamiento de Roles

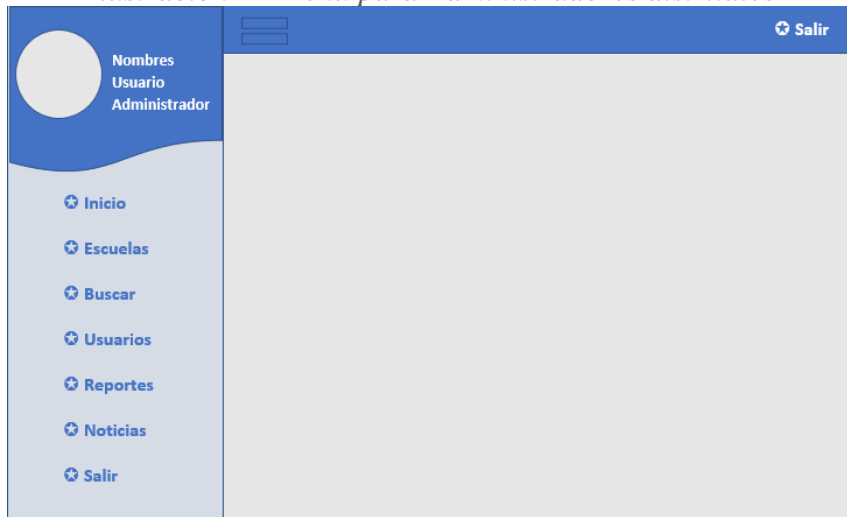
Ingreso nueva Unidad Educativa	
Responsabilidades	Colaboradores
Restringir el acceso según el rol de usuario, para proteger la información del sistema. Administradores distritales: acceso a todo el sistema Administradores institucionales: acceso a toda la información referente a su unidad educativa. Rectores: acceso para visualizar y descargar la información referente a su unidad educativa.	Servicios de acceso a los archivos. Modelo de enrutamiento Ionic.
Observaciones:	

Elaborado por: El investigador.

3.2.6.9.3. Diseño de Interfaces

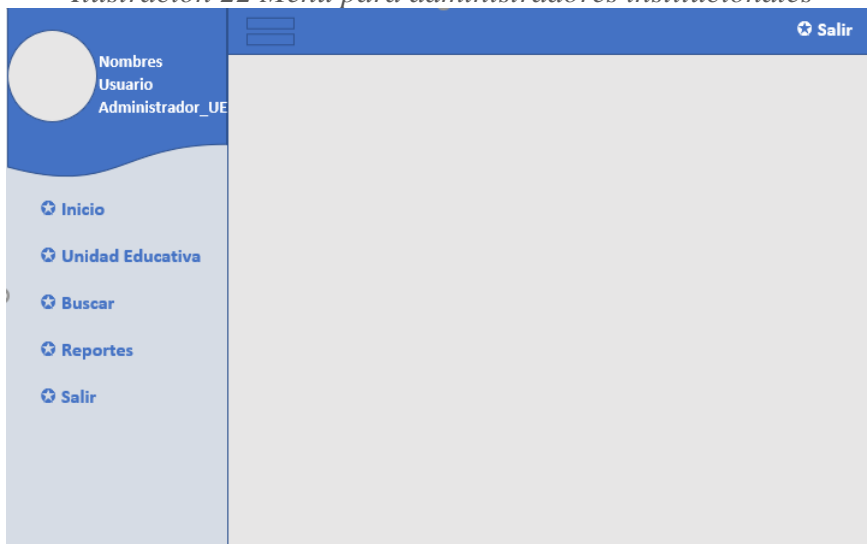
Menús para los usuarios.

Ilustración 21 Menú para Administradores distritales



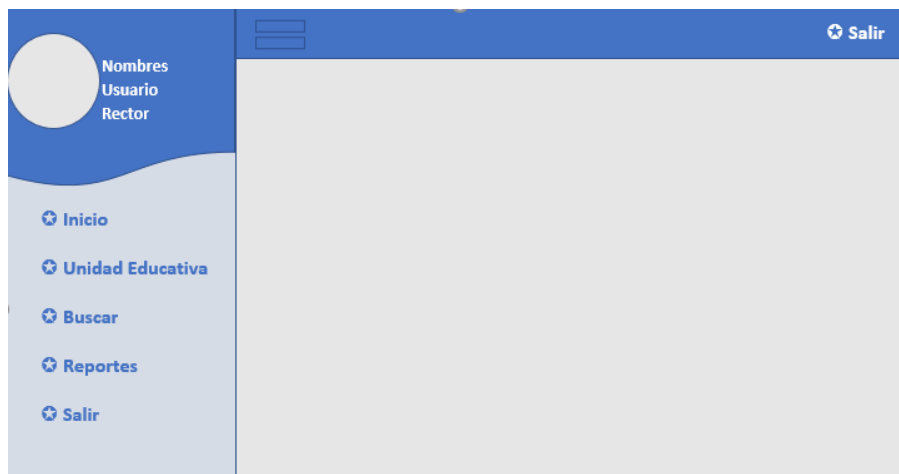
Elaborado por: El Investigador.

Ilustración 22 Menú para administradores institucionales



Elaborado por: El Investigador.

Ilustración 23 Menú para Rectores



Elaborado por: El Investigador.

Pantallas de inicio.

Ilustración 24 Pantalla inicio unidades educativas para administradores distritales



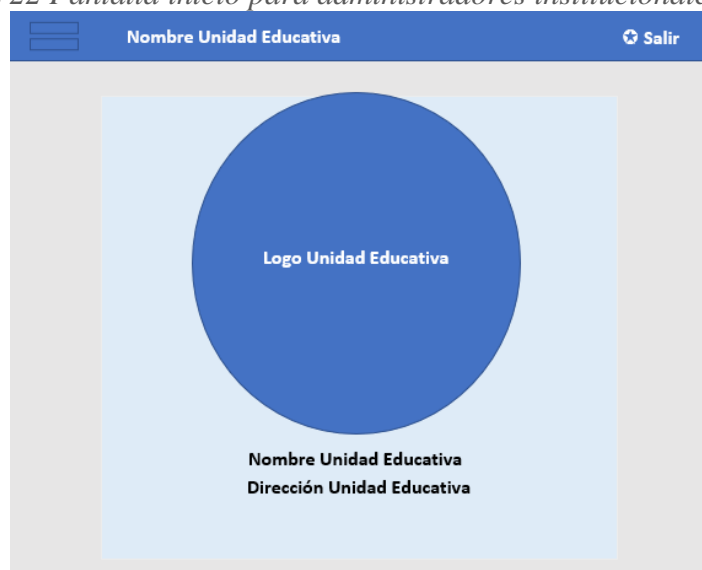
Elaborado por: El Investigador.

Ilustración 25 Pantalla para el ingreso de nueva unidad educativa

The screenshot shows a web application interface with a blue header bar. On the left of the header is a hamburger menu icon, and on the right is the text 'Listado Unidades Educativas' and a 'Salir' button with a circular arrow icon. Below the header is a light blue box containing a form. At the top of this box is a blue button labeled 'Agregar Unidad Educativa'. Below it are three input fields: 'Nombre de la unidad educativa', 'Dirección Unidad Educativa', and 'Seleccionar Logotipo'. At the bottom of the form is a blue 'Guardar' button.

Elaborado por: El Investigador.

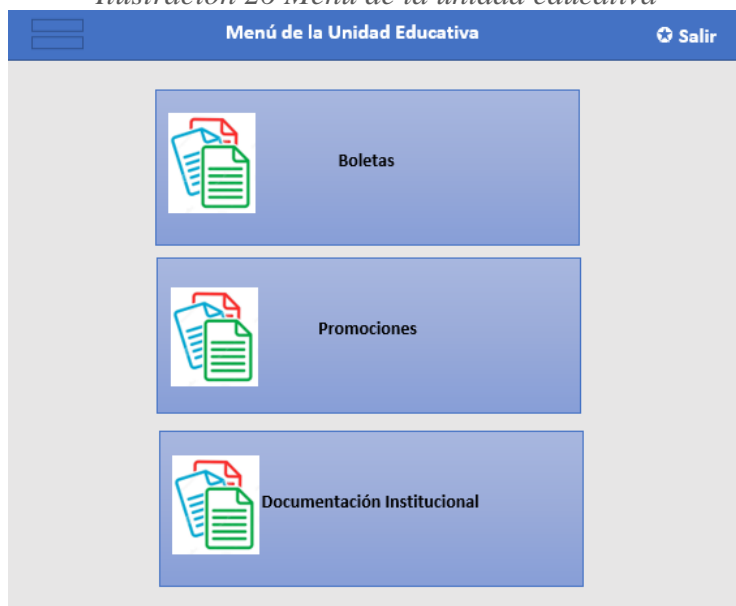
Ilustración 22 Pantalla inicio para administradores institucionales y rectores.



Elaborado por: El Investigador.

Menú para la selección del tipo de documentación.

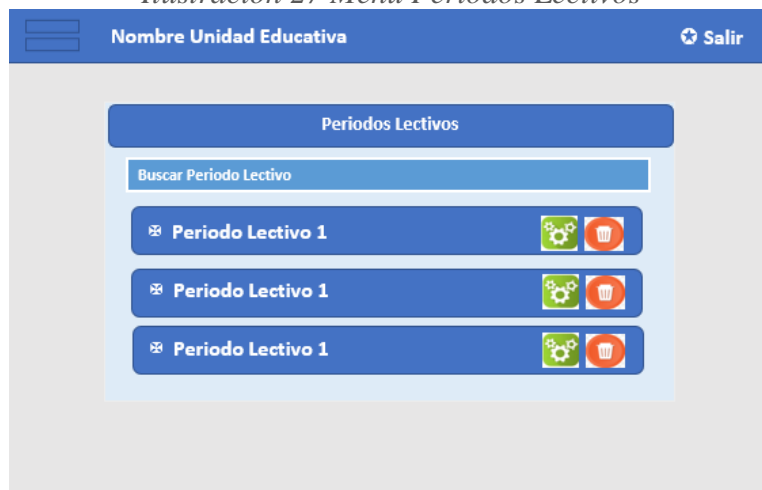
Ilustración 26 Menú de la unidad educativa



Elaborado por: El Investigador.

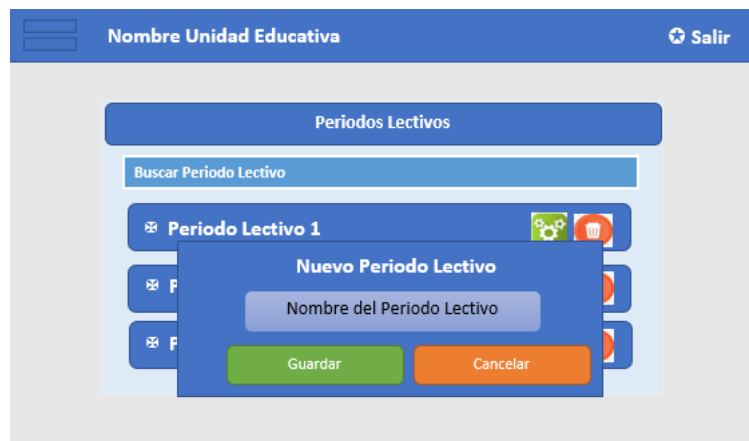
Menú para los periodos lectivos.

Ilustración 27 Menú Periodos Lectivos



Elaborado por: El Investigador.

Ilustración 28 Ingreso nuevo periodo lectivo



Elaborado por: El Investigador.

Menú Niveles o Cursos.

Ilustración 29 Menú niveles



Elaborado por: El Investigador.

Ilustración 30 Ingreso nuevo nivel según la jornada seleccionada.



Elaborado por: El Investigador.

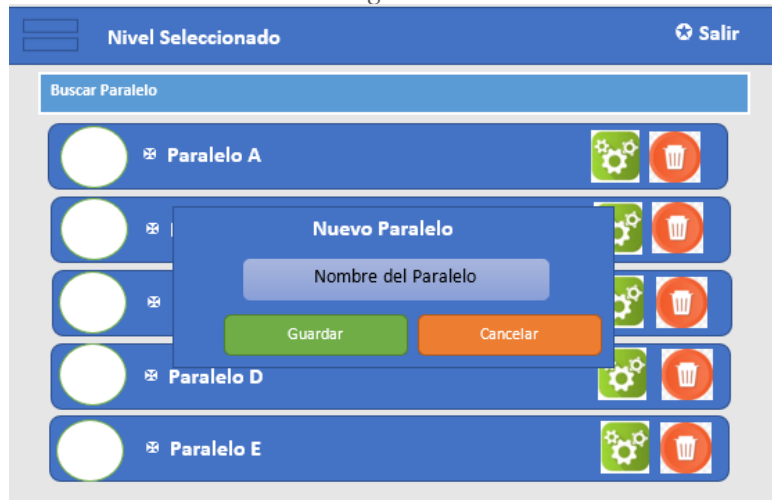
Menú Paralelos.

Ilustración 31 Menú Paralelos.



Elaborado por: El Investigador.

Ilustración 32 Ingreso nuevo Paralelo



Elaborado por: El Investigador.

Menú contenido del paralelo.

Ilustración 33 Menú contenido del paralelo.



Elaborado por: El Investigador.

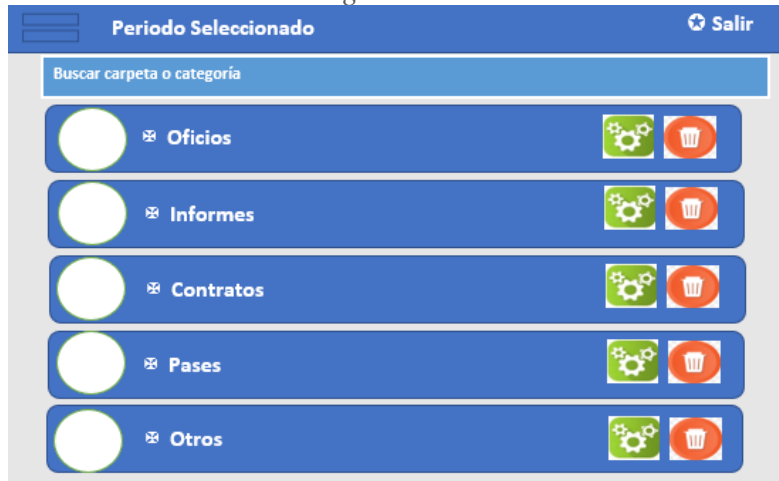
Ilustración 34 Ingreso archivos al paralelo.



Elaborado por: El Investigador.

Menú documentación institucional.

Ilustración 35 Menú Categorías documentos institucionales.



Elaborado por: El Investigador.

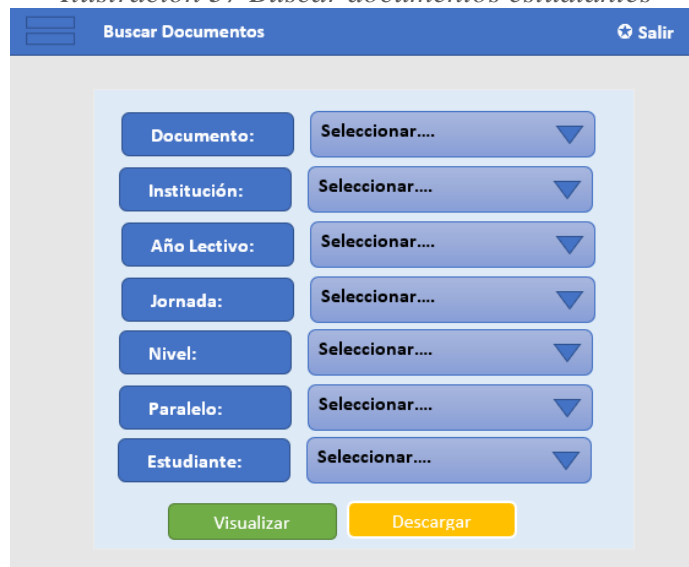
Ilustración 36 Ingreso Archivos institucionales.



Elaborado por: El Investigador.

Buscador de Documentos.

Ilustración 37 Buscar documentos estudiantes



Elaborado por: El Investigador.

Ilustración 38 Buscar documentos institucionales

Buscar Documentos Salir

Documento: Seleccionar...

Institución: Seleccionar...

Año Lectivo: Seleccionar...

Categoria: Seleccionar...

Archivo: Seleccionar...

Visualizar Descargar

Elaborado por: El Investigador.

Gestión de Usuarios.

Ilustración 39 Gestión de Usuarios

Gestión de Usuarios Salir

Buscar Usuario

Nombre	Correo	Institución	Rol	
Usuario1	User1@gmail.co	Institución1	Admin	
Usuario2	User2@gmail.co	Institución2	AdminUE	
Usuario3	User3@gmail.co	Institución3	Rector	

Elaborado por: El Investigador.

Ilustración 40 Agregar nuevo usuario.



Elaborado por: El Investigador.

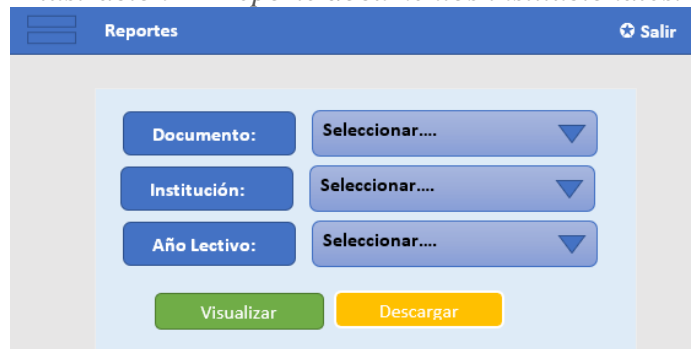
Reportes.

Ilustración 41 Reporte de archivos de estudiantes.



Elaborado por: El Investigador.

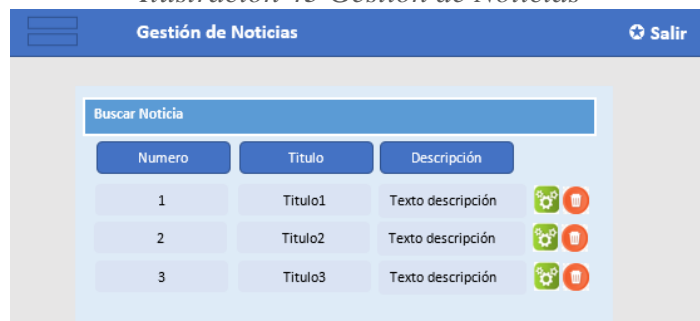
Ilustración 42 Reporte documentos institucionales.



Elaborado por: El Investigador.

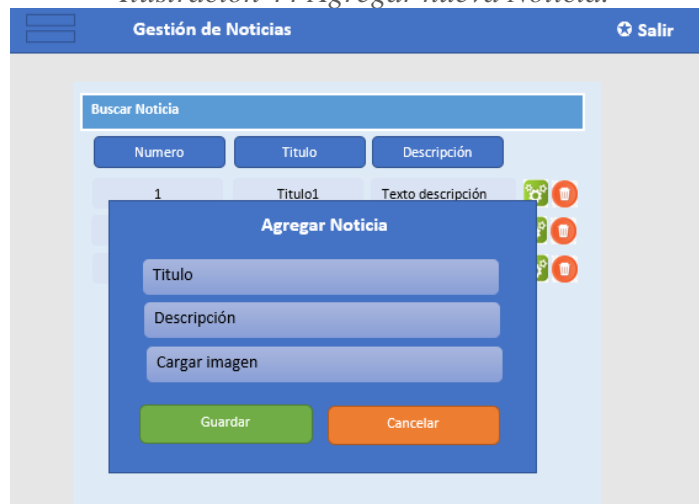
Noticias, eventos o notificaciones.

Ilustración 43 Gestión de Noticias



Elaborado por: El Investigador.

Ilustración 44 Agregar nueva Noticia.



Elaborado por: El Investigador.

Ilustración 45 Pantalla de Inicio.



Elaborado por: El Investigador.

Ilustración 46 Ingreso al Sistema.



Elaborado por: El Investigador.

3.2.6.10. Codificación

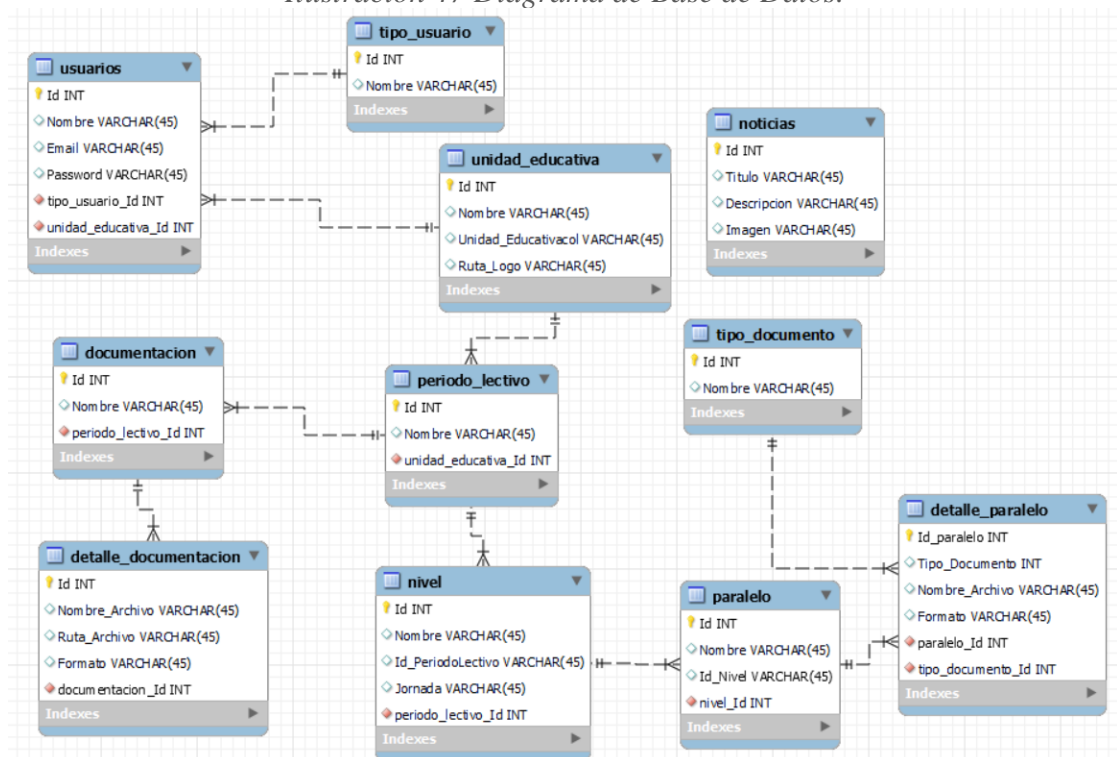
A continuación, se presenta el código fuente para el desarrollo de la aplicación. Se detalló los métodos tanto para el funcionamiento del Back-End con la base de datos, como el Front-End para la presentación y funcionamiento.

3.2.6.10.1. Codificación Back-End

Modelado de la Base de Datos

Para el diseño y modelado de la base de datos se ocupó la herramienta workbench, la cual es una herramienta visual que nos permite la generación de la base de datos de una manera fácil y rápida, además de poder generar la base de datos nos permite administrarla.

