

Aplicación web progresiva enfocada en romper barreras de comunicación con la población no oyente, mediante la enseñanza del Lenguaje de Señas Colombiano, influenciado por la metodología de inclusión a la inversa.

Enseña2

Dianalin Neme & Iván Romero

Directores de Proyecto.

Cristian Johant Rodríguez Bernal

Yeinner Javier Zambrano Santiago.

Mayo de 2021.

Universidad Antonio Nariño.

Facultad de Ingeniería de Sistemas.

Especialización de Ingeniería de Software.

Copyright © 2021 por Cristian Rodríguez & Yeinner Zambrano. Todos los derechos reservados.

Dedicatoria

Hago dedicatoria del presente documento de proyecto de grado a Dios, en quien he puesto mi esperanza y mi confianza, y quien en su bondad me ha dado vida, salud y ha colocado en mí la inteligencia y el deseo de crecer como profesional. A mi madre Ana Santiago Ovallos y mi padre Javier Zambrano Martínez, quienes nunca han dejado de creer en mí y me han apoyado en todo momento y con esfuerzo y dedicación lograron sacarme adelante y me apoyaron en todo el proceso de mi formación como profesional. A mis hermanas Yeimis, Yeritza y Yenifer que a pesar de todas las discusiones como hermanos siempre me han respaldado y cuidado. A la ingeniera Maribel Romero, quien fue mi directora de proyecto de pregrado y que más que una docente fue una mentora. A mi compañero Cristian Rodríguez Bernal, en quien he visto un espíritu de compañerismo y trabajo en equipo, alta capacidad y gran rendimiento en la labor como ingeniero, un profesional destacable al cual le agradezco por todo el conocimiento compartido y la paciencia que ha tenido conmigo.

Por último, dedico este documento de proyecto a Yesenia Gutiérrez Montero, mi amada esposa, quien, con mucho amor y ternura, me apoyó cada vez que estuve cansado y agotado convirtiéndose en mi mayor soporte, y es por eso por lo que mi mayor deseo es poder seguir adelante y demostrarle cada día mi amor por ella.

Yeinner Javier Zambrano Santiago

Dedicatoria

Dedico este documento principalmente a Dios, por darme las oportunidades para poder estudiar, por permitirme haber llegado a este momento tan importante de mi formación profesional, que a cada instante ha tenido diferentes niveles de dificultad, pero con paciencia hemos superado hasta el momento. A mi madre Myriam Bernal y a mis hermanas Blanca Ramos y Diana Rodríguez, a mi hermano Samuel Bernal por ser los pilares más importantes en mi vida, quienes siempre me han demostrado su apoyo incondicional, y que a pesar de todos los obstáculos que se han presentado nunca han dejado de creer en mí. A mi pareja Luisa Morales quien me impulsó a tomar la decisión para dar este paso profesional y que con su cariño me ha apoyado cada instante en este ciclo. A mis sobrinos Ángel Yépez, Jacobo Rodríguez y Joaquín Yépez, quienes son mi motor para cada día querer crecer profesionalmente y como persona. A Yeinner Zambrano mi compañero de este ciclo profesional, que, a pesar de todas las circunstancias, completamos el desarrollo de este ciclo profesional con mucho esfuerzo y dedicación.

Cristian Johant Rodríguez Bernal

Agradecimientos

Steve Jobs una vez dijo: “Si tu no trabajas por tus sueños, alguien te contratará para que trabaje por los suyos.” Esta frase por más sencilla que pueda parecer es una realidad que no debemos ignorar. Es por eso por lo que como profesionales nos establecimos metas, sacrificamos tiempo y dinero con el fin de alcanzar nuestros sueños, todo esto porque nos determinamos en lograr nuestros objetivos. A pesar de todas las dificultades que vivimos en el proceso de formación estamos agradecidos con todas las personas que directa o indirectamente aportaron en el desarrollo de este proyecto. Hoy estamos muy agradecidos con nuestras madres y padres que fueron el pilar de nuestros valores y nos inculcaron el deseo de estudiar y de salir adelante, gracias a nuestros hermanos y hermanas quienes nos motivaron e impulsaron a seguir luchando, muchas gracias a nuestras parejas que día y noche nos alentaron y nos apoyaron cuando más lo necesitábamos, de igual manera damos gracias a nuestros docentes que con dedicación y compromiso impartieron sus enseñanzas y su experiencia, y por último y no menos importante, damos gracias a nuestra universidad, la Universidad Antonio Nariño que nos dio la oportunidad de ser parte de esta gran familia estudiantil y nos otorgó el privilegio de poder decir que somos NumberUAN.