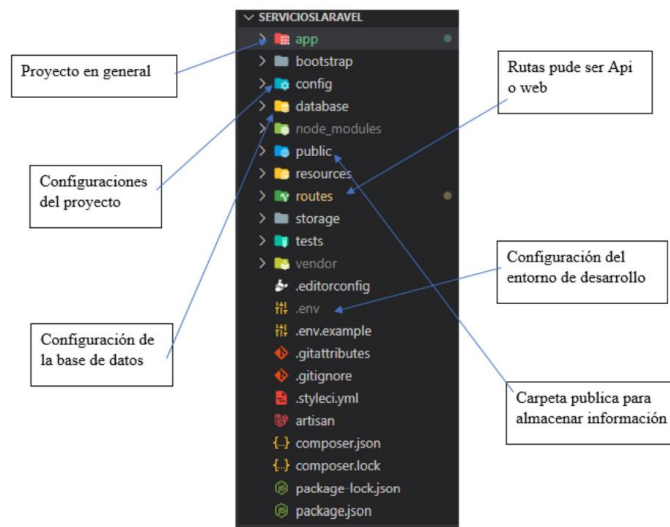
Ilustración 47 Diagrama de Base de Datos.



Elaborado por: El Investigador.

Laravel es una herramienta que trabaja bajo MVC (Modelo vista de controlador) que facilita en gran medida el desarrollo de proyectos que funcionen de Back-End para el manejo de API's.

Ilustración 48 Estructura Laravel.



Elaborado por: El Investigador.

Configuración del MySQL con Laravel.

Ilustración 49 Configuración MySQL-Laravel.

```
'mysql' => [
    'driver' => 'mysql',
    'url' => env('DATABASE_URL'),
    'host' => env('DB_HOST', '127.0.0.1'),
    'port' => env('DB_PORT', '3306'),
    'database' => env('DB_DATABASE', 'forge'),
    'username' => env('DB_USERNAME', 'forge'),
    'password' => env('DB_PASSWORD', ''),
    'unix_socket' => env('DB_SOCKET', ''),
    'charset' => 'utf8mb4',
    'collation' => 'utf8mb4_unicode_ci',
    'prefix' => '',
    'prefix_indexes' => true,
    'strict' => true,
    'engine' => null,
    'options' => extension_loaded('pdo_mysql') ? array_filter([
        PDO::MYSQL_ATTR_SSL_CA => env('MYSQL_ATTR_SSL_CA'),
    ]) : [],
],
```

Elaborado por: El Investigador.

Ilustración 50 Conexión a la base de datos.

```

APP_NAME=Laravel
APP_ENV=local
APP_KEY=base64:H5N1IRFJ3x8A2wdwGA/oa12FOQETe1r26wJLiefEG4=
APP_DEBUG=true
APP_URL=http://localhost

LOG_CHANNEL=stack
LOG_LEVEL=debug

DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=repodistrito
DB_USERNAME=root
DB_PASSWORD=

BROADCAST_DRIVER=log
CACHE_DRIVER=file
QUEUE_CONNECTION=sync
SESSION_DRIVER=file
SESSION_LIFETIME=120

```

Elaborado por: El Investigador.

Creación y configuración de los modelos para la gestión de las tablas en la base de datos.

Ilustración 51 Modelo conexión tablas base de datos.

```

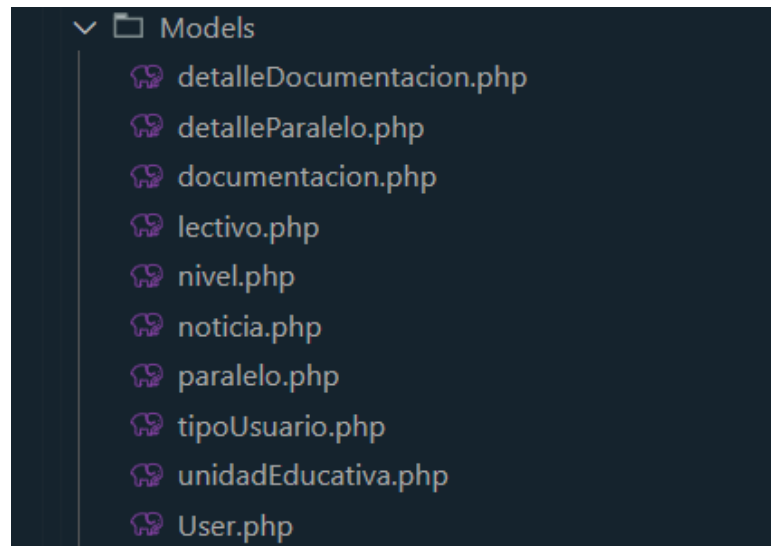
namespace App\Models;
use Tymon\JWTAuth\Contracts\JWTSubject;
use Illuminate\Contracts\Auth\MustVerifyEmail;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
use Illuminate\Foundation\Auth\User as Authenticatable;
use Illuminate\Notifications\Notifiable;

class usuario extends Model
{
    use HasFactory;
    public $timestamps = false;
    protected $table='users';
    protected $fillable = [
        'id',
        'nombre',
        'id_unidad_educativa',
        'email',
        'password',
        'tipo_usuario'
    ];
}

```

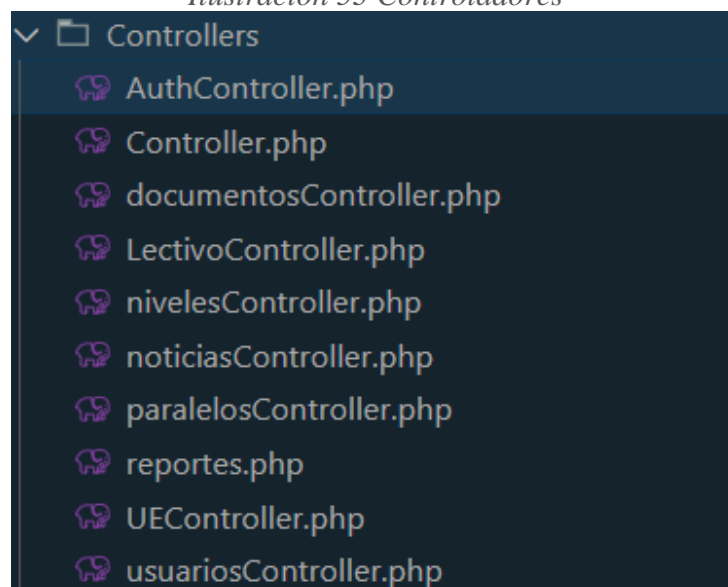
Elaborado por: El Investigador.

Ilustración 52 Modelos para la gestión de los datos.



Elaborado por: El Investigador.

Ilustración 53 Controladores



Elaborado por: El Investigador.

Configuración de las rutas para las API.

Ilustración 54 Rutas para consulta API

```
Route::middleware('auth:api')->get('/user', function (Request $request) {
    return $request->user();
});
///usuarios
Route::group([
    'prefix' => 'auth'
], function ($router) {
    Route::post('logout', [AuthController::class,"logout"]);
});
///unidades educativas services
Route::post('insertarUnidad',[UEController::class,"insertarUnidad"]);
Route::get('getUnidades',[UEController::class,"getUnidades"]);
Route::get('getUnidad/{id}',[UEController::class,"getUnidad"]);
Route::delete('eliminarUnidad/{id}',[UEController::class,"eliminarUnidad"]);
Route::post('editarUnidad',[UEController::class,"editarUnidad"]);
//años Lectivos
Route::get('getLectivos/{id}',[LectivoController::class,"getLectivos"]);
Route::post('insertarLectivo',[LectivoController::class,"insertarLectivo"]);
Route::post('eliminarLectivo',[LectivoController::class,"eliminarLectivo"]);
Route::post('editarLectivo',[LectivoController::class,"editarLectivo"]);
//documentacion
Route::get('getDocs/{id}',[documentosController::class,"getDocumentacionbyLectivo"]);
Route::post('insertarCarpeta',[documentosController::class,"insertarCarpeta"]);
Route::post('eliminarCarpeta',[documentosController::class,"eliminarCarpeta2"]);
Route::put('cambiar',[documentosController::class,"cambiar"]);
Route::post('editarCarpeta',[documentosController::class,"editarCarpeta"]);
```

Elaborado por: El Investigador.

Ilustración 55 Rutas para consulta API

```
//archivos
Route::post('subirArchivo',[documentosController::class,"subirArchivo"]);
Route::get('getArchivos/{id}',[documentosController::class,"getArchivosbyDoc"]);
Route::post('descargarArchivo',[documentosController::class,"descargarArchivo"]);
Route::post('borrarArchivo',[documentosController::class,"eliminarArchivo2"]);
//niveles
Route::post('insertarNivel',[nivelesController::class,"insertarNivel"]);
Route::get('getNiveles/{id}',[nivelesController::class,"getNiveles"]);
Route::delete('borrarNivel/{id}',[nivelesController::class,"borrarNivel"]);
Route::post('editarNivel',[nivelesController::class,"editarNivel"]);
//PARALELOS
Route::get('getParalelos/{id}',[paralelosController::class,"getParalelos"]);
Route::post('subirParalelo',[paralelosController::class,"subirParalelo"]);
Route::delete('borrarParalelo/{id}',[paralelosController::class,"borrarParalelo"]);
Route::post('editarParalelo',[paralelosController::class,"editarParalelo"]);
//paralelos archivos
Route::post('subirArchivoParalelo',[paralelosController::class,"subirArchivo"]);
Route::get('getArchivosParalelo/{id}',[paralelosController::class,"getArchivos"]);
Route::post('descargarArchivoParalelo',[paralelosController::class,"descargarArchivo"]);
Route::delete('borrarArchivoParalelo',[paralelosController::class,"borrarArchivo"]);
//noticias
Route::post('subirNoticia',[noticiasController::class,"subirNoticia"]);
Route::get('getNoticias',[noticiasController::class,"getNoticias"]);
Route::delete('eliminarNoticia/{id}',[noticiasController::class,"eliminarNoticia"]);
//reportes
Route::post('reporteDocumentos/{id}',[reportes::class,"reporteDocumentos"]);
Route::get('reporteNivel/{id}',[reportes::class,"reporteNivel"]);
Route::get('reporteParalelo/{id}',[reportes::class,"reporteParalelo"]);
```

Elaborado por: El Investigador.

Métodos para la Unidad Educativa.

Ilustración 56 Ingresar Unidad Educativa.

```
//Insertar nueva unidad educativa
public function insertarUnidad(Request $request){
    $unidad = new unidadEducativa();
    $unidad->nombre = $request->input('nombre');
    $unidad->direccion = $request->input('direccion');
    $archivo = $request->file('archivo');
    //comprobar si existe otra unidad educativa con ese nombre
    $unidadExiste = unidadEducativa::where([
        ['NOMBRE','=', $request->input('nombre')]]->get();
    if(count($unidadExiste) > 0){
        //si no hay unidades con ese nombre inserto
        $ruta = $archivo->storeAs('unidadesEducativas/logos', $unidad->nombre . '.' . $archivo->getClientOriginalExtension(), $unidad->ruta_logo);
        $unidad->ruta_logo = $ruta;
        $unidad->save();
        Storage::makeDirectory('public/unidadesEducativas/'.$unidad->nombre);
        return response()->json([
            [
                'Unidad Educativa' => $unidad,
                'HttpResponse' => [
                    'title' => 'Correcto',
                    'message' => 'Nueva Unidad Educativa creada!',
                    'status' => 200,
                    'statusText' => 'success',
                    'ok' => true
                ],
            ],
        ],
    );
    }else{
        return response()->json([
            'HttpResponse' => [
                'message' => 'Unidad Educativa ya existente!: '.$request->input('nombre'),
                'status' => 400,
                'statusText' => 'error',
                'ok' => true
            ],
        ],
    );
    }
}
```

Elaborado por: El Investigador.

Ilustración 57 Eliminar Unidad Educativa

```
//eliminar unidad educativa
public function eliminarUnidad($idUnidad){
    //obtengo el objeto de la base para eliminar
    $unidad = unidadEducativa::where('ID','=', $idUnidad);
    // obtengo el nombre de la unidad
    $nombres = unidadEducativa::where('ID','=', $idUnidad)
    ->select('unidad_educativa.NOMBRE as nombreUnidad')
    ->get();
    //verifico que la unidad educativa no tenga años lectivos
    $count = Lectivo::where('ID_Unidad_Educativa', '=', $idUnidad)->count();
    if($count > 0){
        return response()->json([
            'HttpResponse' => [
                'title' => 'Error',
                'message' => 'No puede eliminar Unidades Edicativas con Años Lectivos!',
                'status' => 400,
                'statusText' => 'error',
                'ok' => true
            ],
        ],
    );
    }else{
        try {
            Storage::deleteDirectory('public/unidadesEducativas/'.$nombres[0]->nombreUnidad);
            //elimino el logo de la unidad educativa
            $res = Storage::disk('public')->delete('unidadesEducativas/logos/'.$nombres[0]->nombreUnidad.'.png');
            if($res != 1){
                $res = Storage::disk('public')->delete('unidadesEducativas/logos/'.$nombres[0]->nombreUnidad.'.jpg');
            }
            $unidad->delete();
            return response()->json([
                'HttpResponse' => [
                    'title' => 'Correcto',
                    'message' => 'Unidad Educativa eliminada!',
                    'status' => 200,
                    'statusText' => 'success',
                    'ok' => true
                ],
            ],
        );
    }
}
```

Elaborado por: El Investigador.

Ilustración 58 Actualizar unidad Educativa.

```

//actualizar unidad educativa
public function editarUnidad(Request $request){
    $idUnidad=$request->input('idUnidad');
    $nombreUnidad=$request->input('nombre');
    $direccion=$request->input('direccion');
    $archivo=$request->file('archivo');

    $lastUnidad=unidadEducativa::where('ID','=', $idUnidad)->get();
    try{
        //nombre
        if($nombreUnidad!=0){
            $unidad=unidadEducativa::where('ID','=', $idUnidad)
            ->update(['NOMBRE'=>$nombreUnidad]);
        }
        //direccion
        if($direccion!=0){
            $unidad=unidadEducativa::where('ID','=', $idUnidad)
            ->update(['DIRECCION'=>$direccion]);
        }
        //rutaLogo
        if($archivo!=null){
            //guardo la imagen y la nueva ruta
            $ruta=$archivo->storeAs('unidadesEducativas/logos', $nombreUnidad . '.' . $archivo->getClientOriginalExtens
            $unidad=unidadEducativa::where('ID','=', $idUnidad)
            ->update(['RUTA_LOGO'=>$ruta]);
            //elimino el logo de la unidad educativa
            $res=Storage::disk('public')->delete('unidadesEducativas/logos/' . $lastUnidad[0]->NOMBRE . '.png');
            if($res!=1){
                $res=Storage::disk('public')->delete('unidadesEducativas/logos/' . $lastUnidad[0]->NOMBRE . '.jpg');
            }
            //actualizo el nombre de la carpeta
            Storage::move('public/unidadesEducativas/' . $lastUnidad[0]->NOMBRE,
            'public/unidadesEducativas/' . $nombreUnidad);
        }
    }
    return response()->json(

```

Elaborado por: El Investigador.

Ilustración 59 Obtener listado de Unidades Educativas.

```

//obtener lista de unidades educativas
public function getUnidades(){
    function getImagenes($unidades){
        $array=[];
        foreach($unidades as $unidades){
            $object = (object)['ID' => $unidades->ID, 'URL' => asset('storage/' . $unidades->RUTA_LOGO), 'NOMBRE'
            'DIRECCION' => $unidades->DIRECCION];
            array_push($array, $object);
        }
        return $array;
    }
    $unidades = unidadEducativa::all();
    return response()->json(
        [
            //unidades => $unidades,
            'unidades' => getImagenes($unidades),
            'HttpResponse' => [
                'status' => 200,
                'statusText' => 'OK',
                'ok' => true
            ]
        ],
    );
}

```

Elaborado por: El Investigador.

Ilustración 60 Obtener Unidad Educativa por Id.

```
//obtener unidad educativa por id
public function getUnidad($id){
    $unidad=unidadEducativa::find($id);
    function getImagenes($unidades){
        $object = (object)['ID' => $unidades->ID, 'URL' => asset('storage/'.$unidades->RUTA_LOGO), 'NOMBRE'
        'DIRECCION' => $unidades->DIRECCION];
        return $object;
    }
    return response()->json(
    [
        ['unidades' => $unidades,
        'unidad' => getImagenes($unidad),
        'HttpResponse' => [
            'status' => 200,
            'statusText' => 'OK',
            'ok' => true
        ]
        ],
    );
}
```

Elaborado por: El Investigador.

Usuarios e inicio de sesión.

Ilustración 61 Registrar usuario.

```
//ingresar un nuevo usuario
public function insertarUsuario(Request $request){
    $usuario=new usuario();
    $usuario->nombre=$request->input('nombre');
    $usuario->email=$request->input('email');
    $usuario->password=Hash::make($request->input('password'));
    $usuario->tipo_usuario=$request->input('tipo_usuario');
    $usuario->id_unidad_educativa=$request->input('id_unidad_educativa');
    $count=usuario::where('email', '=', $request->input('email'))->count();
    if($count==1){
        return response()->json(
        [
            ['HttpResponse' => [
                'status' => 400,
                'message'=>"No puede asignar este Correo"
            ]
            ],
        );
    }
    try {
        $usuario->save();
        return response()->json(
        [
            ['HttpResponse' => [
                'status' => 200,
                'message'=>'Usuario ingresado Correctamente',
                'statusText' => 'OK',
                'ok' => true
            ]
            ],
        );
    } catch (Exception $e) {
        return response()->json(
        [
            ['HttpResponse' => [
                'status' => 400,
                'message'=>$e
            ]
            ],
        );
    }
}
```

Elaborado por: El Investigador.

Ilustración 62 Actualizar Usuario.

```
//actualizar usuario
public function editarUsuario2(Request $request){
    $idUserio=$request->input('idUserio');
    $nombreUsuario=$request->input('nombreUsuario');
    $idUnidad=$request->input('idUnidad');
    $password=$request->input('password');
    $tipo_usuario=$request->input('tipo_usuario');
    try{
        //nombre
        if($nombreUsuario!=0){
            $usuario = usuario::where('ID',$idUserio)
            ->update(['NOMBRE'=>$nombreUsuario]);
        }
        //unidad
        if($idUnidad!=0){
            $usuario = usuario::where('ID',$idUserio)
            ->update(['ID_UNIDAD_EDUCATIVA'=>$idUnidad]);
        }
        //password
        if($password!=0){
            $usuario = usuario::where('ID',$idUserio)
            ->update(['PASSWORD'=>Hash::make($request->input('password'))]);
        }
        //tipo usuario
        if($tipo_usuario!=0){
            $usuario = usuario::where('ID',$idUserio)
            ->update(['TIPO_USUARIO'=>$tipo_usuario]);
        }
        return response()->json(
            [
                'HttpResponse' => [
                    'status' => 200,
                    'message'=>'Usuario actualizado Correctamente',
                    'statusText' => 'OK',
                    'ok' => true
                ]
            ]
        );
    }
}
```

Elaborado por: El Investigador.

Ilustración 63 Eliminar Usuario.

```
public function eliminarUsuario($id){
    $usuario=usuario::where('ID','=',$id)->get();
    $object=usuario::where('ID','=',$id);
    if($usuario[0]->TIPO_USUARIO==1){
        $count=usuario::where('TIPO_USUARIO', '=', 1)->count();
        //return $count;
    }
    if($usuario[0]->TIPO_USUARIO==1&&$count==1){
        return response()->json(
            [
                'HttpResponse' => [
                    'status' => 400,
                    'message'=>"No puede eliminar mas Administradores"
                ]
            ],
        );
    }else{
        $object->delete();
        return response()->json(
            [
                'HttpResponse' => [
                    'status' => 200,
                    'message'=>'Usuario eliminado Correctamente',
                    'statusText' => 'OK',
                    'ok' => true
                ]
            ],
        );
    }
    return $usuario[0]->TIPO_USUARIO;
}
```

Elaborado por: El Investigador.

Ilustración 64 Obtener lista de usuarios y roles.

```
public function getUsers() {
    //obtengo los usuarios id, nombre, email, tipo, unidad educativa
    $usuarios = usuario::join('unidad_educativa', 'users.ID_UNIDAD_EDUCATIVA', '=', 'unidad_educativa.ID')
    ->join('tipo_usuario', 'users.TIPO_USUARIO', '=', 'tipo_usuario.ID')
    ->select('users.ID', 'users.Nombre as nombreUser', 'users.email as emailUser', 'tipo_usuario.NOMBRE as tipoUser')
    return $usuarios;
    return response()->json(
        [
            'usuarios' => getImagenes($unidades),
            'HttpResponse' => [
                'status' => 200,
                'statusText' => 'OK',
                'ok' => true
            ]
        ],
    );
}

public function getRoles() {
    $roles = tipoUsuario::all();
    return $roles;
}
```

Elaborado por: El Investigador.

Ilustración 65 Login de usuario.

```
public function login()
{
    $credentials = request(['email', 'password']);

    if (! $token = auth()->attempt($credentials)) {
        return response()->json(
            [
                'HttpResponse' => [
                    'title' => 'ERROR de Inicio de Sesión',
                    'message' => 'Datos de inicio Incorrectos',
                    'status' => 401,
                    'statusText' => 'success',
                    'ok' => true
                ],
            ],
        );
    }

    return response()->json(
        [
            'token' => $token,
            'user' => Auth::user(),
            'HttpResponse' => [
                'title' => 'Iniciando Sesión',
                'message' => 'Ingreso Exitoso',
                'status' => 201,
                'statusText' => 'success',
                'ok' => true
            ],
        ],
    );
}
```

Elaborado por: El Investigador.

Ilustración 66 LogOut

```
public function logout()
{
    auth()->logout();

    return response()->json(['message' => 'Successfully logged out']);
}
```

Elaborado por: El Investigador.

Periodos Lectivos.

Ilustración 67 Ingresar nuevo Periodo Lectivo.

```
//insertar un nuevo año lectivo
public function insertarLectivo(Request $request){
    $idUnidad=$request->input('idUnidad');
    $nuevoNombre= $request->input('nombreLectivo');
    $nombreUnidad=$request->input('nombreUnidad');
    //comprobar si hay un lectivo con ese nombre en la unidad educativa
    $lectivoExiste = lectivo::where([
        ['ID_UNIDAD_EDUCATIVA', '=', $idUnidad],
        ['NOMBRE', '=', $nuevoNombre]])->get();
    if(count($lectivoExiste)<1){
        //obtener el nombre de la unidad
        $nombres = unidadEducativa::where('unidad_Educativa.ID','=', $idUnidad)
        ->join('anio_lectivo', 'anio_lectivo.ID_UNIDAD_EDUCATIVA', '=', 'unidad_educativa.ID')
        ->select('unidad_educativa.NOMBRE as nombreUnidad')
        ->first();
        $lectivo = new lectivo();
        $lectivo->nombre=$request->input('nombreLectivo');
        $lectivo->id_unidad_educativa=$request->input('idUnidad');
        $lectivo->save();
        //crear la carpeta en el disco para documentos
        Storage::makeDirectory('public/unidadesEducativas/'.$nombreUnidad.'/Documentos/'.$nuevoNombre);
        //crear la carpeta en el disco para boletas, con prom
        Storage::makeDirectory('public/unidadesEducativas/'.$nombreUnidad.'/BoletasConProm/'.$nuevoNombre);
        return response()->json([
            'HttpResponse' => [
                'message' => 'Año lectivo Creado!',
                'status' => 200,
                'statusText' => 'OK',
                'ok' => true
            ]
        ]);
    }else{
        return response()->json([
            'HttpResponse' => [
                'message' => 'Ya existe un Año Lectivo con este Nombre!',
                'status' => 400,
                'statusText' => 'error',
            ]
        ]);
    }
}
```

Elaborado por: El Investigador.

Ilustración 68 Eliminar Periodo Lectivo.

```
//eliminar año lectivo o periodo
public function eliminarLectivo(Request $request){
    $idLectivo= $request->input('idLectivo');
    //comprobar si existe el periodo o lectivo
    $lectivo = lectivo::where('ID', '=', $idLectivo);
    if (!$lectivo) {
        return response()->json([
            'HttpResponse' => [
                'title' => 'Error',
                'message' => 'No se encontro el lectivo!',
                'status' => 400,
                'statusText' => 'error',
                'ok' => true
            ]
        ]);
    }
    //obtener los nombres del lectivo y la unidad educativa
    $nombres = unidadEducativa::where('anio_lectivo.ID','=', $idLectivo)
    ->join('anio_lectivo', 'anio_lectivo.ID_UNIDAD_EDUCATIVA', '=', 'unidad_educativa.ID')
    ->select('unidad_educativa.NOMBRE as nombreUnidad', 'anio_lectivo.NOMBRE as nombreLectivo')
    ->first();
    //comprobar si el año lectivo tiene carpetas o documentos
    $count = documentacion::where('ID_ANIO_LECTIVO', '=', $idLectivo)->count();
    //comprobar si el año lectivo tiene algun nivel
    $countNivel= nivel::where('ID_ANIO_LECTIVO', '=', $idLectivo)->count();
    if($count>=1){
        return response()->json([
            'HttpResponse' => [
                'title' => 'Error',
                'message' => 'No puede eliminar Periodos con Documentos!',
                'status' => 400,
                'statusText' => 'error',
                'ok' => true
            ]
        ]);
    }
}
```

Elaborado por: El Investigador.

Ilustración 69 Actualizar periodo lectivo.

```
//editar año lectivo
public function editarLectivo(Request $request){
    $lectivo= $request->input('idlectivo');
    $nuevoNombre= $request->input('nombrelectivo');
    //comprobar si no existe otro lectivo con ese nombre
    $lectivoExiste = lectivo::where([
        ['ID', '=', $idlectivo],
        ['NOMBRE', '=', $nuevoNombre]])->get();

    if(count($lectivoExiste)<1){
        //
        $nombres = unidadEducativa::where("anio lectivo.ID", "=", $idlectivo)
        ->join("anio lectivo", "anio lectivo.ID UNIDAD EDUCATIVA", "=", "unidad educativa.ID")
        ->select("unidad educativa.NOMBRE as nombreUnidad", "anio lectivo.NOMBRE as nombrelectivo")
        ->first();
        $lectivo=lectivo::where("ID", "=", $idlectivo)
        ->update(['NOMBRE'=>$nuevoNombre]);

        Storage::move("public/unidadesEducativas/" . $nombres->nombreUnidad . "/Documentos/" . $nombres->nombrelectivo,
            "public/unidadesEducativas/" . $nombres->nombreUnidad . "/Documentos/" . $nuevoNombre);

        Storage::move("public/unidadesEducativas/" . $nombres->nombreUnidad . "/BoletasConProm/" . $nombres->nombrelectivo,
            "public/unidadesEducativas/" . $nombres->nombreUnidad . "/BoletasConProm/" . $nuevoNombre);

    }

    if(!$lectivo){
        return "no existe el periodo";
    }else{
        return response()->json(
            [
                'HttpResponse' => [
                    'title' => 'Correcto',
                    'message' => 'Año Lectivo Actualizado!',
                    'status' => 200,
                    'statusText' => 'success',
                    'ok' => true
                ]
            ]
        );
    }
}
```

Elaborado por: El Investigador.

Ilustración 70 Obtener periodos lectivos por unidad educativa.

```
//obtenerAños LectivosxidUnidadEducativa
public function getLectivos($id){
    $lectivos = lectivo::where('ID_UNIDAD_EDUCATIVA', '=', $id)->get();

    if (!$lectivos) {
        return response()->json([
            'HttpResponse' => [
                'title' => 'Error',
                'message' => 'No se cargo losc Años Lectivos',
                'status' => 400,
                'statusText' => 'error',
                'ok' => true
            ]
        ]
    );
    }

    return response()->json([
        'lectivos' => $lectivos,
        'HttpResponse' => [
            'message' => 'Años lectivos Consultados',
            'status' => 200,
            'statusText' => 'OK',
            'ok' => true
        ]
    ]
    );
}
```

Elaborado por: El Investigador.

Ilustración 71 Ingresar Nueva Categoría.

```
//insertar una nueva categoría a disco
public function insertarCarpeta(Request $request){
    $idlectivo= $request->input('idlectivo');
    $nuevoNombre= $request->input('nombre');
    //obtener nombre de unidad y lectivo
    $nombres = unidadEducativa::where('anio lectivo.ID','=', $idlectivo)
    ->join('anio lectivo','anio lectivo.ID UNIDAD EDUCATIVA','=', 'unidad educativa.ID')
    ->select('unidad_educativa.NOMBRE as nombreUnidad','anio_lectivo.NOMBRE as nombrelectivo')
    ->first();
    $carpeta = documentacion::where([
        ['ID ANIO LECTIVO', '=', $idlectivo],
        ['NOMBRE','=', $nuevoNombre]])->get();
    $nuevaCarpeta = new documentacion();
    if(count($carpeta)<1){
        $nuevaCarpeta->id anio lectivo= $request->input('idlectivo');
        $nuevaCarpeta->nombre= $request->input('nombre');
        $nuevaCarpeta->save();
        //creo la carpeta en disco
        Storage::makeDirectory('public/unidadesEducativas/'.$nombres->nombreUnidad.'/Documentos/'.$nombres->
    return response()->json(
        [
            'unidad Educativa' => $nuevaCarpeta,
            'HttpResponse' => [
                'title' => 'Correcto',
                'message' => 'Nueva Carpeta de documentacion creada!',
                'status' => 200,
                'statusText' => 'success',
                'ok' => true
            ],
        ],
    );
    }else{
        return response()->json([
            'HttpResponse' => [
                'message' => 'Carpeta con nombre ya existente! '.$request->input('nombre'),
                'status' => 400,
            ],
        ],
    );
    }
}
```

Elaborado por: El Investigador.

Ilustración 72 Actualizar categoría.

```
//editar el nombre de una carpeta
public function editarCarpeta(Request $request){
    $idCarpeta= $request->input('idCarpeta');
    $idlectivo= $request->input('idlectivo');
    $nuevoNombre= $request->input('nombreCarpeta');
    //comprobar si no existe otra carpeta con ese nombre
    $carpeta = documentacion::where([
        ['ID ANIO LECTIVO', '=', $idlectivo],
        ['NOMBRE','=', $nuevoNombre]])->get();

    if(count($carpeta)<1){
        //
        $nombre = unidadEducativa::where('documentacion.ID','=', $idCarpeta)
        ->join('anio lectivo','anio lectivo.ID UNIDAD EDUCATIVA','=', 'unidad educativa.ID')
        ->join('documentacion','documentacion.ID ANIO LECTIVO','=', 'anio lectivo.ID')
        ->select('unidad_educativa.NOMBRE as nombreUnidad','anio_lectivo.NOMBRE as nombrelectivo','documentacion.NOMBRE
        ->first();
        $carpeta=documentacion::where('ID','=', $idCarpeta)
        ->update(['NOMBRE'=>$nuevoNombre]);

        Storage::move('public/unidadesEducativas/'.$nombre->nombreUnidad.'/Documentos/'.$nombre->nombrelectivo.'/'.'
        'public/unidadesEducativas/'.$nombre->nombreUnidad.'/Documentos/'.$nombre->nombrelectivo.'/'.' $nuevoNombre);

    if(!$carpeta){
        return "no existe la carpeta";
    }else{
        return response()->json(
            [
                'HttpResponse' => [
                    'title' => 'correcto',
                    'message' => 'Carpeta actualizada!',
                    'status' => 200,
                    'statusText' => 'success',
                ],
            ],
        );
    }
}
```

Elaborado por: El Investigador.

Ilustración 73 Eliminar categoría.

```
//eliminar una carpeta o doc
public function eliminarCarpeta2(Request $request){
    $idLectivo= $request->input('idLectivo');
    $nombreCarpeta= $request->input('nombreCarpeta');
    $id= $request->input('idCarpeta');
    $carpeta = documentacion::where('ID', '=', $id);

    //obtener nombre de unidad y lectivo
    $nombres = unidadEducativa::where('anio_lectivo.ID','=', $idLectivo)
->join('anio_lectivo','anio_lectivo.ID_UNIDAD_EDUCATIVA','=', 'unidad_educativa.ID')
->select('unidad_educativa.NOMBRE as nombreUnidad','anio_lectivo.NOMBRE as nombreLectivo')
->first();

    if (!$carpeta) {
        return response()->json([
            'HttpResponse' => [
                'title' => 'Error',
                'message' => 'No se encontro la carpeta!',
                'status' => 400,
                'statusText' => 'error',
                'ok' => true
            ]
        ]);
    }
    $count = detalleDocumentacion::where('ID_DOCUMENTACION', '=', $id)->count();

    if ($count>=1){
        return response()->json([
            'HttpResponse' => [
                'title' => 'Error',
                'message' => 'No puede eliminar Carpetas con Archivos!',
                'status' => 400,
                'statusText' => 'error',
                'ok' => true
            ]
        ]);
    }
};
```

Elaborado por: El Investigador.

Ilustración 74 Subida de Archivos a documentación.

```
//archivos
public function subirArchivo(Request $request){
    //verifico si no hay otro archivo con ese nombre
    ///idunidad / Documentos / idaniolectivo / idcarpeta
    $idUnidad= $request->input('idUnidad');
    $idLectivo= $request->input('idLectivo');
    $idCarpeta= $request->input('idCarpeta');
    $nombre= $request->input('nombreArchivo');
    $archivo = detalleDocumentacion::where([
        ['ID_DOCUMENTACION', '=', $idCarpeta],
        ['NOMBRE_ARCHIVO', '=', $nombre]])->get();
    if(count($archivo)<1){
        //get el nombre de la unidad
        $unidad = unidadEducativa::where('id','=', $idUnidad)->get();
        //GET el nombre del anio lectivo
        $lectivo= lectivo::where("id", '=', $idLectivo)->get();
        //GET el nombre de la carpeta
        $carpeta= documentacion::where('id', '=', $idCarpeta)->get();
        //comprobar nombre del archivo con el id de la carpeta
        $archivo = new detalleDocumentacion();
        $archivo->id_documentacion = $request->input('idCarpeta');
        $archivo->nombre_archivo = $request->input('nombreArchivo');
        //rectivo y guardo el archivo
        $archivoFile = $request->file('archivo');
        $linkRoute= 'unidadesEducativas/'.$unidad[0]->NOMBRE.'/'.$Documentos/'.$lectivo[0]->NOMBRE.'/'.$carpeta[0]->NOMBRE;
        $archivo->ruta_archivo = $archivoFile->storeAs($linkRoute, $archivo->nombre_archivo, 'public');
        $archivo->formato = $request->input('formato');
        $archivo->save();

        return response()->json([
            'HttpResponse' => [
                'message' => 'archivoGuardado',
                'status' => 200,
                'statusText' => 'OK',
                'ok' => true
            ]
        ]);
    }
};
```

Elaborado por: El Investigador.

Ilustración 75 Listar los archivos dentro de la categoría de documentación.

```
//Listar los archivos por el id de la carpeta o doc
public function getArchivosbyDoc($id){
    $archivos = detalleDocumentacion::where('ID_DOCUMENTACION', '=', $id)->get();

    if (!$archivos) {
        return response()->json([
            'HttpResponse' => [
                'title' => 'Error',
                'message' => 'No se cargo los documentos',
                'status' => 400,
                'statusText' => 'error',
                'ok' => true
            ]
        ]);
    }
    return response()->json([
        'Archivos' => $archivos,
        'HttpResponse' => [
            'message' => 'Documentos Consultados',
            'status' => 200,
            'statusText' => 'OK',
            'ok' => true
        ]
    ]);
}
```

Elaborado por: El Investigador.

Ilustración 76 Obtener el Archivo

```
//descargar el archivo
public function descargarArchivo(Request $request){

    $idCarpeta=$request->input('idCarpeta');
    $formato=$request->input('formato');
    $nombre=$request->input('nombre');

    //asi para obtener los nombres de las carpetas
    $nombres = unidadEducativa::where('detalle_documentacion.ID_DOCUMENTACION', '=', $idCarpeta)
    ->join('anio_lectivo', 'anio_lectivo.ID_UNIDAD_EDUCATIVA', '=', 'unidad_educativa.ID')
    ->join('documentacion', 'documentacion.ID_ANIO_LECTIVO', '=', 'anio_lectivo.ID')
    ->join('detalle_documentacion', 'detalle_documentacion.ID_DOCUMENTACION', '=', 'documentacion.ID')
    ->select('unidad_educativa.NOMBRE as nombreUnidad', 'anio_lectivo.NOMBRE as nombreLectivo', 'documentacion.NOMBRE as nombreCarpeta')
    ->first();

    $ruta='unidadesEducativas/'.$nombres->nombreUnidad.'/documentos/'.$nombres->nombreLectivo.'/'.$nombres->nombreCarpeta.'/'.$nombre;

    $file = storage_path()."app/public/$ruta";
    $headers = [
        'Content-Type' => $formato,
    ];
    return response()->download($file,$nombre,$headers);
}
```

Elaborado por: El Investigador.

Ilustración 77 Eliminar Archivo

```
//Eliminar Archivo con la ruta
public function eliminarArchivo(Request $request){
    $idCarpeta=$request->input('idCarpeta');
    $nombre=$request->input('nombre');
    //GET para obtener los nombres de las carpetas
    $nombres = unidadEducativa::where('detalle_documentacion.ID_DOCUMENTACION','=',$idCarpeta)
    ->join('anio_lectivo','anio_lectivo.ID_UNIDAD_EDUCATIVA','=',$unidad_educativa.ID')
    ->join('documentacion','documentacion.ID_ANIO_LECTIVO','=',$anio_lectivo.ID')
    ->join('detalle_documentacion','detalle_documentacion.ID_DOCUMENTACION','=',$documentacion.ID')
    ->select('unidad_educativa.NOMBRE as nombreunidad','anio_lectivo.NOMBRE as nombrelectivo','documentacion.NOMBRE as nombreCarpeta')
    ->first();
    $ruta='unidadesEducativas/'.$nombres->nombreunidad.'/Documentos/'.$nombres->nombrelectivo.'/'.$nombres->nombreCarpeta.'/'.$nombre;
    $idCarpeta=$request->input('idCarpeta');
    $archivo = detalleDocumentacion::where([
        ['NOMBRE_ARCHIVO','=',$nombre],
        ['ID_DOCUMENTACION','=',$idCarpeta],
    ]);
    if (!$archivo) {
        return response()->json([
            'HttpResponse' => [
                'title' => 'Error',
                'message' => 'No se encontro el archivo',
                'status' => 400,
                'statusText' => 'error',
                'ok' => true
            ]
        ]);
    }
    try {
        Storage::disk('public')->delete($ruta);
        $archivo->delete();
        return response()->json([
            'HttpResponse' => [
                'title' => 'Correcto',
                'message' => 'Archivo eliminado!',
                'status' => 200,
            ]
        ]);
    }
}
```

Elaborado por: El Investigador.

Niveles o cursos según el periodo lectivo.

Ilustración 78 Ingresar nivel.

```
//insertar nivel
public function insertarNivel(Request $request){
    $nivel = new nivel();
    //verifico si no hay un nivel con ese nombre en esa jornada
    $existe=nivel::where([
        ['NOMBRE','=',$request->input('nombrenivel')],
        ['JORNADA','=',$request->input('jornada')],
        ['ID_ANIO_LECTIVO','=',$request->input('idlectivo')]
    ])->get();
    if(count($existe)>0){
        //obtengo los nombres para las carpetas
        $nombres = unidadEducativa::where('anio_lectivo.ID','=',$request->input('idlectivo'))
        ->join('anio_lectivo','anio_lectivo.ID_UNIDAD_EDUCATIVA','=',$unidad_educativa.ID')
        ->select('unidad_educativa.NOMBRE as nombreunidad','anio_lectivo.NOMBRE as nombrelectivo')->first();
        $nivel->nombre=$request->input('nombrenivel');
        $nivel->id_anio_lectivo=$request->input('idlectivo');
        $nivel->jornada=$request->input('jornada');
        $nivel->save();
        //creo la carpeta en disco
        Storage::makeDirectory('public/unidadesEducativas/'.$nombres->nombreunidad.'/BoletasConProm/'.$nombres->nombrelectivo.'/'.$request->input('nombrenivel'));
        return response()->json([
            'HttpResponse' => [
                'message' => 'Nivel Creado!',
                'status' => 200,
                'statusText' => 'OK',
                'ok' => true
            ]
        ]);
    }else{
        return response()->json([
            'HttpResponse' => [
                'message' => 'Nivel:'.$request->input('nombrenivel').'.'.$request->input('jornada').' ya existe!',
                'status' => 400,
            ]
        ]);
    }
}
```

Elaborado por: El Investigador.

Ilustración 79 Eliminar Nivel.

```
//eliminar nivel
public function borrarNivel($idNivel){
//compruebo si el nivel tiene paralelos
$count = paralelo::where('ID_NIVEL', '=', $idNivel)->count();
if($count>1){
return response()->json([
'HttpResponse' => [
'title' => 'Error',
'message' => 'No puede eliminar Niveles que contengan Paralelos!',
'status' => 400,
'statusText' => 'error',
'ok' => true
]
]);
}else{
try {
//obtengo los nombres para las carpetas
$nombr = unidadEducativa::where('nivel.ID', '=', $idNivel)
->join('anio_lectivo', 'anio_lectivo.ID_UNIDAD_EDUCATIVA', '=', 'unidad_educativa.ID')
->join('nivel', 'nivel.ID_ANIO_LECTIVO', '=', 'anio_lectivo.ID')
->select('unidad_educativa.NOMBRE as nombreUnidad', 'nivel.JORNADA as jornada', 'nivel.NOMBRE as nombreNivel',
'anio_lectivo.NOMBRE as nombreLectivo')->first();
//obtengo el nivel y lo elimino
$nivel = nivel::where('ID', '=', $idNivel);
$nivel->delete();
//elimino la carpeta en el disco
Storage::deleteDirectory('public/unidadesEducativas/'.$nombr->nombreUnidad.'/BoletasConProm/'.$nombr->nombreLectivo.'/'.$nombr->nombreNivel);
$nivel->delete();

return response()->json([
'HttpResponse' => [
'title' => 'Correcto',
'message' => 'Nivel eliminado!',
'status' => 200,
'statusText' => 'success',
'ok' => true
]
]);
}
}
}
```

Elaborado por: El Investigador.

Ilustración 80 Actualizar Nivel.

```
//editar nivel
public function editarNivel(Request $request){
$idNivel=$request->input('idNivel');
$nombreNivel=$request->input('nombreNivel');
//obtengo los nombres para las carpetas
$nombr = unidadEducativa::where('nivel.ID', '=', $idNivel)
->join('anio_lectivo', 'anio_lectivo.ID_UNIDAD_EDUCATIVA', '=', 'unidad_educativa.ID')
->join('nivel', 'nivel.ID_ANIO_LECTIVO', '=', 'anio_lectivo.ID')
->select('unidad_educativa.NOMBRE as nombreUnidad', 'anio_lectivo.ID as idlectivo', 'nivel.JORNADA as jornada', 'nivel.NOMBRE as nombreNivel',
'anio_lectivo.NOMBRE as nombreLectivo')->first();
//compruebo si no existe otro nivel con ese nombre
$count = nivel::where([
['NOMBRE', '=', $nombreNivel],
['JORNADA', '=', $nombr->jornada],
['ID_ANIO_LECTIVO', '=', $nombr->idlectivo]])->count();
if($count>1){
return response()->json([
'HttpResponse' => [
'message' => 'No se puede asignar este nombre!',
'status' => 400,
'statusText' => 'error',
'ok' => true
]
]);
}else{
$nivel=nivel::where('ID', '=', $idNivel)
->update(['NOMBRE'=>$nombreNivel]);

Storage::move('public/unidadesEducativas/'.$nombr->nombreUnidad.'/BoletasConProm/'.$nombr->nombreLectivo.'/'.$nombr->nombreNivel, 'public/unidadesEducativas/'.$nombr->nombreUnidad.'/BoletasConProm/'.$nombr->nombreLectivo.'/'.$nombr->nombreNivel);
if($idNivel){
return "no existe el nivel";
}else{
return response()->json(
[
'HttpResponse' => [

```

Elaborado por: El Investigador.

Ilustración 81 Obtener listado de niveles por periodo lectivo.

```
//obtener los niveles por anio lectivo
public function getNiveles($idLectivo){
    $niveles = nivel::where('ID_ANIO_LLECTIVO','=', $idLectivo)->get();
    return response()->json([
        'niveles' => $niveles,
        'HttpResponse' => [
            'message' => 'Años lectivos Consultados',
            'status' => 200,
            'statusText' => 'OK',
            'ok' => true
        ]
    ]);
}
```

Elaborado por: El Investigador.

Paralelos según el nivel seleccionado.

Ilustración 82 Ingresar Paralelo

```
//subir paralelo
public function subirParalelo(Request $request){
    $idnivel=$request->input('idnivel');
    $nombreParalelo=$request->input('nombreParalelo');
    //obtengo los nombres para las carpetas
    $nombres = unidadEducativa::where('nivel_ID','=', $idnivel)
    ->join('anio_lectivo','anio_lectivo.ID_UNIDAD_EDUCATIVA',' ','unidad_educativa.ID')
    ->join('nivel','nivel.ID_ANIO_LLECTIVO',' ','anio_lectivo.ID')
    ->select('unidad_educativa.NOMBRE as nombreInidad','nivel.JORNADA as jornada','nivel.NOMBRE as nombreInivel','anio_lectivo.NOMBRE as anio');
    //comprobar si no existe otro paralelo con ese nombre en el nivel
    $existe=paralelo::where([
        ['NOMBRE','=', $nombreParalelo],
        ['ID_NIVEL','=', $idnivel]
    ])->get();
    if(count($existe)>0){
        $paralelo = new paralelo();
        $paralelo->nombre=$nombreParalelo;
        $paralelo->id_nivel=$idnivel;
        $paralelo->save();
        //creo lo carpeta en disco
        Storage::makeDirectory('public/unidadesEducativas/'.$nombres->nombreInidad.'/BoletasConProm/'.$nombres->nombreInivel.'/'.$nombres->nombreInidad);
    }
    return response()->json([
        'HttpResponse' => [
            'message' => 'Paralelo Creado Correctamente',
            'status' => 200,
            'statusText' => 'OK',
            'ok' => true
        ]
    ]);
}
else{
    return response()->json([
        'HttpResponse' => [
            'message' => 'Paralelo con nombre ya existente! '.$request->input('nombreParalelo'),
            'status' => 400,
            'statusText' => 'error',
        ]
    ]);
}
```

Elaborado por: El Investigador.

Ilustración 83 Eliminar Paralelo.

```
//eliminar paralelo
public function borrarParalelo($id){
//compruebo si el paralelo no tiene archivos
$count = detalleParalelo::where('ID_PARALELO', '=', $id)->count();
if($count>1){
return response()->json([
'HttpResponse' => [
'tittle' => 'Error',
'message' => 'no puede eliminar Paralelos con Documentos!',
'status' => 400,
'statusText' => 'error',
'ok' => true
]
]);
}else{
try {
//obtengo los nombres para las carpetas
$nombres = unidadEducativa::where('paralelo.ID', '=', $id)
->join('anio_lectivo', 'anio_lectivo.ID_UNIDAD_EDUCATIVA', '=', 'unidad_educativa.ID')
->join('nivel', 'nivel.ID_ANIO_LECTIVO', '=', 'anio_lectivo.ID')
->join('paralelo', 'paralelo.ID_NIVEL', '=', 'nivel.ID')
->select('unidad_educativa.NOMBRE as nombreUnidad', 'nivel.JORNADA as jornada', 'paralelo.NOMBRE as nombreParalelo', 'nivel.NOMBRE as nom
//obtengo el paralelo y lo elimino
$paralelo = paralelo::where('ID', '=', $id);
$paralelo->delete();
//elimino la carpeta en el disco
Storage::deleteDirectory('public/unidadesEducativas/'.$nombres->nombreUnidad.'/BoletasConProm/'.$nombres->nombrelectivo.'/'.$nomb
$paralelo->delete();

return response()->json([
'HttpResponse' => [
'tittle' => 'Correcto',
'message' => 'Paralelo eliminado!',
'status' => 200,
'statusText' => 'success',
'ok' => true
]
],
);
}
}
```

Elaborado por: El Investigador.

Ilustración 84 Actualizar Paralelo.

```
//editar paralelo
public function editarParalelo(Request $request){
$idParalelo=$request->input('idParalelo');
$nombreParalelo=$request->input('nombreParalelo');
//comprobar si no existe otro paralelo con ese nombre
$existe = paralelo::where([
['NOMBRE', '=', $nombreParalelo]])->get();

if(count($existe)<1){
//obtengo los nombres para las carpetas
$nombres = unidadEducativa::where('paralelo.ID', '=', $idParalelo)
->join('anio_lectivo', 'anio_lectivo.ID_UNIDAD_EDUCATIVA', '=', 'unidad_educativa.ID')
->join('nivel', 'nivel.ID_ANIO_LECTIVO', '=', 'anio_lectivo.ID')
->join('paralelo', 'paralelo.ID_NIVEL', '=', 'nivel.ID')
->select('unidad_educativa.NOMBRE as nombreUnidad', 'nivel.JORNADA as jornada', 'paralelo.NOMBRE as nombreParalelo', 'nivel.NOMBRE as nom
'anio_lectivo.NOMBRE as nombrelectivo')->first();

$paralelo=paralelo::where('ID', '=', $idParalelo)
->update(['NOMBRE' => $nombreParalelo]);

Storage::move('public/unidadesEducativas/'.$nombres->nombreUnidad.'/BoletasConProm/'.$nombres->nombrelectivo.'/'.$nombres->jornada
'public/unidadesEducativas/'.$nombres->nombreUnidad.'/BoletasConProm/'.$nombres->nombrelectivo.'/'.$nombres->jornada.'_'.$nombres->

if(!$paralelo){
return "no existe el periodo";
}else{
return response()->json(
[
'HttpResponse' => [
'tittle' => 'Correcto',
'message' => 'Paralelo Actualizado!',
'status' => 200,
'statusText' => 'success',
'ok' => true
]
]
);
}
}
```

Elaborado por: El Investigador.

Ilustración 85 Subir archivos al paralelo.

```
//subirArchivo
public function subirArchivo(Request $request){
    //verifico si no hay otro archivo con ese nombre
    //idUnidad / Documentos / idAniolectivo / idCarpeta
    $idParalelo= $request->input('idParalelo');
    $tipoDocumento= $request->input('tipoDocumento');
    $nombre= $request->input('nombreArchivo');

    $archivo = detalleParalelo::where([
        ['ID_PARALELO', '=', $idParalelo],
        ['NOMBRE_ARCHIVO', '=', $nombre],
        ['TIPO_DOCUMENTO', '=', $tipoDocumento]])->get();
    if(count($archivo)<1){
        $nombres = unidadEducativa::where('paralelo.ID', '=', $idParalelo)
        ->join('anio_lectivo', 'anio_lectivo.ID_UNIDAD_EDUCATIVA', '=', 'unidad_educativa.ID')
        ->join('nivel', 'nivel.ID_ANIO_LECTIVO', '=', 'anio_lectivo.ID')
        ->join('paralelo', 'paralelo.ID_NIVEL', '=', 'nivel.ID')
        ->select('unidad_educativa.NOMBRE as nombreUnidad', 'nivel.JORNADA as jornada', 'paralelo.NOMBRE as nombreParalelo', 'nivel.NOMBRE as no
        ///
        $archivo = new detalleParalelo();
        $archivo->id_paralelo = $request->input('idParalelo');
        $archivo->nombre_archivo = $request->input('nombreArchivo');
        $archivo->tipo_documento = $request->input('tipoDocumento');
        if($request->input('tipoDocumento')==1){
            $tipo="Boletas";
        }else if($request->input('tipoDocumento')==2){
            $tipo="Promociones";
        }else{
            $tipo="Concentrado";
        }
    }
    //recibo y guardo el archivo
    $archivoFile = $request->file('archivo');
    $linkRoute= 'unidadesEducativas/'.$nombres->nombreUnidad.'/BoletasConProm/'.$nombres->nombreLectivo.'/'.$nombres->jornada.'_'.$nombre
    //linkRoute= 'unidadesEducativas/pruebas/';
    $archivoGuardado= $archivoFile->storeAs($linkRoute, $archivo->nombre_archivo, 'public');
    $archivo->formato = $request->input('formato');
    $archivo->save();
}
```

Elaborado por: El Investigador.

Ilustración 86 Obtener los archivos del paralelo.

```
//obtener los archivos por paralelos
public function getArchivos($id){
    // $id=$request->input('id');
    $lectivos = detalleParalelo::where('ID_PARALELO', '=', $id)->get();
    if (!$lectivos) {
        return response()->json([
            'HttpResponse' => [
                'tittle' => 'Error',
                'message' => 'No se cargo los archivos',
                'status' => 400,
                'statusText' => 'error',
                'ok' => true
            ]
        ]);
    }
    return response()->json([
        'archivos' => $lectivos,
        'HttpResponse' => [
            'message' => 'Archivos Consultados',
            'status' => 200,
            'statusText' => 'OK',
            'ok' => true
        ]
    ]);
}
```

Elaborado por: El Investigador.

Ilustración 87 Obtener el archivo.

```
//descargar el archivo
public function descargarArchivo(Request $request){

    $idParalelo=$request->input('idParalelo');
    $formato=$request->input('formato');
    $nombre=$request->input('nombre');
    $tipoDocumento=$request->input('tipoDocumento');
    if($request->input('tipoDocumento')==1){
        $tipo="Boletas";
    }else if($request->input('tipoDocumento')==2){
        $tipo="Promociones";
    }else{
        $tipo="concentrado";
    }
    //GET para obtener los nombres de las carpetas
    $nombres = unidadEducativa:where('paralelo.ID','=', $request->input('idParalelo'))
    ->join('anio_lectivo','anio_lectivo.ID UNIDAD EDUCATIVA','=', 'unidad_educativa.ID')
    ->join('nivel','nivel.ID ANIO LECTIVO','=', 'anio_lectivo.ID')
    ->join('paralelo','paralelo.ID NIVEL','=', 'nivel.ID')
    ->select('unidad_educativa.NOMBRE as nombreUnidad','nivel.JORNADA as jornada','paralelo.NOMBRE as nombreParalelo','nivel.NOMBRE as
    'anio_lectivo.NOMBRE as nombreLectivo')->first();
    $ruta= 'unidadesEducativas/'.$nombres->nombreUnidad.'/BoletasConProm/'.$nombres->nombreLectivo.'/'.$nombres->jornada.'_'.$nombres->nombreParalelo.'.'.$formato;

    $file = storage_path()."\app\public/$ruta";
    $headers = [
        'Content-Type' => $formato,
    ];
    return response()->download($file,$nombre,$headers);
}
```

Elaborado por: El Investigador.

Ilustración 88 Eliminar Archivo.

```
//eliminar archivo
public function borrarArchivo(Request $request){

    $idParalelo=$request->input('idParalelo');
    $nombre=$request->input('nombre');
    $tipoDocumento=$request->input('tipoDocumento');
    if($request->input('tipoDocumento')==1){
        $tipo="Boletas";
    }else if($request->input('tipoDocumento')==2){
        $tipo="Promociones";
    }else{
        $tipo="concentrado";
    }
    //GET para obtener los nombres de las carpetas
    $nombres = unidadEducativa:where('paralelo.ID','=', $request->input('idParalelo'))
    ->join('anio_lectivo','anio_lectivo.ID UNIDAD EDUCATIVA','=', 'unidad_educativa.ID')
    ->join('nivel','nivel.ID ANIO LECTIVO','=', 'anio_lectivo.ID')
    ->join('paralelo','paralelo.ID NIVEL','=', 'nivel.ID')
    ->select('unidad_educativa.NOMBRE as nombreUnidad','nivel.JORNADA as jornada','paralelo.NOMBRE as nombreParalelo','nivel.NOMBRE as nombreLectivo')->first();

    $ruta= 'unidadesEducativas/'.$nombres->nombreUnidad.'/BoletasConProm/'.$nombres->nombreLectivo.'/'.$nombres->jornada.'_'.$nombres->nombreParalelo.'.'.$formato;
    $archivo = detalleParalelo:where([
        ['NOMBRE ARCHIVO','=', $nombre],
        ['TIPO DOCUMENTO','=', $tipoDocumento],
        ['ID PARALELO','=', $idParalelo]
    ]);

    if (!$archivo) {
        return response()->json([
            'HttpResponse' => [
                'title' => 'Error',
                'message' => 'No se encontro el archivo!',
                'status' => 400,
                'statusText' => 'error',
                'ok' => true
            ]
        ]);
    }
}
```

Elaborado por: El Investigador.

Noticias, notificaciones o eventos.

Ilustración 89 Ingresar Noticia.

```
public function subirNoticia(Request $request){
    $noticia=new noticia();
    $noticia->titulo=$request->input('titulo');
    $noticia->descripcion=$request->input('descripcion');
    $imagen=$request->file('imagen');
    $noticia->imagen=$noticia->titulo.'.'.$imagen->getClientOriginalExtension();
    try {
        $noticia->save();
        $imagen->storeAs('noticias', $noticia->titulo . '.' . $imagen->getClientOriginalExtension(), 'public');
        return response()->json(
            [
                'HttpResponse' => [
                    'status' => 200,
                    'message'=>'Noticia ingresada Correctamente',
                    'statusText' => 'OK',
                    'ok' => true
                ]
            ],
        );
    } catch (Exception $e) {
        return response()->json(
            [
                'HttpResponse' => [
                    'status' => 400,
                    'message'=>$e
                ]
            ],
        );
    }
}
```

Elaborado por: El Investigador.

Ilustración 90 Obtener Noticias.

```
public function getNoticias(){
    function getImagenes($noticias){
        $array=[];
        foreach($noticias as $noticias){
            $object = (object)['ID' => $noticias->ID, 'URL' => asset('storage/noticias/'.$noticias->IMAGEN),'TITULO' => $noticias->TITULO,
                'DESCRIPCION' => $noticias->DESCRIPCION];
            array_push($array, $object);
        }
        return $array;
    }
    $noticias = noticia::all();
    return response()->json(
        [
            //unidades' => $unidades,
            'noticias' => getImagenes($noticias),
            'HttpResponse' => [
                'status' => 200,
                'statusText' => 'OK',
                'ok' => true
            ]
        ],
    );
}
```

Elaborado por: El Investigador.

Ilustración 91 Eliminar Noticia.

```
public function eliminarNoticia($id){
    $noticia = noticia::where('ID','=', $id)->get();
    $objeto=noticia::where('ID','=', $id);
    try{
        $objeto->delete();
        Storage::disk('public')->delete('noticias/'.$noticia[0]->IMAGEN);
        return response()->json(
            [
                'HttpResponse' => [
                    'status' => 200,
                    'message'=>'Noticia eliminada Correctamente',
                    'statusText' => 'OK',
                    'ok' => true
                ]
            ],
        );
    } catch (Exception $e) {
        return response()->json(
            [
                'HttpResponse' => [
                    'status' => 400,
                    'message'=>$e
                ]
            ],
        );
    }
}
```

Elaborado por: El Investigador.

Reportes.

Ilustración 92 Reporte de documentos institucionales.

```
//reporte de documentos de una unidad educativo en un periodo lectivo
public function reporteDocumentos($idLectivo){
    $carpetas=documentacion::where('ID_ANIO_LECTIVO', '=', $idLectivo)->get();
    function getDocs($carpetas){
        $array=[];
        $diaActual = Carbon::now()->locale('es')->translatedFormat('l d \d\e F \d\e l Y');
        foreach($carpetas as $carpetas){
            $object = (object)['Fecha'=>$diaActual, 'ID' => $carpetas->ID, 'NOMBRE' => $carpetas->NOMBRE,
            'lectivo' => $carpetas->ID_ANIO_LECTIVO, 'docs'=>detalleDocumentacion::where('ID_DOCUMENTACION', '=', $carpetas->ID)->get()];
            array_push($array, $object);
        }
        return $array;
    }
    return getDocs($carpetas);
}
```

Elaborado por: El Investigador.

Ilustración 93 Reporte de archivos estudiantes por nivel.

```
//reporte de documentos de paralelos y estudiantes en un nivel
public function reporteNivel($idNivel){
    $paralelos = paralelo::where('ID_NIVEL', '=', $idNivel)->get();
    //return $paralelos;
    function getArchivos($paralelos){
        $array=[];
        $diaActual = Carbon::now()->locale('es')->translatedFormat('l d \d\e F \d\e l Y');
        foreach($paralelos as $paralelos){
            $object = (object)['Fecha'=>$diaActual, 'ID' => $paralelos->ID, 'NOMBRE' => $paralelos->NOMBRE,
            'nivel' => $paralelos->ID_NIVEL, 'archivos'=>detalleParalelo::where('ID_PARALELO', '=', $paralelos->ID)
            ->orderBy('TIPO_DOCUMENTO')->get()];
            array_push($array, $object);
        }
        return $array;
    }
    return getArchivos($paralelos);
}
```

Elaborado por: El Investigador.

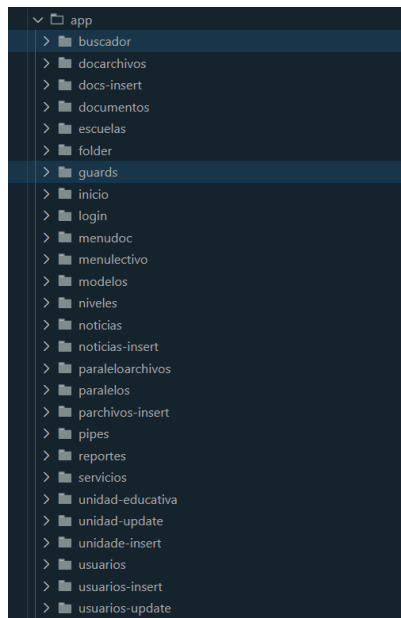
Ilustración 94 Reporte archivos de estudiantes por paralelo.

```
///reporte de documentos de estudiantes por paralelo
public function reporteParalelo($idParalelo){
    $detalle = detalleParalelo::where('ID_PARALELO', '=', $idParalelo)
    ->orderBy('TIPO_DOCUMENTO')->get();
    $diaActual = Carbon::now()->locale('es')->translatedFormat('l d \d\e F \d\e\l Y');
    return response()->json(
        [
            'archivos' => $detalle,
            'fecha'=> $diaActual,
            'HttpResponse' => [
                'status' => 200,
                'statusText' => 'OK',
                'ok' => true
            ]
        ],
    );
}
```

Elaborado por: El Investigador.

Pages y métodos en el Front-End para la comunicación con el Back-End.

Ilustración 95 Pages para el Front-End



Elaborado por: El Investigador.

3.2.6.10.2. Codificación Front-End

Métodos de peticiones http para consumir.

Ilustración 96 Peticiones para Unidad Educativa.

```
//
sendObjectSource(data:any){
  this.objectSource.next(data);
}
//insertar nueva UnidadEducativa
postUnidad(unidad:FormData){
  return this.httpClient.post(this.Api+'insertarUnidad',unidad);
}
//eliminar unidadEducativa
deleteUnidad(id){
  return this.httpClient.delete(this.Api+'eliminarUnidad/'+id);
}
//actualizar la unidadEducativa
updateUnidad(unidad:FormData){
  return this.httpClient.post(this.Api+'editarUnidad',unidad);
}
//obtener listado de unidades educativas
getUnidades(){
  return this.httpClient.get(this.Api+'getUnidades');
}
//obtener la unidad por el id
getUnidad($id){
  return this.httpClient.get(this.Api+'getUnidad/'+$id);
}
```

Elaborado por: El Investigador.

Ilustración 97 Peticiones para periodos lectivos.

```
///años lectivos
//obtener listado de años lectivos
getLectivos(id:any){
  return this.httpClient.get(this.Api+'getLectivos/'+id);
}
//insertar nuevo año lectivo
postLectivo(lectivo:FormData){
  return this.httpClient.post(this.Api+'insertarLectivo',lectivo);
}
//eliminar lectivo
eliminarLectivo(lectivo:FormData){
  return this.httpClient.post(this.Api+'eliminarLectivo',lectivo);
}
//editar lectivo
editarLectivo(lectivo:FormData){
  return this.httpClient.post(this.Api+'editarLectivo',lectivo);
}
```

Elaborado por: El Investigador.

Ilustración 98 Peticiones para niveles.

```
//obtener listado de niveles x año lectivo
getNiveles(id){
  return this.httpClient.get(this.Api+'getNiveles/'+id);
}
//insertar nivel
insertarNivel(nivel:FormData){
  return this.httpClient.post(this.Api+'insertarNivel',nivel);
}
//eliminar nivel
borrarNivel(id){
  return this.httpClient.delete(this.Api+'borrarNivel/'+id);
}
//editar nivel
editarNivel(nivel:FormData){
  return this.httpClient.post(this.Api+'editarNivel',nivel);
}
```

Elaborado por: El Investigador.

Ilustración 99 Peticiones paralelos.

```
//variables
Api='http://127.0.0.1:8000/api/';
//obtener listado de paralelos x nivel
getParalelos(id){
  return this.httpClient.get(this.Api+'getParalelos/'+id);
}
//paralelos
//subirParalelo
postParalelo(paralelo:FormData){
  return this.httpClient.post(this.Api+'subirParalelo',paralelo);
}
//eliminarParalelo
borrarParalelo(id:any){
  return this.httpClient.delete(this.Api+'borrarParalelo/'+id);
}
//editarParalelo
editarParalelo(paralelo:FormData){
  return this.httpClient.post(this.Api+'editarParalelo',paralelo);
}
```

Elaborado por: El Investigador.

Ilustración 100 Peticiones para los archivos del paralelo.

```
//Archivos
//subirArchivo
postArchivo(archivo:FormData){
  return this.httpClient.post(this.Api+'subirArchivoParalelo',archivo);
}
//cargarArchivos
getArchivos(id:any){
  return this.httpClient.get(this.Api+'getArchivosParalelo/'+id);
}
//descargar
getdownloadArchivo(idParalelo:any,formato:any,nombre:any,tipoDocumento:any):Observable<Blob>{
  return this.httpClient.get(this.Api+'descargarArchivoParalelo?formato='+formato+'&nombre='+nombre+'&tipoDocumento='+tipoDocumento+'&idParalelo'+idParalelo);
}
//eliminarArchivo
borrarArchivo(nombre:any,idParalelo:any,tipoDocumento:any){
  return this.httpClient.delete(this.Api+'borrarArchivoParalelo?nombre='+nombre+'&idParalelo='+idParalelo+'&tipoDocumento='+tipoDocumento);
}
```

Elaborado por: El Investigador.

Ilustración 101 Peticiones para la documentación institucional.

```
////Documentos Carpetas
//listar las carpetas o documentos
getDocumentos(id:any){
  return this.httpClient.get(this.Api+'getDocs/'+id);
}
//insertar carpeta
postCarpeta(carpetas){
  return this.httpClient.post(this.Api+'insertarCarpeta',carpetas);
}
//eliminarCarpeta
deleteCarpeta(carpetas:FormData){
  return this.httpClient.post(this.Api+'eliminarCarpeta',carpetas);
}
//editarCarpeta
editarCarpeta(carpetas:FormData){
  return this.httpClient.post(this.Api+'editarCarpeta',carpetas);
}
```

Elaborado por: El Investigador.

Ilustración 102 Peticiones para los archivos de la documentación.

```
//Archivos
//subirArchivo
postArchivo(archivo:FormData){
  return this.httpClient.post(this.Api+'subirArchivo',archivo);
}
getArchivos(id:any){
  return this.httpClient.get(this.Api+'getArchivos/'+id);
}
//descargar
getdownloadArchivo(formato:any,nombre:any,idCarpeta:any):Observable<Blob>{
  return this.httpClient.get(this.Api+'descargarArchivo?formato='+formato+'&nombre='+nombre+'&idCarpeta='+idCarpeta, { responseType: "blob" });
}

//eliminar
postborrarArchivo(ruta:FormData){
  return this.httpClient.post(this.Api+'borrarArchivo',ruta);
}
```

Elaborado por: El Investigador.

Ilustración 103 Peticiones para reportes.

```
//getReportDocumentos
getreporteDocumentos(idlectivo:any){
  return this.httpClient.get(this.Api+'reporteDocumentos/'+idlectivo);
}
//getReportEstudiantesxNivel
getreporteEstudiantesxNivel(idNivel:any){
  return this.httpClient.get(this.Api+'reporteNivel/'+idNivel);
}
//getReportEstudiantesxParalelo
getreporteEstudiantesxParalelo(idNivel:any){
  return this.httpClient.get(this.Api+'reporteParalelo/'+idNivel);
}
```

Elaborado por: El Investigador.

Ilustración 104 Método para el ingreso de usuario.

```
async ingresar(){
  let formData = new FormData();
  this.presentLoading();
  formData.append("email",this.usuarioForm.value.user);
  formData.append("password",this.usuarioForm.value.password);
  this.authService.getLogin(formData).toPromise().then(res=>{
    if(res['HttpResponse'].status==201){
      this.hideLoader();
      this.presentToast(res['HttpResponse'].message);
      this.usuarioConectado=res['user'];
      localStorage.setItem('exit','1');
      localStorage.setItem('token',res['token']);
      localStorage.setItem('viewmenu','1');
      localStorage.setItem('usuario',this.encriptar.EncriptarObjeto(JSON.stringify(res['user']),environment.key));
      this.cerrarModal();
      this.router.navigate(['/inicio']);
      window.location.reload();
    }else{
      this.hideLoader();
      this.presentToast(res['HttpResponse'].message);
    }
  });
}
```

Elaborado por: El Investigador.

Ilustración 105 Método de navegación según el rol de usuario.

```
let uni = localStorage.getItem('usuario');
if(uni==null){
  this.correoUsuario="Correo de Usuario";
  this.nombreUsuario="Nombre del Usuario";
  this.rolUsuario="Rol del Usuario";
}else{
  this.Usuario=JSON.parse(this.encryptar.DescriptarObjeto(uni,environment.key));
  this.correoUsuario=this.Usuario['email'];
  this.nombreUsuario=this.Usuario['NOMBRE'];
  this.tipoUsuario=this.Usuario['TIPO_USUARIO'];
  if(this.tipoUsuario==1){
    //administrador
    this.rolUsuario="Administrador"
    this.appPages = [
      { title: 'Inicio', url: '/inicio', icon: 'home' },
      { title: 'Escuelas', url: '/escuelas', icon: 'school' },
      { title: 'Buscar', url: '/buscador', icon: 'search-circle' },
      { title: 'Usuarios', url: '/usuarios', icon: 'people-circle' },
      { title: 'Reportes', url: '/reportes', icon: 'notifications-circle' },
      { title: 'Noticias', url: '/noticias', icon: 'newspaper' }
    ];
  }else if(this.tipoUsuario==3){
    //admin unidad educativa
    this.rolUsuario="Administrador_UE"
    this.appPages = [
      { title: 'Inicio', url: '/inicio', icon: 'home' },
      { title: 'Unidad Educativa', url: '/unidad-educativa', icon: 'school' },
      { title: 'Buscar', url: '/buscador', icon: 'search-circle' },
      { title: 'Reportes', url: '/reportes', icon: 'notifications-circle' },
    ];
  }else{
    //rector
    this.rolUsuario="Rector"
    this.appPages = [
      { title: 'Inicio', url: '/inicio', icon: 'home' },
      { title: 'Unidad Educativa', url: '/unidad-educativa', icon: 'school' },
      { title: 'Buscar', url: '/buscador', icon: 'search-circle' },
      { title: 'Reportes', url: '/reportes', icon: 'notifications-circle' }
    ];
  }
}
```

Elaborado por: El Investigador.

Ilustración 106 Guards para el control de navegación.

```
canActivate(
  route: ActivatedRouteSnapshot,
  state: RouterStateSnapshot): Observable<boolean | UrlTree> | Promise<boolean | UrlTree> | boolean | UrlTree {
  let user = localStorage.getItem('usuario');
  this.usuario=JSON.parse(this.encryptar.DescriptarObjeto(user,environment.key));
  if(this.usuario.TIPO_USUARIO!=2){
    return true;
  }else{
    //no esta Logueado
    this.router.navigate(['/inicio']);
    return false;
  }
}
```

Elaborado por: El Investigador.

Ilustración 107 Rutas de Navegación.

```
const routes: Routes = [
  {
    path: 'inicio',
    loadChildren: () => import('./inicio/inicio.module').then( m => m.InicioPageModule)
  },
  {
    path: '',
    redirectTo: 'inicio',
    pathMatch: 'full'
  },
  {
    path: 'folder/:id',
    loadChildren: () => import('./folder/folder.module').then( m => m.FolderPageModule),
  },
  {
    path: 'escuelas',
    canActivate:[AuthGuard,AdminGuard],
    loadChildren: () => import('./escuelas/escuelas.module').then( m => m.EscuelasPageModule),
  },
  {
    path: 'unidade-insert',
    canActivate:[ModalsGuard],
    loadChildren: () => import('./unidade-insert/unidade-insert.module').then( m => m.UnidadeInsertPageModule),
  },
  {
    path: 'memudoc',
    canActivate:[AuthGuard,MemudocGuard],
    loadChildren: () => import('./memudoc/memudoc.module').then( m => m.MemudocPageModule),
  },
  {
    path: 'menulectivo',
    canActivate:[ModalsGuard],
    loadChildren: () => import('./menulectivo/menulectivo.module').then( m => m.MenulectivoPageModule),
  },
  {
    path: 'documentos',
    canActivate:[AuthGuard,DocumentosGuard],
  }
]
```

Elaborado por: El Investigador.

Métodos para subida, descarga y visualización de archivos.

Ilustración 108 Método para subir Archivos.

```
onFileSelected(event) {  
  this.numeroArchivos=0;  
  for (let i = 0; i < event.target.files.length; i++) {  
    let data = event.target.files[i];  
    this.listaArchivos.push(event.target.files[i]);  
    this.numeroArchivos++;  
    let reader = new FileReader();  
    reader.readAsDataURL(data);  
  }  
  if(this.listaArchivos.length==0){  
    this.condicion=true;  
  }else{  
    this.condicion=false;  
  }  
}  
  
async subir() {  
  this.presentLoading();  
  this.num = 0;  
  for (let index = 0; index < this.listaArchivos.length; index++) {  
    const newFormData = new FormData();  
  
    newFormData.append('nombreArchivo', this.listaArchivos[index].name);  
    newFormData.append('formato', this.listaArchivos[index].type);  
    newFormData.append('archivo', this.listaArchivos[index]);  
    newFormData.append('idParalelo', this.idParalelo);  
    newFormData.append('tipoDocumento', this.tipoDocumento);  
    let respuesta = await this.paraleloService.postArchivo(newFormData).toPromise().then(res=>{  
      if (res['HttpResponse'].status==200) {  
        this.num++;  
      }else{  
        this.presentToast(res['HttpResponse'].message);  
      }  
    });  
  }  
  return this.num;  
}
```

Elaborado por: El Investigador.

Ilustración 109 Método para descargar el Archivo.

```
descargarArchivo(archivo) {  
  this.paraleloService.getdownloadArchivo(this.idParalelo, archivo.FORMATO, archivo.NOMBRE_ARCHIVO, archivo.TIPO_DOCUMENTO).subscribe(  
    res => {  
      //console.log(res);  
      var newBlob = new Blob([res], { type: archivo.FORMATO });  
      if (window.navigator && window.navigator.msSaveOrOpenBlob) {  
        window.navigator.msSaveOrOpenBlob(newBlob);  
        return;  
      }  
  
      const data = window.URL.createObjectURL(newBlob);  
      var link = document.createElement('a');  
      link.href = data;  
      link.download = archivo.NOMBRE_ARCHIVO;  
      link.dispatchEvent(new MouseEvent('click', { bubbles: true, cancelable: true, view: window }));  
      setTimeout(function () {  
        window.URL.revokeObjectURL(data);  
      }, 100);  
    }  
  );  
}
```

Elaborado por: El Investigador.

Ilustración 110 Método para Visualizar el Archivo.

```
//visualizar archivo  
visualizarArchivo(archivo) {  
  this.paraleloService.getdownloadArchivo(this.idParalelo, archivo.FORMATO,  
    archivo.NOMBRE_ARCHIVO, archivo.TIPO_DOCUMENTO).  
  subscribe(  
    result=>{  
      var url = window.URL.createObjectURL(result);  
      window.open(url);  
    }  
  );  
}
```

Elaborado por: El Investigador.

Ilustración 111 Método para eliminar un archivo.

```
//eliminarArchivoSeleccionado
eliminarArchivo(archivo){
  let form = new FormData();
  form.append('formato',this.parametrosIn['idDocumento']);
  form.append('nombre',archivo.NOMBRE_ARCHIVO);
  this.paraleloService.borrarArchivo(archivo.NOMBRE_ARCHIVO,this.idParalelo,archivo.TIPO_DOCUMENTO).
  toPromise().then(
    res=>{
      if(res['HttpResponse'].status==200){
        this.eliminado();
      }
    }
  );
}
```

Elaborado por: El Investigador.

Ilustración 112 Métodos para el cifrado de información.

```
constructor() { }
EncriptarSimple(mensaje: string, llave: string){
  return CryptoJS.AES.encrypt(mensaje, llave);
}

DesencriptarSimple(mensaje: string, llave: string){
  const desencript = CryptoJS.AES.decrypt(mensaje, llave);
  return desencript.toString(CryptoJS.enc.Utf8);
}

DesencriptarObjeto(object: any, llave: string){
  const decrypt = CryptoJS.AES.decrypt(object, llave);
  return JSON.parse(decrypt.toString(CryptoJS.enc.Utf8));
}

EncriptarObjeto(objeto: any, llave: string){
  return CryptoJS.AES.encrypt(JSON.stringify(objeto), llave).toString();
}
```

Elaborado por: El Investigador.

3.2.6.11. Pruebas

La fase de pruebas es una de las partes más importantes del proyecto. Para poder determinar el correcto funcionamiento de la aplicación es de vital importancia la evaluación de las pruebas de usuario en busca de errores e inconsistencias que pueden alterar el funcionamiento.

Tabla 104. Prueba de Aceptación Historia 1

Prueba de aceptación	
Número: 1	Historia de Usuario: 01
Nombre: Roles de Usuario	
Revisado por: Ing. Bolívar Alcocer-Tester	
Descripción: Permitir a los usuarios tener un cierto nivel de acceso y privilegios al manejar la aplicación.	
Condiciones de ejecución: Los administradores distritales tienen acceso sobre todo el sistema. Y se puede eliminar administradores distritales hasta que el sistema cuente con 1.	
Entrada: Solo los administradores distritales, pueden asignar los roles a los demás usuarios.	
Resultado esperado: El administrador distrital puede seleccionar roles para los demás usuarios.	
Evaluación de la prueba: Prueba satisfactoria	

Elaborado por: El Investigador.

Tabla 105. Prueba de Aceptación Historia 4

Prueba de aceptación	
Número: 2	Historia de Usuario: 04
Nombre: Presentación para administradores distritales.	
Revisado por: Ing. Bolívar Alcocer-Tester	
Descripción: El usuario visualiza el listado de unidades educativas registradas, además puede, buscar, ingresar, actualizar, eliminar una unidad educativa	
Condiciones de ejecución: Solo los usuarios con el rol de administrador distrital pueden visualizar la información.	
Entrada: Los usuarios dan click para acceder a la información de la unidad educativa que seleccionen. Además de gestionar la información de las unidades educativas.	
Resultado esperado: El usuario puede gestionar la información de las unidades educativas.	
Evaluación de la prueba: Prueba satisfactoria	

Elaborado por: El Investigador.

Tabla 106. Prueba de Aceptación Historia 5

Prueba de aceptación	
Número: 3	Historia de Usuario: 05
Nombre: Menú unidad educativa.	
Revisado por: Ing. Bolívar Alcocer-Tester	
Descripción: Pantalla para la selección de información a la que desea acceder.	
Condiciones de ejecución: El usuario debe seleccionar anteriormente la unidad educativa.	
Entrada: Los usuarios seleccionan la información que desean acceder.	
Resultado esperado: El usuario puede visualizar el menú de la unidad educativa y seleccionar la categoría que desea.	
Evaluación de la prueba: Prueba satisfactoria	

Elaborado por: El Investigador.

Tabla 107. Prueba de Aceptación Historia 6

Prueba de aceptación	
Número: 4	Historia de Usuario: 06
Nombre: Periodos/ años lectivos.	
Revisado por: Ing. Bolívar Alcocer-Tester	
Descripción: Pantalla para la gestión de periodos lectivos.	
Condiciones de ejecución: El usuario debe seleccionar anteriormente la categoría a la que desea acceder (boletas, documentación, promociones).	
Entrada: Los usuarios pueden seleccionar, ingresar, actualizar o eliminar los periodos lectivos con los que cuente la unidad educativa.	
Resultado esperado: El usuario puede visualizar el menú de los periodos lectivos y gestionar la información de los mismos.	
Evaluación de la prueba: Prueba satisfactoria	

Elaborado por: El Investigador.

Tabla 108. Prueba de Aceptación Historia 7

Prueba de aceptación	
Número: 5	Historia de Usuario: 07
Nombre: Niveles/Cursos.	
Revisado por: Ing. Bolívar Alcocer-Tester	
Descripción: Pantalla para la gestión de niveles o cursos.	
Condiciones de ejecución: El usuario debe seleccionar anteriormente el periodo lectivo al que quiere acceder.	
Entrada: Los usuarios pueden seleccionar, ingresar, actualizar o eliminar los niveles o cursos que maneje el periodo lectivo anteriormente seleccionado, además de clasificarlos según la jornada.	
Resultado esperado: El usuario puede visualizar el menú de los niveles según la jornada además de poder gestionar los mismos.	
Evaluación de la prueba: Prueba satisfactoria	

Elaborado por: El Investigador.

Tabla 109. Prueba de Aceptación Historia 8

Prueba de aceptación	
Número: 6	Historia de Usuario: 08
Nombre: Paralelos.	
Revisado por: Ing. Bolívar Alcocer-Tester	
Descripción: Pantalla para la gestión de paralelos.	
Condiciones de ejecución: El usuario debe seleccionar anteriormente el nivel al que quiere acceder.	
Entrada: Los usuarios pueden seleccionar, ingresar, actualizar o eliminar los paralelos que maneje el nivel anteriormente seleccionado.	
Resultado esperado: El usuario puede visualizar el menú de los paralelos además de poder gestionar los mismos.	
Evaluación de la prueba: Prueba satisfactoria	

Elaborado por: El Investigador.

Tabla 110. Prueba de Aceptación Historia 9

Prueba de aceptación	
Número: 7	Historia de Usuario: 09
Nombre: Contenido Paralelo.	
Revisado por: Ing. Bolívar Alcocer-Tester	
Descripción: Pantalla para la gestión de archivos del paralelo.	
Condiciones de ejecución: El usuario debe seleccionar el paralelo al que quiere acceder.	
Entrada: Los usuarios pueden seleccionar los archivos que se encuentren dentro del paralelo como boletas, promociones y concentrado. Puede visualizarlos o descargarlos. Además, pueden ingresar nuevos archivos o eliminarlos.	
Resultado esperado: El usuario puede visualizar los archivos del paralelo además puede ingresar uno o varios archivos a la vez y también eliminarlos. Puede visualizar y descargar los archivos.	
Evaluación de la prueba: Prueba satisfactoria	
Elaborado por: El Investigador.	

Tabla 111. Prueba de Aceptación Historia 10

Prueba de aceptación	
Número: 8	Historia de Usuario: 10
Nombre: Documentación Institucional.	
Revisado por: Ing. Bolívar Alcocer-Tester	
Descripción: Pantalla para la gestión de categorías para la información institucional.	
Condiciones de ejecución: El usuario debe seleccionar el periodo lectivo al que desea acceder.	
Entrada: Los usuarios seleccionan la categoría de documentos a los que desea acceder. Además, pueden ingresar, actualizar o eliminar categorías.	
Resultado esperado: El usuario puede visualizar el menú de categorías de documentación institucional. Además, puede gestionar las categorías.	
Evaluación de la prueba: Prueba satisfactoria	
Elaborado por: El Investigador.	

Tabla 112. Prueba de Aceptación Historia 11

Prueba de aceptación	
Número: 9	Historia de Usuario: 11
Nombre: Archivos documentación institucional.	
Revisado por: Ing. Bolívar Alcocer-Tester	
Descripción: Pantalla para la gestión de archivos dentro de las categorías de documentación institucional.	
Condiciones de ejecución: El usuario debe seleccionar la categoría a la que desea acceder.	
Entrada: Los usuarios pueden seleccionar el archivo que deseen visualizar o descargar. Además, pueden ingresar nuevos archivos o eliminarlos.	
Resultado esperado: El usuario puede visualizar los archivos de la categoría además puede ingresar uno o varios archivos a la vez y también eliminarlos. Puede visualizar y descargar los archivos.	
Evaluación de la prueba: Prueba satisfactoria	
Elaborado por: El Investigador.	

Tabla 113. Prueba de Aceptación Historia 12

Prueba de aceptación	
Número: 10	Historia de Usuario: 12
Nombre: Presentación administradores institucionales.	
Revisado por: Ing. Bolívar Alcocer-Tester	
Descripción: Pantalla de presentación de la unidad educativa para administradores institucionales.	
Condiciones de ejecución: El usuario debe ingresar al sistema y seleccionar la opción Unidad Educativa.	
Entrada: Los usuarios pueden dar click sobre el logotipo de la unidad educativa para acceder al menú de información.	
Resultado esperado: El usuario puede visualizar la información de la unidad educativa y acceder al menú de documentación.	
Evaluación de la prueba: Prueba satisfactoria	
Elaborado por: El Investigador.	

Tabla 114. Prueba de Aceptación Historia 13

Prueba de aceptación	
Número: 11	Historia de Usuario: 13
Nombre: Buscador de archivos.	
Revisado por: Ing. Bolívar Alcocer-Tester	
Descripción: Pantalla para la búsqueda de archivos.	
Condiciones de ejecución: El usuario debe ingresar al sistema y seleccionar la opción Buscar.	
Entrada: Los usuarios seleccionan el tipo de documento que buscan, seleccionar la institución educativa, el periodo lectivo, la jornada, el nivel, el paralelo para posteriormente seleccionar el archivo con el nombre del estudiante que desean visualizar o descargar. Si busca el concentrado solo debe seleccionar el paralelo. Si busca documentación deberá seleccionar el periodo y la categoría para que se presente la lista de archivos a seleccionar.	
Resultado esperado: El usuario puede visualizar y descargar la información que necesita con las diferentes opciones de filtrado.	
Evaluación de la prueba: Prueba satisfactoria	

Elaborado por: El Investigador.

Tabla 115. Prueba de Aceptación Historia 14

Prueba de aceptación	
Número: 12	Historia de Usuario: 14
Nombre: Gestión de usuarios.	
Revisado por: Ing. Bolívar Alcocer-Tester	
Descripción: Pantalla para la gestión de usuarios.	
Condiciones de ejecución: El usuario debe ingresar al sistema y seleccionar la opción Usuarios.	
Entrada: Los administradores distritales pueden visualizar el listado de usuarios en el sistema. Pueden ingresar nuevos usuarios con un rol para el acceso a la información, además actualizar su información y eliminar un usuario. Siempre y cuando no sea el único administrador distrital en la base de datos.	
Resultado esperado: El usuario puede visualizar la información y gestionar la información de los usuarios.	
Evaluación de la prueba: Prueba satisfactoria	

Elaborado por: El Investigador.

Tabla 116. Prueba de Aceptación Historia 15

Prueba de aceptación	
Número: 13	Historia de Usuario: 15
Nombre: Reportes.	
Descripción: Pantalla para la emisión de reportes.	
Revisado por: Ing. Bolívar Alcocer-Tester	
Condiciones de ejecución: El usuario debe ingresar al sistema y seleccionar la opción Reportes.	
Entrada: Los usuarios seleccionan el tipo de reporte que desean generar. Para posteriormente visualizar o descargarlo.	
Resultado esperado: El usuario puede visualizar o descargar el reporte seleccionado.	
Evaluación de la prueba: Prueba satisfactoria	

Elaborado por: El Investigador.

Tabla 117. Prueba de Aceptación Historia 16

Prueba de aceptación	
Número: 14	Historia de Usuario: 16
Nombre: Noticias/notificaciones.	
Revisado por: Ing. Bolívar Alcocer-Tester	
Descripción: Pantalla para la gestión de noticias.	
Condiciones de ejecución: El usuario debe ingresar al sistema y seleccionar la opción Noticias.	
Entrada: Los administradores distritales pueden visualizar el listado de noticias que se presentarán en el inicio de la aplicación, además podrán ingresar y eliminar nuevas noticias.	
Resultado esperado: El usuario puede visualizar y gestionar las noticias para la presentación del sistema.	
Evaluación de la prueba: Prueba satisfactoria	

Elaborado por: El Investigador

Tabla 118. Prueba de Aceptación Historia 17

Prueba de aceptación	
Número: 15	Historia de Usuario: 17
Nombre: Pantalla de inicio.	
Revisado por: Ing. Bolívar Alcocer-Tester	
Descripción: Pantalla de inicio al sistema y presentación de noticias.	
Condiciones de ejecución: Todos los usuarios tendrán acceso a la pantalla de inicio del sistema. Y pueden acceder tras dar click en el botón de ingresar.	
Entrada: Los usuarios pueden visualizar las noticias, si desean acceder al sistema deberán ingresar su usuario y contraseña.	
Resultado esperado: Los usuarios pueden visualizar las noticias o eventos del distrito, además pueden acceder al sistema mediante la opción de ingreso.	
Evaluación de la prueba: Prueba satisfactoria	

Elaborado por: El Investigador.

Tabla 119. Prueba de Aceptación Historia 18

Prueba de aceptación	
Número: 16	Historia de Usuario: 18
Nombre: Acceso a la aplicación.	
Revisado por: Ing. Bolívar Alcocer-Tester	
Descripción: Controlar y validar el acceso a la aplicación según el rol de usuario.	
Condiciones de ejecución: Los usuarios que ingresen su usuario y contraseña para el ingreso al sistema.	
Entrada: Controlar que los usuarios naveguen solo en opciones que su rol lo permita, validaciones de enrutamiento.	
Resultado esperado: Los usuarios pueden navegar de manera correcta en las diferentes ventanas del sistema.	
Evaluación de la prueba: Prueba satisfactoria	

Elaborado por: El Investigador.

CAPITULO IV: CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

- Con el desarrollo del proyecto se pudo implementar un repositorio digital mediante el manejo de una aplicación web progresiva, que permite a los empleados del departamento de tecnologías del distrito obtener la documentación de las diferentes unidades educativas fiscales de una manera más rápida y segura, con la capacidad de almacenarla en un servidor y reducir drásticamente la posibilidad de pérdida de información.
- En base a la información recopilada en la investigación realizada sobre el manejo de repositorios y bibliotecas virtuales en instituciones tanto públicas como privadas, se pudo definir que el tipo de repositorio que se implemente va acorde al tipo de información que se quiera almacenar. Ya sean repositorios institucionales, multimedia, de investigación entre otros, pero se deben definir nuevos conceptos para los temas relacionados a repositorios. El manejo de información de varias instituciones mediante un repositorio conllevaría más a un concepto de repositorio multiinstitucional o para el caso presentado el concepto de un repositorio distrital.
- Mediante el análisis de las herramientas que permiten el desarrollo de una aplicación web progresiva, por su eficacia y características se determina el manejo de Ionic como herramienta ya que permite la creación de aplicaciones híbridas y progresivas que pueden ser utilizadas en dispositivos móviles dando a los usuarios la posibilidad de realizar las tareas que antes las podían realizar solamente en su computador en cualquier momento en cualquier lugar donde se encuentren.
- La aplicación permite al personal del distrito designar una persona responsable en cada unidad educativa fiscal de Ambato la tarea de gestionar la documentación de su institución. A tal manera que se encarguen de proveer los archivos tanto de estudiantes como Boletas, Promociones, Concentrado y de documentos institucionales al repositorio digital, para posteriormente ser verificada y expedir

un informe de cumplimiento a la institución. Reduciendo de gran manera la carga laboral para el departamento del distrito.

- La aplicación permite a los rectores de las unidades educativas acceder a la información que necesiten en cualquier momento en cualquier lugar que se encuentren, ya sea información de estudiantes o información de su institución. Evitando la necesidad de comunicarse con el departamento del distrito. Además, la aplicación permite dar a conocer noticias o eventos que conciernen a la comunidad educativa a todas las personas que visiten el sitio del repositorio digital.
- En la Unidad Distrital de Tecnologías de Información y Comunicaciones del Distrito 18D01-Ambato 1 Educación al contar con el desarrollo de la Aplicación Web Progresiva ha mejorado el manejo de la documentación por parte de los usuarios ya que cada uno de ellos fueron evaluados desde el inicio del desarrollo obteniendo resultados y pruebas exitosas además de que permite la instalación directa desde el navegador y facilita el uso desde cualquier tipo de dispositivo.

4.2. Recomendaciones

- Se debe tomar en cuenta que la aplicación permite designar a una persona responsable en cada unidad educativa y asignando el rol respectivo para que proporcione la documentación de su unidad educativa de manera directa, con el fin de no tener inconsistencias en la información requerida.
- En el transcurso del periodo lectivo posterior a la recolección de información debe existir un proceso de análisis, verificación y validación de documentación, así como realizar respaldos constantes para conservar los datos.
- Para el correcto manejo de la aplicación web progresiva se recomienda utilizar un navegador actualizado ya que este mejoraría de gran manera la interacción del usuario.
- La aplicación permite poder acceder a la información de una unidad educativa en cualquier momento, pero también cuenta con la opción de poder agregar

información es recomendable notificar a la unidad distrital si se presenta algún fallo, error, inconsistencia en la documentación.

- Es recomendable permanecer en constante capacitación a los usuarios sobre el uso de las aplicaciones web progresivas con el fin de aumentar su usabilidad.

BIBLIOGRAFÍA

- [1] J. M. G. T. C. Barrueco, «REPOSITARIOS INSTITUCIONALES UNIVERSITARIOS: EVOLUCIÓN Y PERSPECTIVAS,» *Fesabid*, vol. 09, pp. 99-107, 2009.
- [2] F. B. D. Santos, «Diseño e implementación de un portal Web y un repositorio virtual de contenido multimedia académico para el área de Ingeniería Electrónica y Redes,» Universidad de las Fuerzas Armadas ESPE, Latacunga, 2014.
- [3] M. P. P. V. M. T. S. J. L. ANDRADE, «DISEÑO E IMPLEMENTACIÓN DE UNA BIBLIOTECA VIRTUAL BASADA EN UNIVERSIDADES,» CORPORACIÓN UNIVERSITARIA TECNOLÓGICA DE BOLIVAR, Cartagena, 2010.
- [4] D. D. P. Á. L. Y. B. P. M. A. O. Agüero, «Repositorio Institucional de contenido educativo de la Escuela,» *Panorama Cuba y Salud*, vol. 10, pp. 2-9, 2015.
- [5] B. A. P. Calderón, «Diseño y Aplicación de una biblioteca virtual de tesis digitales de la Escuela de Ingeniería Industrial,» Escuela Superior Politécnica de Chimborazo, Riobamba, 2009.
- [6] R. H. y. M. Hernández, «Diseño e Implementación de la Biblioteca Virtual especializada sobre los derivados de la caña de azúcar,» Instituto Cubano de Investigaciones de los Derivados de la Caña de Azúcar, La Habana, 2009.
- [7] W. O. T. Ochoa, «REPOSITARIO VIRTUAL DE LA BIBLIOTECA DE UNIANDES,» UNIVERSIDAD REGIONAL AUTÓNOMA DE LOS ANDES, Babahoyo, 2016.
- [8] L. F. S. Escobar, «Gestión Documental enfocada a procesos: una mirada desde la administración pública distrital,» *Revista Interamericana de Bibliotecología*, vol. 35, n° 1, pp. pp. 243-255, 2012.
- [9] L. I. G. Pérez, «Repositorios como soportes para diseminar experiencias de innovación,» Sistema Nacional de Investigadores, Mexico, 2017.
- [10] EducarEcuador, 01 Enero 2020. [En línea]. Available: <https://ecuadorec.com/ingresar-plataforma-carmenta-ministerio-educacion/#:~:text=La%20plataforma%20Carmenta%20disponible%20v%C3%99,> [Último acceso: 04 Mayo 2021].
- [11] A. H. TRASOBARES, «LOS SISTEMAS DE INFORMACIÓN:EVOLUCIÓN,» *Revista de relaciones laborales*, vol. 10, pp. 149-165, 2003.

- [12] F. Alarcón, «Desarrollo de sistemas de información: una metodología basada en el modelado,» Vicenc, Mexico, 2006.
- [13] R. N. Martínez, «El Proceso de Desarrollo de Software: 2ª Edición,» IT Campus Academy, Estados Unidos, 2017.
- [14] M. ARRIBAS DEL POZO, «Sistemas de archivo y clasificación de documentos,» Paraninfo, España, 2015.
- [15] N. G. L. D. L. C. P. P. Thomas, «Desarrollo de Aplicaciones Multiplataforma,» Universidad Nacional de La Plata, Buenos Aires, 2018.
- [16] P. M. V. M. R. M. C. G. A. M. G. R. A. Rodríguez, «Aplicaciones Web Progresivas Enfocadas en el Uso y Optimización de Cache,» Universidad Abierta Interamericana, Buenos Aires, 2020.
- [17] J. Lafosse, «Struts 2: El framework de desarrollo de aplicaciones Java EE,» ENI, España, 2010.
- [18] I. a. JavaScript, «Javier Eguíluz Pérez,» España, librosweb, 2019, p. 5.
- [19] E. P. Farinango Caiza, «Estudio del framework angular para desarrollar aplicaciones single-page. Desarrollo del sistema de información, seguimiento y control para la hacienda ganadera La Vega,» Universidad Técnica del Norte , Otavalo, 2020.
- [20] H. C. L. Fernando, «DESARROLLO DE APLICACIONES MÓVILES HIBRIDAS CON IONIC,» *Revista Científica del departamento de Infotmática y sistemas*, pp. 7-8, 2017.
- [21] H. Spona, Programación de Base de Datos con Mysql y PHP, Barcelona: Marcombo, 2010.
- [22] D. J. O. Ortega, «“Bootstrap y Laravel, herramientas para el desarrollo de aplicaciones web,» UNIVERSIDAD POLITÉCNICA DE SINALOA, Mexico, 2019.
- [23] Á. Arias, «Bases de Datos con MySQL: 2ª Edición,» IT Campus Academy, España, 2014.
- [24] L. L. A. Z. G. y. L. E. R. R. L. O. R. Vidot, «EPrints y la creación de repositorios virtuales. Estudio de un caso. La biblioteca virtual de la Facultad de Matemática y Computación de la Universidad de La Habana,» Universidad de La Habana, La Habana, 2007.
- [25] R. G. Figueroa, «Metodologías Tradicionales vs Metodologías Ágiles,» Universidad Tecnica Particular de Loja, Loja, 2008.


- [26] G. Vidal, «ESTATUTO ORGANICO POR PROCESOS DEL MINISTERIO DE EDUCACION,» Lexis, Quito, 2012.
- [27] IONOS, «Digital Guide IONOS,» 25 08 2020. [En línea]. Available: <https://www.ionos.es/digitalguide/servidores/know-how/que-es-el-cloud/>. [Último acceso: 08 12 2021].
- [28] P. Letelier, «Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP),» Universidad Politécnica de Valencia, Valencia, 2006.

ANEXOS Y APÉNDICES

ANEXO A

Documentos Académicos

Boletas



UNIDAD EDUCATIVA AMBATO DE LOS ANGELES
 Educación General Básica Básica Superior
REPORTE DE EVALUACIÓN
 Año Lectivo: 2021-2022

Ministerio de Educación
 Régimen: SIERRA

Datos Informativos del Estudiante:

Nombre: CHIMBORAZO TUQUERES ANDERSON FABRICIO	Grado/Paralelo: 10mo año Básica A	Institución: UNIDAD EDUCATIVA AMBATO DE LOS ANGELES
Código Mineduc: 0250340247	Jornada: MATUTINA	Modalidad: PRESENCIAL

REPORTE ANUAL DE CALIFICACIONES																							
Áreas	Asignatura	PRIMER QUIMESTRE								SEGUNDO QUIMESTRE								PROMEDIO ANUAL		ASISTENCIA			
		P1	P2	PROM	%PRO (80%)	EXAM	%EX (20%)	QUI	EC	P1	P2	PROM	%PRO (80%)	EXAM	%EX (20%)	QUI	EC	SUPLE	PA	EC	H.J	H.I	ASIS
LENGUA Y LITERATURA	LENGUA Y LITERATURA	7.00		3.50	2.80			2.80	NA									0.00	1.40	NA			
EDUCACIÓN CULTURAL Y ARTÍSTICA	EDUCACIÓN CULTURAL Y ARTÍSTICA																	0.00	0.00	NA	0	0	48
MATEMÁTICA	MATEMÁTICA																	0.00	0.00	NA			
CIENCIAS SOCIALES	ESTUDIOS SOCIALES																	0.00	0.00	NA	0	0	93
EDUCACIÓN FÍSICA	EDUCACIÓN FÍSICA	6.63	7.94	7.28	5.82	7.00	1.40	7.22	AA									0.00	3.61	NA	0	0	116
CIENCIAS NATURALES	CIENCIAS NATURALES																	0.00	0.00	NA	0	0	92
LENGUA EXTRANJERA	INGLÉS	6.69		3.34	2.67			2.67	NA									0.00	1.33	NA	0	0	117
PROMEDIO GENERAL		2.90	1.13	4.70	1.61	1.00	0.20	1.81		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.90					

COMPORTAMIENTO	QUIMESTRE 1		QUIMESTRE 2	
	P1	P2	P1	P2

ASIGNATURA	QUIMESTRE 1		QUIMESTRE 2	
	P1	P2	P1	P2

PROYECTOS ESCOLARES	MB

COMPORTAMIENTO	PROYECTOS ESCOLARES	ESCALA CUALITATIVA CUANTITATIVA	ESCALA
A = Muy satisfactorio B = Satisfactorio C = Poco Satisfactorio D = Marginal E = Insatisfactorio	EX= Excelente MB= Muy Buena B= Buena R= Regular	DA Domina los aprendizajes requeridos 9.00-10.00 AA Alcanza los aprendizajes requeridos 7.00-8.99 PA Próximo a alcanzar los aprendizajes requeridos 4.01-6.99 NA No alcanza los aprendizajes requeridos <= 4	

RECTORIA - DIRECTORIA - LIDER EDUCATIVO	DOCENTE TUTOR	PADRE DE FAMILIA
---	---------------	------------------

Dirección: Av. Amazonas N34-451 y Av. Atahualpa.
 Código postal: 170507 / Quito-Ecuador
 Teléfono: 593-2-396-1300 / www.educacion.gob.ec



Promociones



Ministerio de Educación

CÓDIGO AMIE: 18H00220

AÑO LECTIVO: 2020-2021

RÉGIMEN: SIERRA

CERTIFICADO DE APROBACIÓN

El Rector (a) / Director (a) de la Institución Educativa:

UNIDAD EDUCATIVA AMBATO DE LOS ANGELES

De conformidad con lo prescrito en el Art. 197 del Reglamento General a la Ley Orgánica de Educación Intercultural y demás normativas vigentes, certifica que el/la estudiante

MEJIA ABRIL ALAN MATEO

del TERCER AÑO DE BACHILLERATO GENERAL UNIFICADO

CÓDIGO ESTUDIANTE No 1804767943, obtuvo las siguientes calificaciones durante el presente año lectivo.

ÁREA	ASIGNATURA	PROMEDIO ANUAL	
		CALIFICACIÓN CUANTITATIVA	CALIFICACIÓN CUALITATIVA
LENGUA Y LITERATURA	LENGUA Y LITERATURA	8.29	Alcanza los aprendizajes requeridos
MATEMÁTICA	MATEMÁTICA	9.13	Domina los aprendizajes requeridos
EDUCACIÓN FÍSICA	EDUCACIÓN FÍSICA	8.71	Alcanza los aprendizajes requeridos
CIENCIAS SOCIALES	HISTORIA	8.69	Alcanza los aprendizajes requeridos
CIENCIAS NATURALES	BIOLOGÍA	9.09	Domina los aprendizajes requeridos
CIENCIAS NATURALES	QUÍMICA	9.11	Domina los aprendizajes requeridos
CIENCIAS NATURALES	FÍSICA	8.91	Alcanza los aprendizajes requeridos
LENGUA EXTRANJERA	INGLÉS	8.94	Alcanza los aprendizajes requeridos
MÓDULO INTERDISCIPLINAR	EMPRESARIADO Y GESTIÓN	8.84	Alcanza los aprendizajes requeridos
ASIGNATURAS OPTATIVAS	INVESTIGACIÓN EN CIENCIA Y TECNOLOGÍA OP	8.70	Alcanza los aprendizajes requeridos
ASIGNATURAS OPTATIVAS	LECTURA CRÍTICA DE MENSAJES OP	8.46	Alcanza los aprendizajes requeridos
ASIGNATURAS OPTATIVAS	INFORMÁTICA APLICADA A LA EDUCACIÓN OP	8.79	Alcanza los aprendizajes requeridos
ASIGNATURAS OPTATIVAS	FÍSICA SUPERIOR OP	8.79	Alcanza los aprendizajes requeridos
ASIGNATURAS OPTATIVAS	BIOLOGÍA SUPERIOR OP	9.01	Domina los aprendizajes requeridos
ASIGNATURAS OPTATIVAS	REDACCIÓN CREATIVA OP	8.50	Alcanza los aprendizajes requeridos
PROMEDIO		8.79	Alcanza los aprendizajes requeridos

Concentrado



UNIDAD EDUCATIVA AMBATO DE LOS ANGELES
 Bachillerato Ciencias Ministerio de Educación
CONCENTRADO DE CALIFICACIONES POR PARALELO

Institución: 18H00220 - UNIDAD EDUCATIVA AMBATO DE LOS ANGELES Grado: 3er Año Bachillerato - A Jornada: MATUTINA Modalidad: PRESENCIAL Especialidad:

LISTADO DE ESTUDIANTES	BIOLOGÍA														PRIMERA AVAL	BIOLOGÍA SUPERIOR OP														
	PRIMER QUIMESTRE							SEGUNDO QUIMESTRE								PRIMER QUIMESTRE							SEGUNDO QUIMESTRE							
	P1	P2	PRD M	ZPR O(80%)	EXA	MEVA (20%)	QUI	P1	P2	PRD M	ZPR O(80%)	EXA	MEVA (20%)	QUI		nota supte	P1	P2	PRD M	ZPR O(80%)	EXA	MEVA (20%)	QUI	P1	P2	PRD M	ZPR O(80%)	EXA	MEVA (20%)	QUI
BENAVIDES TUBON MELANY LIZETH	8.88	10.00	9.44	7.95	7.23	1.44	8.99	9.50	9.26	8.97	7.49	8.72	1.74	9.23		9.11	8.68	10.00	9.44	7.95	7.23	1.44	8.99	9.75	9.63	9.69	7.75	8.72	1.74	9.49
CEDERO TAPIA JOSE DAMIAN	7.25	7.50	7.37	5.89	6.00	1.20	7.69	7.75	7.25	7.50	6.00	7.25	1.45	7.45		7.27	7.25	7.50	7.37	5.89	6.00	1.20	7.69	8.00	7.25	7.62	6.09	7.25	1.45	7.54
CHANGO BUNCI MIRIAM JEANETH	8.75	7.50	9.12	6.43	6.33	1.26	7.75	9.00	9.25	9.12	7.29	7.59	1.51	8.80		8.27	8.75	7.50	9.12	6.43	6.33	1.26	7.75	9.00	9.13	8.56	6.84	7.59	1.51	8.35
CHANGO YOPAMPA DIANA ABIGAIL	9.12	9.00	9.06	7.24	5.20	1.04	8.20	8.75	8.53	9.19	7.35	7.89	1.57	8.92		8.60	9.12	9.00	9.06	7.24	5.20	1.04	8.20	8.63	9.25	9.34	7.89	7.89	1.57	8.72
CHICAIZA LOPEZ EVELYN DANIELA	10.00	9.75	9.97	7.89	8.20	1.64	8.50	10.00	9.88	9.94	7.95	8.56	1.91	9.86		9.63	10.00	9.75	9.97	7.89	8.20	1.64	8.50	10.00	9.38	9.69	7.75	9.56	1.91	9.65
CHILQUINGA GUEYARA DAYANA LIZBETH	10.00	10.00	10.00	8.00	9.73	1.94	9.94	10.00	10.00	10.00	8.00	9.94	1.98	9.98		9.96	10.00	10.00	10.00	8.00	9.73	1.94	9.94	10.00	10.00	10.00	8.00	9.94	1.98	9.98
CORDERO TELLO MARIA VICTORIA	10.00	9.75	9.87	7.89	9.93	1.98	9.87	10.00	10.00	10.00	8.00	10.00	2.00	10.00		9.93	10.00	9.75	9.87	7.89	9.93	1.98	9.87	10.00	10.00	10.00	8.00	10.00	2.00	10.00
CHOLLO CHICAIZA ANDRES PAUL	7.13	8.00	7.56	6.04	5.00	1.00	7.04	8.75	9.38	9.06	7.24	7.89	1.67	8.81		7.92	7.13	8.00	7.56	6.04	5.00	1.00	7.04	8.50	9.13	8.81	7.04	7.89	1.67	8.61
CUÑALAYA YOPAXI JOSELYN ELIZABETH DUTAN ARGILES JOSE FILADELFO	10.00	10.00	10.00	8.00	6.60	1.32	9.32	10.00	9.00	9.50	7.60	9.16	1.83	9.43		9.37	10.00	10.00	10.00	8.00	6.60	1.32	9.32	10.00	8.75	9.37	7.49	9.16	1.83	9.32
ESPIÑOZA FREIRE ESTEBAN ROBERTO	9.13	10.00	9.56	7.64	5.97	1.19	8.83	9.50	8.75	9.12	7.29	8.63	1.72	9.01		8.92	9.13	10.00	9.56	7.64	5.97	1.19	8.83	9.50	8.63	9.06	7.24	8.63	1.72	8.96
MAYORGA ZAMBRANO EDDIN EDUARDO	7.25	10.00	8.82	8.89	8.60	1.72	8.61	9.00	9.13	9.06	7.24	9.44	1.89	9.12		8.98	7.25	10.00	8.82	8.89	8.60	1.72	8.61	9.25	8.88	9.06	7.24	9.44	1.89	9.12
MEJIA ABRIL ALAN MATEO	10.00	10.00	10.00	8.00	8.47	1.69	8.83	10.00	9.88	9.94	7.96	9.72	1.94	9.89		9.79	10.00	10.00	10.00	8.00	8.47	1.69	8.83	10.00	9.88	9.94	7.96	9.72	1.94	9.89
MORETA MORETA JONATHAN STALIN	9.25	10.00	9.62	7.89	9.00	1.60	9.23	9.63	9.75	9.19	7.36	7.72	1.54	8.89		9.09	9.25	10.00	9.62	7.89	9.00	1.60	9.23	9.38	9.83	9.00	7.20	7.72	1.54	8.74
MOJEDA ACURIO JENNY ARACELLY	7.13	7.75	7.44	5.95	6.53	1.10	7.05	8.00	7.50	7.75	6.20	7.25	1.45	7.65		7.35	7.13	7.75	7.44	5.95	6.53	1.10	7.05	8.13	7.25	7.69	6.15	7.25	1.45	7.60
PIEDRAHITA ALDAS FRANCISCO JAVIER	10.00	10.00	10.00	8.00	8.83	1.76	9.76	10.00	10.00	10.00	8.00	9.69	1.93	9.93		9.94	10.00	10.00	10.00	8.00	8.83	1.76	9.76	10.00	10.00	10.00	8.00	9.69	1.93	9.93
PINTO BRAVO JESUS EDUARDO	10.00	10.00	10.00	8.00	8.27	1.65	9.65	10.00	10.00	10.00	8.00	9.75	1.95	9.95		9.95	10.00	10.00	10.00	8.00	8.27	1.65	9.65	10.00	10.00	10.00	8.00	9.69	1.93	9.93
PROARO ACURIO JOSUE AARON	7.90	7.75	7.37	5.89	5.67	1.13	7.02	8.25	7.75	8.20	7.25	1.45	7.85		7.33	7.00	7.75	7.37	5.89	5.67	1.13	7.02	8.50	7.00	7.75	8.20	7.25	1.45	7.85	
QUICALQUIN SISA BRIGITTE ALEXANDRA TELLO BALBUENA EDGAR JOSUE	10.00	10.00	10.00	8.00	9.30	1.86	9.86	10.00	10.00	10.00	8.00	9.91	1.98	9.98		9.92	10.00	10.00	10.00	8.00	9.30	1.86	9.86	10.00	10.00	10.00	8.00	9.91	1.98	9.98
VILATUBA POZO JOSE DAVID	7.13	9.00	8.06	6.44	7.40	1.48	7.92	9.50	9.38	9.44	7.56	7.89	1.57	9.12		8.82	7.13	9.00	8.06	6.44	7.40	1.48	7.92	9.63	9.25	9.44	7.56	7.89	1.57	9.12
VILLA AMALUISA ANTHONY ANDRES	7.25	10.00	8.82	8.89	8.97	1.77	8.66	10.00	10.00	10.00	8.00	9.84	1.96	9.96		9.31	7.25	10.00	8.82	8.89	8.97	1.77	8.66	10.00	10.00	10.00	8.00	9.84	1.96	9.96
Totales	10.00	10.00	10.00	8.00	8.53	1.70	9.70	10.00	9.75	9.67	7.89	8.75	1.95	9.94		9.77	10.00	10.00	10.00	8.00	8.53	1.70	9.70	10.00	9.75	9.67	7.89	8.75	1.95	9.94

COMPORTAMIENTO

PROYECTOS ESCOLARES

ANEXO B

B.1. Manual de Usuario

A continuación, se presentará la guía para el manejo de la aplicación. Se debe tomar en cuenta que la aplicación funcionara de igual manera tanto en dispositivos móviles como en el ordenador.

B2. Ingreso a la aplicación

Pantalla de Inicio: Se puede visualizar las noticias o eventos que postee el departamento del distrito.



Al presionar el botón para el ingreso se mostrará la pantalla para llenar los datos del usuario.

- A. Se ingresará el correo del usuario.
- B. Se ingresará la contraseña del usuario.
- C. Botón para validar el usuario e ingresar al sistema.

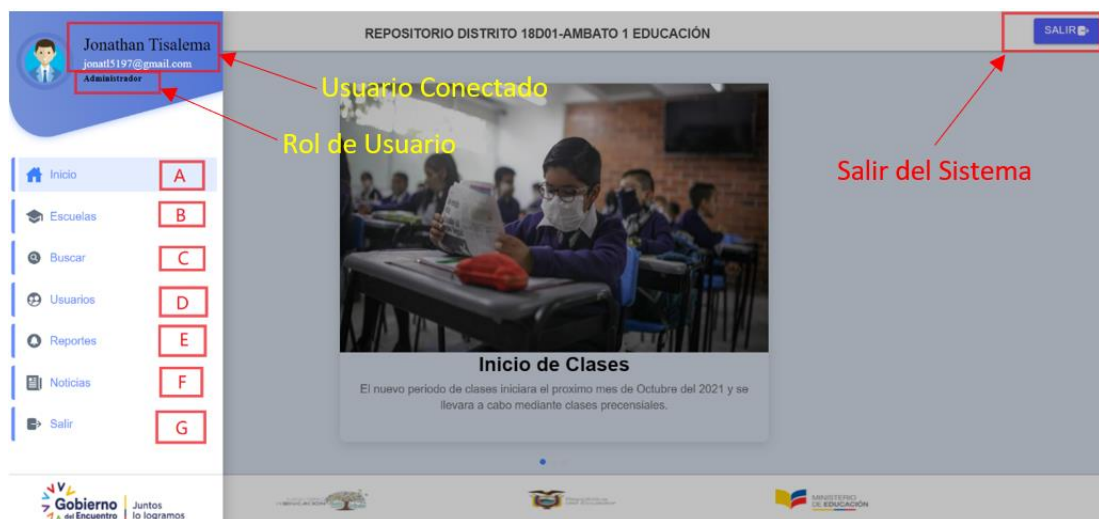
Si los datos de inicio de sesión son incorrectos se mostrará el mensaje de error.



B.3. Menú y pantalla Principal

Si los datos de inicio de sesión son correctos se presentará la pantalla de inicio del sistema y el menú al que podemos acceder según el rol de usuario con el que ingresemos.

Al Ingresar como administrador distrital se presentará las opciones a las que puede acceder en el sistema:



- A. **Botón de inicio:** navega a la pantalla inicial o sección de noticias.
- B. **Escuelas:** se presentará el listado de las unidades educativas registradas.
- C. **Buscar:** ingresara al buscador de archivos.
- D. **Usuarios:** ingresa a la interfaz de gestión de usuarios.
- E. **Reportes:** ingresara a la interfaz para generar los reportes.

- F. **Noticias:** ingresara a la interfaz de gestión de noticias.
- G. **Salir:** se presentará la opción para salir del sistema.

Al Ingresar como administrador institucional y rector se presentará las opciones a las que puede acceder en el sistema:



- A. **Botón de inicio:** navega a la pantalla inicial o sección de noticias.
- B. **Unidad Educativa:** se presentará la pantalla de presentación de la unidad Educativa
- C. **Buscar:** ingresara al buscador de archivos.
- D. **Reportes:** ingresara a la interfaz para generar los reportes.
- E. **Salir:** se presentará la opción para salir del sistema.

B.4. Unidades Educativas

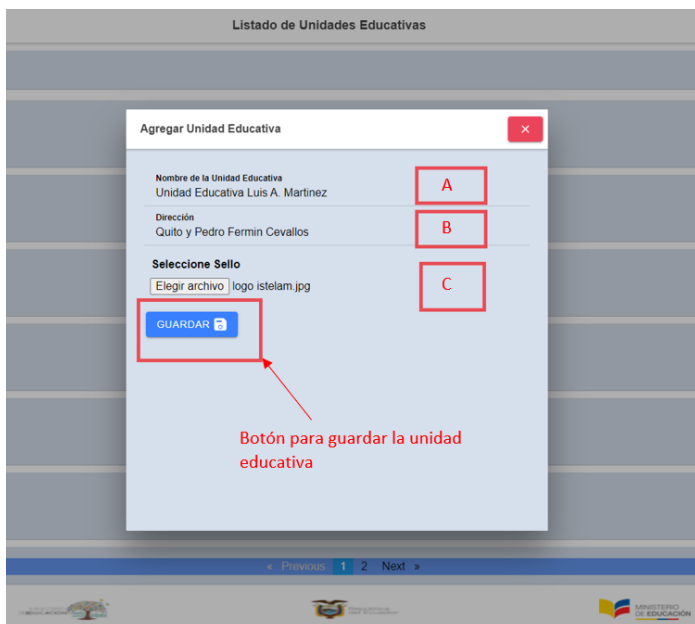
Al ingresar como administrador distrital a la opción de escuelas se presentará la pantalla para la gestión de las unidades educativas.

Nota: Los rectores no tendrán el acceso a las opciones de agregar, actualizar o eliminar información dentro del sistema.



- A. Buscador de unidades educativas.
- B. Listado de unidades educativas
- C. Botón para editar la información de la unidad educativa seleccionada.
- D. Botón para ingresar una nueva unidad educativa.
- E. Botón para eliminar una unidad educativa **que no contenga categorías ni archivos.**

Registrar nueva unidad educativa.



- A. Nombre de la nueva unidad educativa.
- B. Dirección de la nueva unidad educativa.
- C. Logotipo de la nueva unidad educativa.

Actualizar información de una unidad educativa, se presenta el formulario con los datos de la unidad educativa que queremos actualizar.

Actualizar Unidad Educativa

Reemplazar: Unidad Educativa Las Americas
Nuevo Nombre A

Dirección: Ambato Centro
Nueva Dirección B

Seleccione Sello
Elegir archivo No se eligió ningún archivo C

GUARDAR

Botón para actualizar

Previous 1 2 Next

- A. Nuevo nombre para la unidad educativa.
- B. Nueva dirección.
- C. Nuevo logotipo de la unidad educativa.

Para Eliminar una unidad educativa se presionará el botón para eliminar, se debe tomar en cuenta que para poder eliminar una unidad educativa primero hay que eliminar todos los periodos lectivos que contenga.

Listado de Unidades Educativas

Buscar Unidad Educativa.

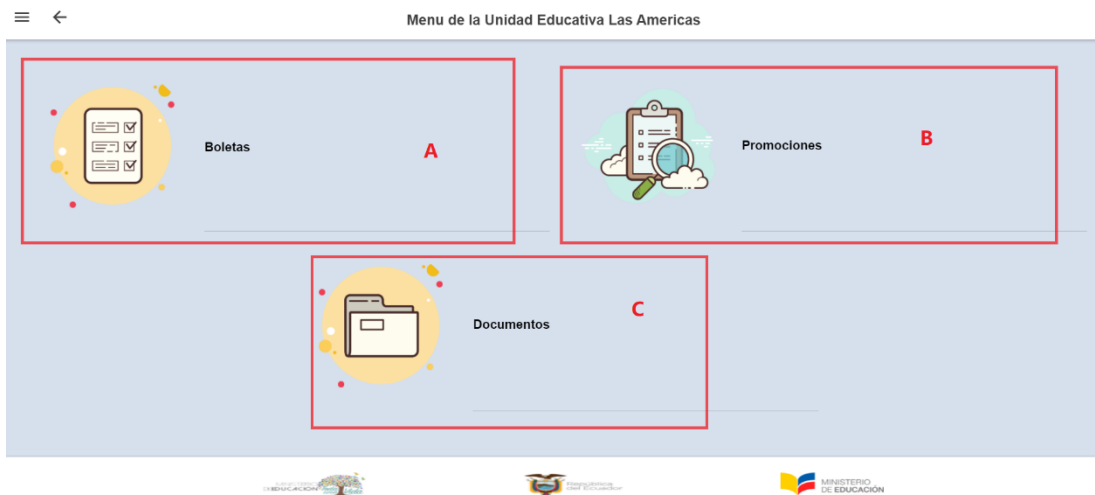
	Unidad Educativa Las Americas Direccion: Ambato Centro		
	Unidad Educativa Guayaquil Direccion: Av. Bolivariana		
	Unidad Educativa Bolivar Direccion: Ambato Centro		
	Unidad Educativa Santo Domingo Direccion: Ambato-Ficoa		
	Unidad Educativa Pichincha Direccion: Av. Bolivariana y Unidad Nacional		
	Unidad Educativa La Inmaculada Direccion: Ambato- Ficoa		

Esta seguro de ELIMINAR la Unidad Educativa: Unidad Educativa Guayaquil

OK CANCELAR

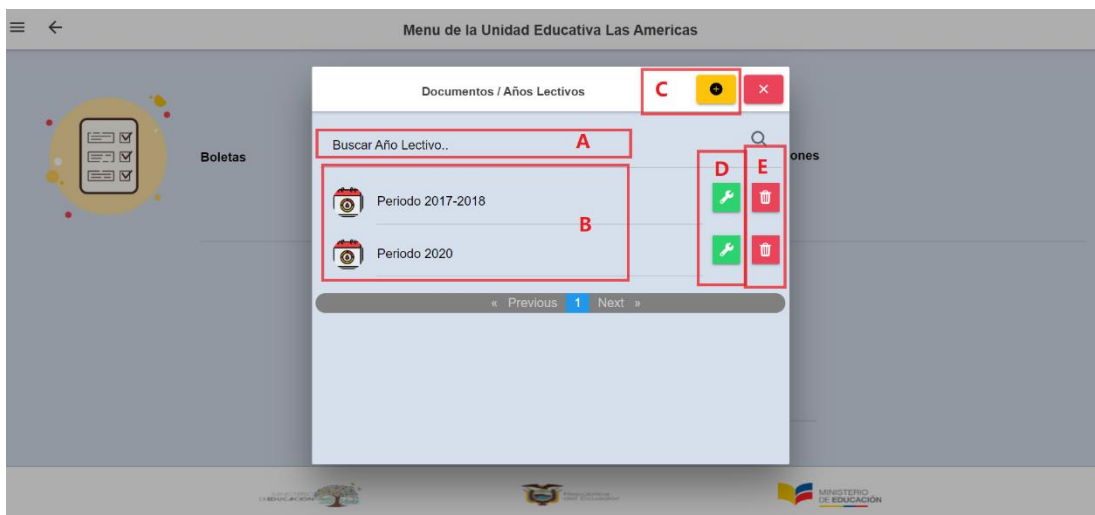
Previous 1 2 Next

Al seleccionar una unidad educativa, se presentará el menú para la documentación de la unidad educativa. Boletas, Concentrado y Documentación Institucional.



- A. Opción para acceder a las boletas de los estudiantes
- B. Opción para acceder a las promociones de los estudiantes.
- C. Opción para acceder a los documentos institucionales.

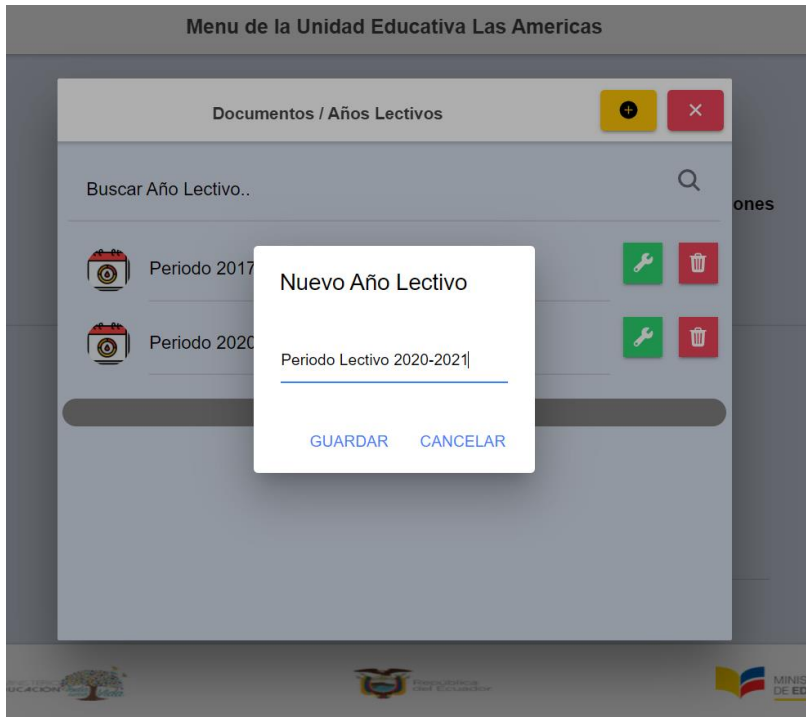
Al seleccionar cualquiera de las tres opciones se presentará el menú para seleccionar el periodo lectivo.



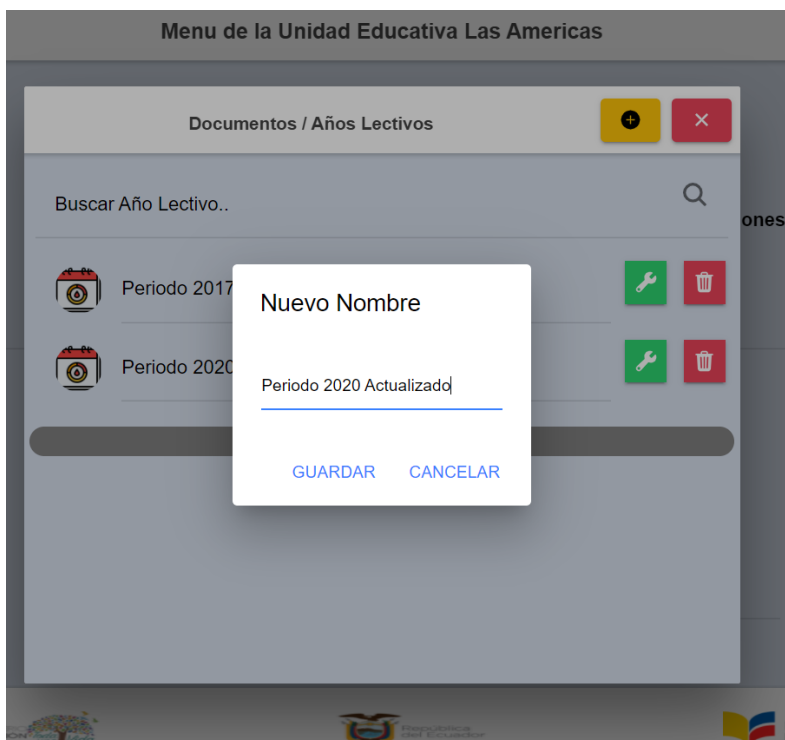
- A. Buscador de periodos lectivos.
- B. Listado de periodos lectivos de la unidad educativa.
- C. Botón para agregar nuevos periodos lectivos.
- D. Botón para editar la información de un periodo lectivo.

- E. Botón para eliminar un periodo lectivo, se debe tomar en cuenta que no debe tener categorías de documentación o niveles para poder eliminar.

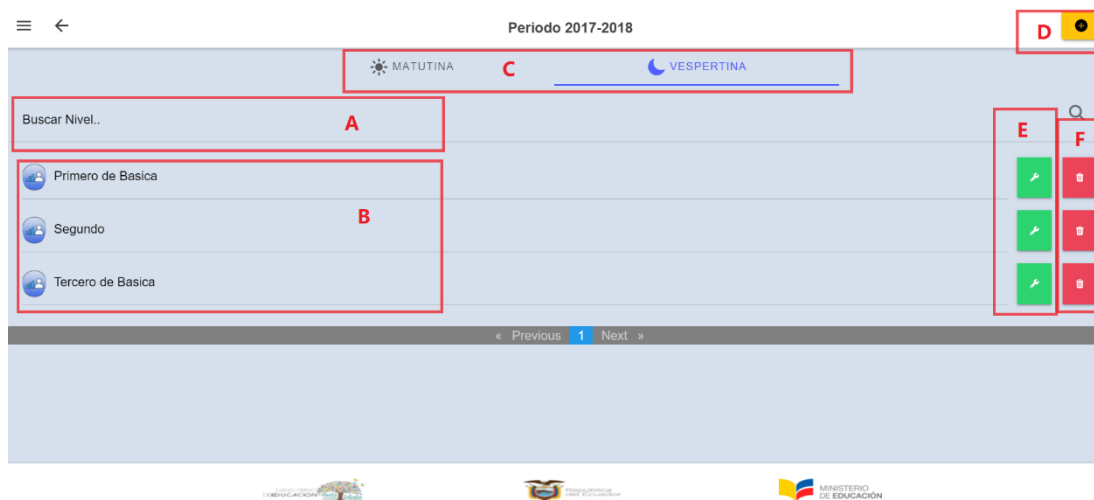
Registrar un nuevo periodo lectivo.



Actualizar el nombre de un periodo lectivo.

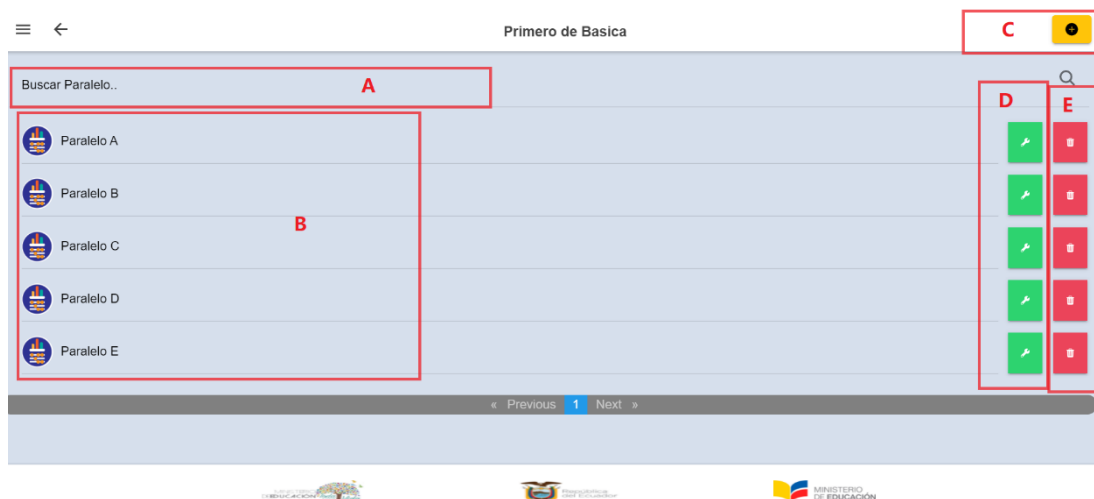


Una vez se ha seleccionado el periodo lectivo y se desea buscar los archivos de los estudiantes bajo la opción Boletas o Promociones del menú principal, se mostrará la pantalla para los niveles dentro del periodo lectivo.



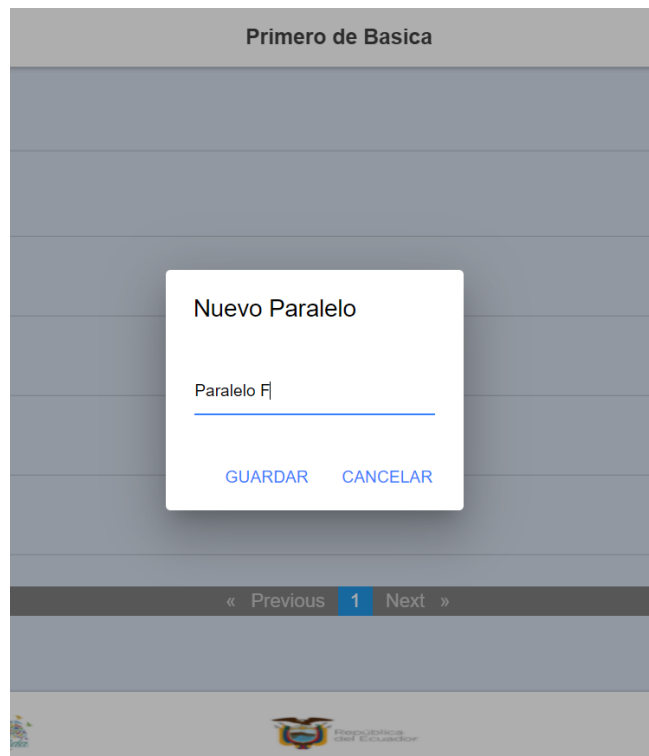
- A. Buscador de niveles.
- B. Listado de niveles según la jornada seleccionada.
- C. Jornada a elegir para el filtrado de niveles.
- D. Botón para agregar un nuevo nivel en la jornada que se haya seleccionado.
- E. Botón para editar el nombre de un nivel.
- F. Botón para eliminar un nivel, se debe tomar en cuenta que el nivel no debe tener paralelos para poder ser eliminado.

Al seleccionar un nivel, se mostrará la pantalla con los paralelos.

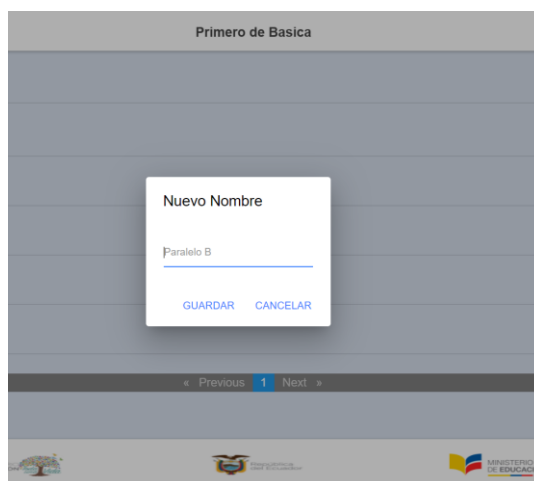


- A. Buscador de paralelos.
- B. Listado de paralelos registrados.
- C. Botón para agregar un nuevo paralelo.
- D. Botón para editar el nombre del paralelo.
- E. Botón para eliminar un paralelo, se debe tomar en cuenta que el paralelo no debe tener archivos de estudiantes para poder ser eliminado.

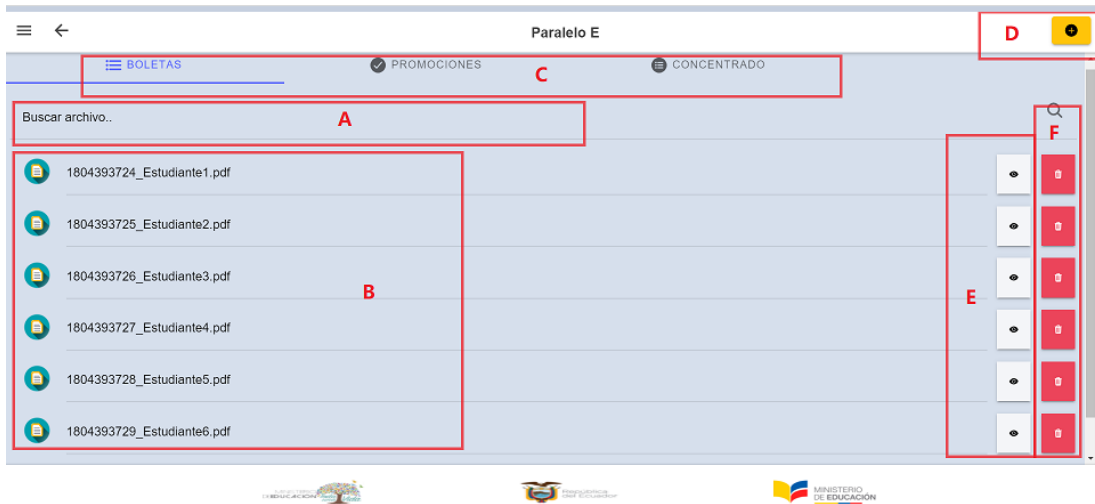
Agregar nuevo Paralelo.



Actualizar el nombre del paralelo tras seleccionar la opción.

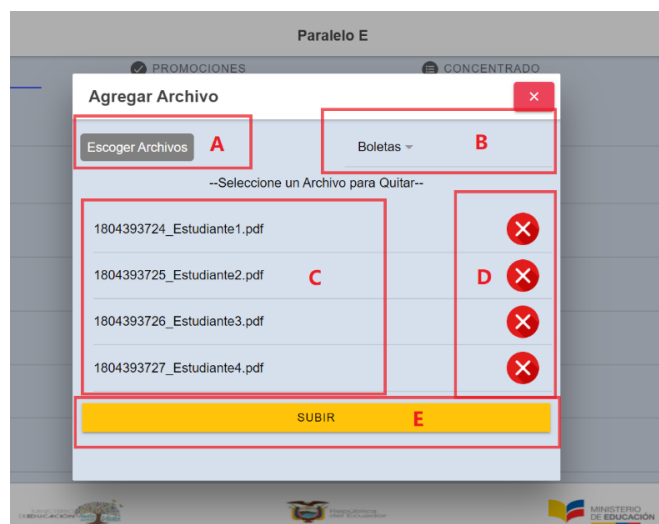


Tras seleccionar un paralelo, se mostrará la pantalla con los archivos de los estudiantes correspondientes a boletas, promociones y concentrado.



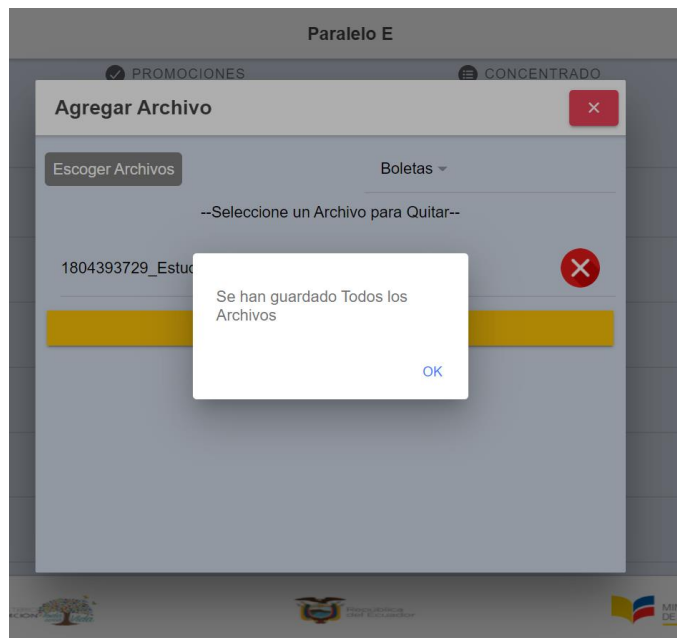
- A. Buscador de Archivos del paralelo
- B. Listado de archivos según el filtro de Boletas, Promociones, Concentrado. Al seleccionar uno se presenta la opción para poder descargar.
- C. Filtros de Archivos.
- D. Botón para el ingreso de archivos.
- E. Botón para visualizar los archivos.
- F. Botón para eliminar archivos.

Agregar Archivos, se puede agregar 1 archivo o varios archivos a la vez como se muestra a continuación.



- A. Botón para seleccionar los archivos para agregar al contenido del paralelo.
- B. Botón para seleccionar a que categoría agregar los archivos Boletas, Promociones, Concentrado.
- C. Listado de Archivos que se vana agregar al contenido del paralelo.
- D. Botones para quitar los archivos que no se vayan a agregar.
- E. Botón para subir los archivos al contenido del paralelo.

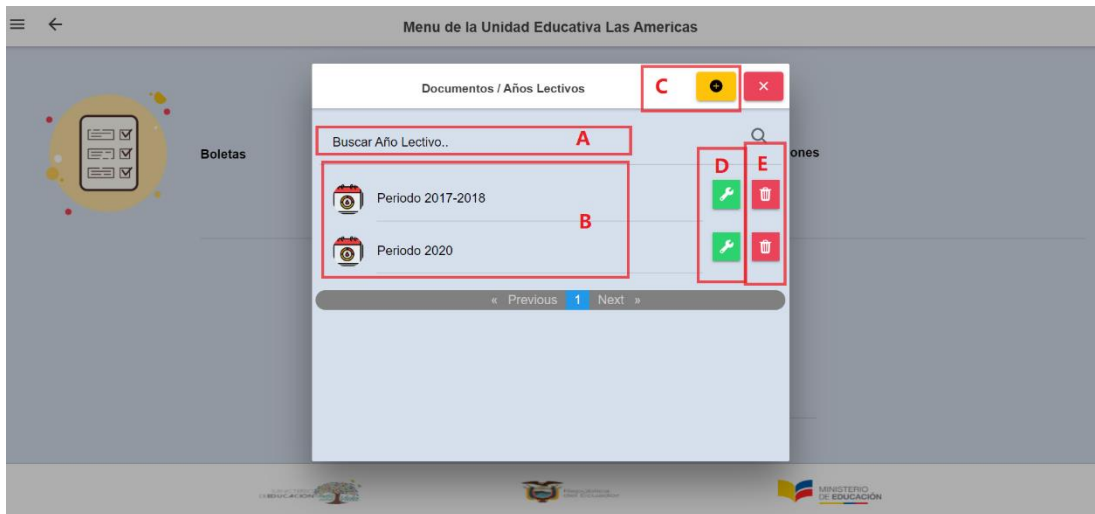
Si los archivos se han agregado correctamente se mostrará el siguiente mensaje.



Para evitar errores e inconsistencias en la documentación, no se puede agregar archivos con el mismo nombre y formato, se presentará el siguiente mensaje.

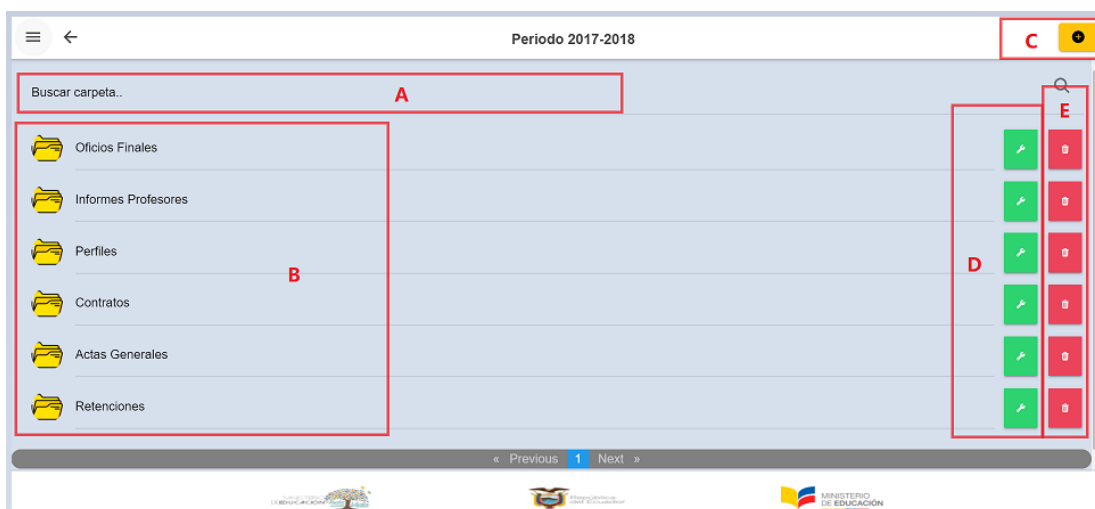


Si se opta por la opción de Documentos Institucionales en el Menú Principal de la documentación de la unidad educativa, se presentará de igual manera la pantalla para seleccionar el periodo lectivo.



- A. Buscador de periodos lectivos.
- B. Listado de periodos lectivos de la unidad educativa.
- C. Botón para agregar nuevos periodos lectivos.
- D. Botón para editar la información de un periodo lectivo.
- E. Botón para eliminar un periodo lectivo, se debe tomar en cuenta que no debe tener categorías de documentación o niveles para poder eliminar.

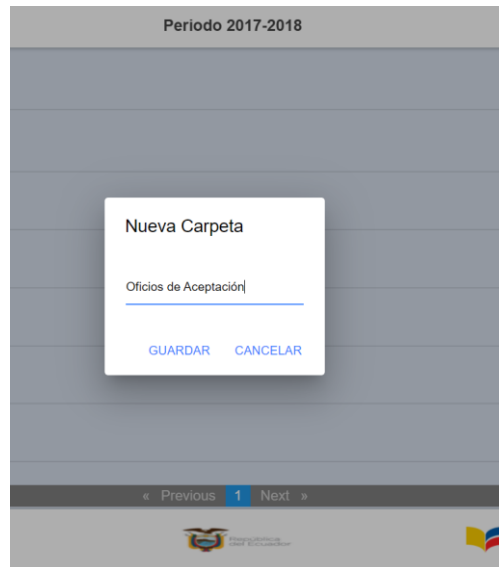
Tras seleccionar un periodo lectivo, se mostrará la pantalla de categorías o carpetas para la documentación institucional.



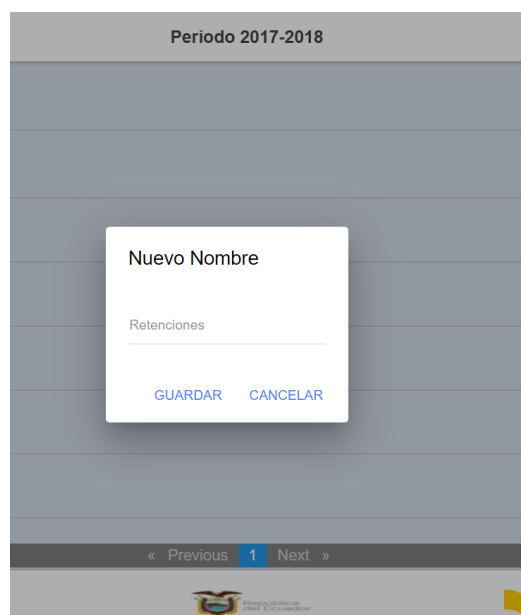
- A. Buscador de carpetas o categorías.

- B. Listado de carpetas o categorías.
- C. botón para agregar nuevas categorías o carpetas.
- D. botón para editar o actualizar el nombre de la categoría.
- E. botón para eliminar la categoría, se debe tomar en cuenta que no debe tener archivos para poder eliminar la categoría.

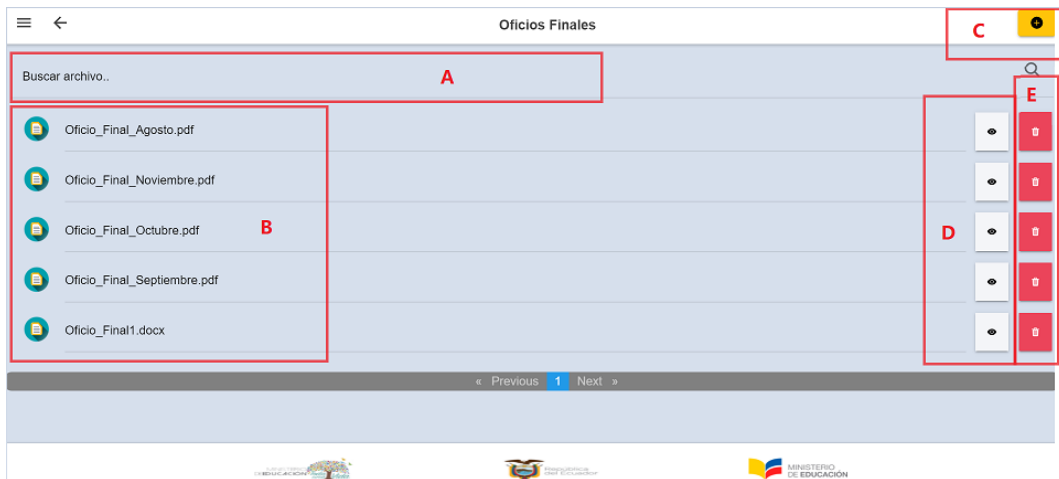
Agregar una carpeta o categoría.



Actualizar el nombre de una carpeta o categoría.

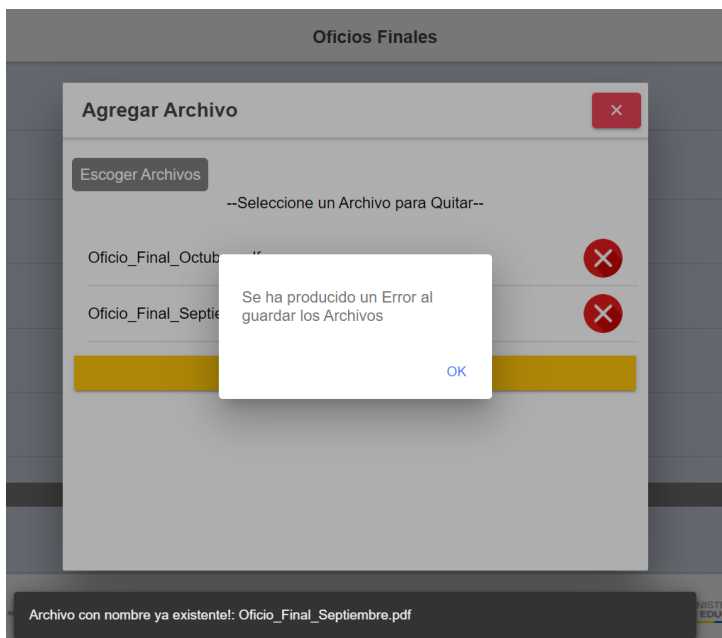


Tras seleccionar una carpeta o categoría se presentará la pantalla con los archivos que contenga.



- A. Buscador de Archivos.
- B. Listado de archivos en la categoría. Al seleccionar uno se presenta la opción para poder descargar.
- C. Botón para agregar archivos al contenido de la categoría.
- D. Botón para poder visualizar el archivo seleccionado.
- E. Botón para eliminar el archivo.

Para evitar errores e inconsistencias en la documentación, no se puede agregar archivos con el mismo nombre y formato, se presentará el siguiente mensaje.



B.5. Buscar Archivos

Para facilitar y acelerar un poco la búsqueda de un archivo de un estudiante en específico, se implementó el buscador de archivos. Para acceder a el solo debemos seleccionar la opción buscar en el menú de navegación.



Al seleccionar la opción se mostrará la pantalla de búsqueda con las opciones para el filtrado de información.

Buscar Documentos

	Documento a Elegir:	Boleta de Calificaciones	A
Institución:		Unidad Educativa Las Americas	B
Año Lectivo:		Periodo 2017-2018	C
Jornada:		Vespertina	D
Grado/Nivel:		Primero de Basica	E
Paralelo:		Paralelo E	F
Estudiante:		1804393724_Estudiante1.pdf	G
		VISUALIZAR  DESCARGAR 	H



- A. El tipo de documento a elegir puede seleccionar entre Boletas de Calificaciones, Promociones, Concentrado y Documentos Institucionales. Tomando en cuenta que para las 3 primeras búsquedas las opciones serán las mismas ya que se filtra hasta llegar a un paralelo en específico para obtener los archivos.
- B. Se selecciona la unidad educativa a la que se quiere acceder. En el caso de administradores institucionales y rectores solo se presentará la unidad educativa a la que pertenezcan.
- C. Se selecciona el periodo lectivo.
- D. Se selecciona la jornada Matutina o Vespertina.
- E. Se selecciona el nivel o grado al que va a acceder.
- F. Se selecciona el paralelo al que va a acceder.
- G. Se selecciona el archivo del estudiante que se desea visualizar o descargar.
- H. Botones para visualizar o descargar el archivo.

Si el tipo de documento a elegir es Documentos Institucionales se mostrarán las siguientes opciones.

Buscar Documentos SALIR

Documento a Elegir:	Documentos Institucionales	A
Institución:	Unidad Educativa Las Americas	B
Año Lectivo:	Periodo 2017-2018	C
Carpeta/Tipo Documento:	Oficios Finales	D
Archivos:	Oficio_Final_Noviembre.pdf	E
VISUALIZAR DESCARGAR F		

- A. El tipo de documento a elegir: Documentos Institucionales.
- B. Se selecciona la unidad educativa a la que se quiere acceder. En el caso de administradores institucionales y rectores solo se presentará la unidad educativa a la que pertenezcan.
- C. Se selecciona el periodo lectivo.
- D. Selecciona la carpeta o la categoría de la documentación.
- E. Se selecciona el archivo que se desee visualizar o descargar.
- F. Botones para visualizar o descargar el archivo.

B.6. Gestión de usuarios

Los administradores distritales podrán acceder a la pantalla para la gestión de usuarios al seleccionar la opción usuarios en el menú de navegación.

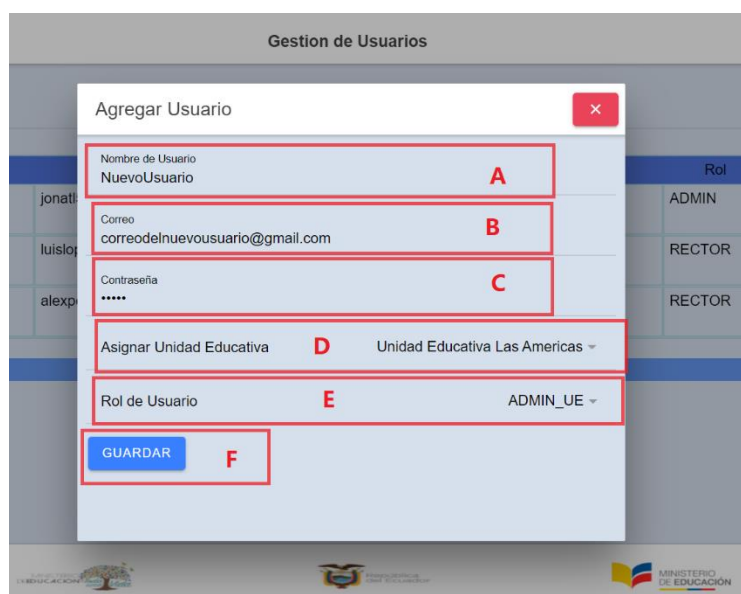


Se presentará la pantalla para la gestión de usuarios.



- A. Buscador de usuarios.
- B. Listado de usuarios registrados
- C. Botón para el ingreso de nuevos usuarios.
- D. Botón para actualizar información de un usuario.
- E. Botón para eliminar un usuario, hay que tomar en cuenta que se puede eliminar usuarios con el rol de ADMIN (administrador distrital) hasta que el sistema quede con un mínimo de 1.

Agregar un nuevo usuario.



- A. Ingresar el nombre del usuario.
- B. Ingresar el correo del usuario.
- C. Ingresar la contraseña del usuario.
- D. Se debe asignar una unidad educativa de la cual será responsable el usuario. Y a la cual podrá acceder y gestionar la información en el caso de ser ADMIN_UE (administrador institucional), si el rol seleccionado es para un administrador distrital la selección de este campo no influirá en la presentación de la información.
- E. Rol de usuario con el que contara el usuario.
- F. botón para guardar la información del nuevo usuario.

Actualizar la información de un usuario. Los datos se cargarán en los campos y solo se actualizará la información de los campos que cambien.

Gestion de Usuarios

Actualizar Usuario
×

Reemplazar: Alex Perez
 A

B

Actualizar Contraseña
 C

Asignar Unidad Educativa D
 ▼

Rol de Usuario E
 ▼

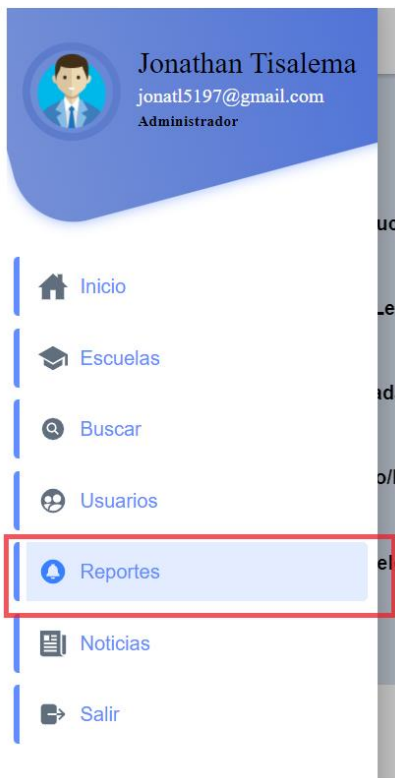
F

- A. Ingresar el nuevo nombre del usuario.
- B. El correo electrónico es único se utiliza como usuario así que no se puede actualizar ni editar.

- C. Ingresar la nueva contraseña.
- D. Asignar la unidad educativa de la que será responsable.
- E. El nuevo rol de usuario que tendrá el usuario.
- F. Botón para actualizar la información del usuario.

B.7. Reportes.

Los usuarios podrán acceder a la pantalla para la emisión de reportes al seleccionar la opción reportes en el menú de navegación.



Se presentará la pantalla para la emisión de reportes.

The image shows a screenshot of the 'Generar Reportes' (Generate Reports) form. The form is titled 'Generar Reportes' and has a 'SALIR' button in the top right corner. The form contains several dropdown menus for selecting report type, institution, year, shift, grade, and parallel. The 'Reporte a Elegir' dropdown is set to 'Estudiantes por Paralelo'. The 'Institución' is 'Unidad Educativa Las Americas', 'Año Lectivo' is 'Periodo 2017-2018', 'Jornada' is 'Vespertina', 'Grado/Nivel' is 'Primero de Basica', and 'Paralelo' is 'Paralelo E'. There are buttons for 'VISUALIZAR', 'DESCARGAR', and 'G'.

- A. El reporte a elegir puede seleccionar el reporte para Estudiantes por Paralelo, Estudiantes por Nivel y Documentos Institucionales.
- B. Se selecciona la unidad educativa a la que se quiere acceder. En el caso de administradores institucionales y rectores solo se presentará la unidad educativa a la que pertenezcan.
- C. Se selecciona el periodo lectivo.
- D. Se selecciona la jornada Matutina o Vespertina.
- E. Se selecciona el nivel o grado al que va a acceder.
- F. Se selecciona el paralelo al que va a acceder.
- G. Botones para visualizar o Descargar el reporte.

Una vez generado el reporte se visualizará de la siguiente manera.


**DIRECCIÓN DISTRITAL 18D01
AMBATO 1 - EDUCACIÓN**

Reporte de Estudiantes por Paralelo

Institución: Unidad Educativa Las Americas
Año Lectivo: Periodo 2017-2018
Nivel: Primero de Basica - Vespertina
Paralelo: Paralelo E
Fecha: lunes 13 de diciembre del 2021

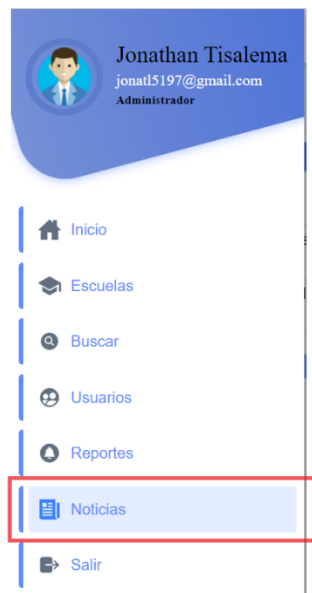
Boletas

1. 1804393724_Estudiante1.pdf
2. 1804393725_Estudiante2.pdf
3. 1804393726_Estudiante3.pdf
4. 1804393727_Estudiante4.pdf
5. 1804393728_Estudiante5.pdf
6. 1804393729_Estudiante6.pdf
7. 1804393729_Estudiante7.pdf

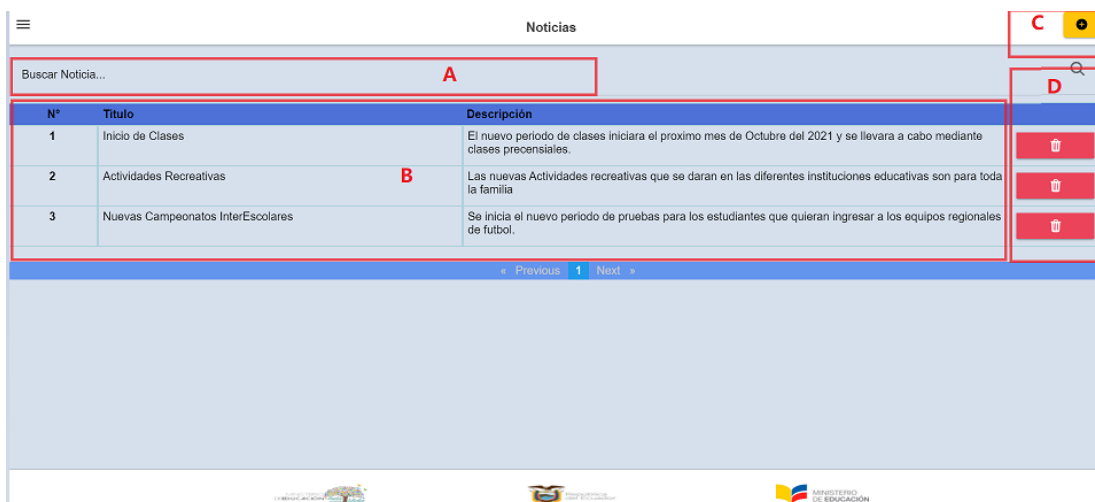
 **Ministerio
de Educación**

B.8. Noticias.

Los usuarios podrán acceder a la pantalla para la gestión de noticias al seleccionar la opción Noticias en el menú de navegación.



Se presentará la pantalla para la gestión de noticias que se visualizará en el inicio de la aplicación.



- A. Buscador de Noticias
- B. Listado de Noticias.
- C. Botón para agregar noticias.
- D. Botón para eliminar noticias.

Agregar Noticias

Noticias

Agregar Noticias [X]

Titulo
Nueva Noticia **A**

Descripción
La descripción para la nueva Noticia **B**

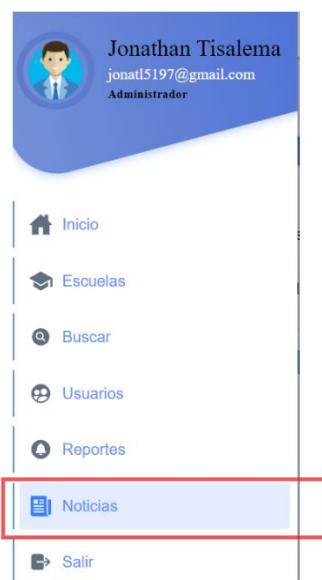
Elegir archivo | images.jfif **C**

GUARDAR **D**

- A. Ingresar el título para la noticia.
- B. Ingresar la descripción para la noticia.
- C. Elegir una imagen para la presentación de la noticia.
- D. Botón para guardar la noticia.

B.9. Salir del Sistema

Los usuarios podrán salir del sistema al seleccionar la opción Salir en el menú de navegación o en el botón en la parte superior derecha.





Inicio de Clases

El nuevo periodo de clases iniciara el proximo mes de Octubre del 2021 y se llevara a cabo mediante clases presenciales.



Al seleccionar la opción se mostrará el mensaje para salir del sistema.