Abstract

In Colombia there are few inclusion programs for people with hearing problems, in daily activities in the social, educational or work environment, programs which are aimed at the inclusion of this population with disabilities in the environment of hearing people, which generates an overexertion in the execution of these activities to this population. In accordance with the concepts of inclusive education, which establishes that inclusion benefits people with special needs as well as people who do not have it, it is desired to implement a multiplatform application prototype that allows the development of the concept of inclusion to the reverse. Which establishes the inclusion of people without any type of disability to populations with special needs.

This concept will be implemented through the development of a progressive web application oriented to the teaching of sign language, which allows the inclusion of games and recreational activities that stimulate the interaction of these two populations, and the learning of sign language. Colombian or LSC at any time or place through information technologies.

Resumen

En Colombia se presentan pocos programas de inclusión para personas con problemas de audición, en las actividades cotidianas de ámbito social, educativo o laboral, programas los cuales están orientados a la inclusión de esta población con discapacidad al entorno de las personas oyentes, lo cual genera un sobre esfuerzo en la ejecución de dichas actividades a esta población. De acuerdo con los conceptos de la educación inclusiva, la cual establece que la inclusión beneficia a las personas con necesidades especiales como a las personas que no lo tienen, se desea implementar un prototipo de aplicativo multiplataforma que permita el desarrollo del concepto de inclusión a la inversa. El cual establece la inclusión de las personas sin ningún tipo de discapacidad a las poblaciones con necesidades especiales.

Este concepto será ejecutado a través de la elaboración de un prototipo aplicación web progresiva orientada a la enseñanza del lenguaje de señas, la cual permita, la inclusión de juegos y actividades lúdicas que estimulen la interacción de estas dos poblaciones, y el aprendizaje del lenguaje de señas colombiano o LSC en cualquier momento o lugar a través las tecnologías de la información.

Tabla de Contenidos

| | |
|--|----|
| Tabla de Contenidos | 8 |
| Lista de tablas | 10 |
| Lista de figuras | 11 |
| 1. Introducción | 14 |
| 2. Definición del Problema | 16 |
| 2.1. Descripción del Problema | 16 |
| 2.2. Formulación del Problema | 17 |
| 3. Objetivo General | 18 |
| 4. Objetivos Específicos | 18 |
| 5. Marco Referencial | 19 |
| 5.1. Estado del arte | 19 |
| 5.1.1. Investigación. | 19 |
| 5.2. Impacto | 29 |
| 5.3. Componente de Innovación | 30 |
| 5.4. Marco Teórico | 32 |
| 5.4.1. Antecedentes de la Investigación. | 32 |
| 5.4.1.1. El lenguaje de señas. | 32 |
| 5.4.1.2. Educación inclusiva. | 33 |
| 5.4.1.3. Software orientado a la enseñanza. | 35 |
| 5.4.1.4. Aplicación Web Progresiva. | 37 |
| 5.4.1.5. Inclusión a la inversa | 38 |
| 5.4.1.6. Los juegos en la enseñanza. | 39 |
| 5.4.2. Bases o Fundamentos teóricos. | 40 |
| 5.4.2.1. Arquitectura propuesta. | 40 |
| 5.4.2.1.1. Bases de datos: | 40 |
| 5.4.2.1.2. Backend: | 41 |
| 5.4.2.1.3. FrontEnd: | 42 |
| 5.4.2.2. Hardware y Software. | 44 |
| 5.4.2.2.1. Servidor Base de datos: | 45 |
| 5.4.2.2.2. Servidor de Aplicaciones Java: | 45 |
| 5.4.2.2.3. Servidor Web: | 46 |
| 5.4.3. Definición de términos básicos. | 47 |
| 6. Metodología | 50 |
| 7. Proceso de Software | 53 |
| 7.1. Requerimientos Funcionales y No Funcionales | 53 |
| 7.2. Diseño y Arquitectura | 64 |
| 7.2.1. Diagrama de Despliegue | 64 |
| 7.2.2. Casos de Usos Arquitecturalmente Relevantes | 67 |
| 7.2.3. Diagrama de Secuencia | 71 |
| 7.2.4. Diagrama de Clases | 75 |
| 7.2.5. Arquitectura de Alto Nivel | 77 |

| | | |
|------------|---|-----|
| 7.3. | Construcción | 81 |
| 7.3.1. | Construcción componente de Bases de Datos | 82 |
| 7.3.2. | Construcción componente Backend | 87 |
| 7.3.3. | Construcción componente FrontEnd | 91 |
| 7.4. | Pruebas | 97 |
| 7.4.1. | Pruebas Funcionales | 97 |
| 7.4.2. | Pruebas Lighthouse | 106 |
| 7.4.3. | Pruebas Sonarlint | 108 |
| 7.4.4. | Pruebas Sonarcloud | 115 |
| 7.4.5. | Pruebas de Seguridad y Vulnerabilidad | 120 |
| 7.4.5.1. | Análisis de Vulnerabilidades con Nikto. | 121 |
| 7.4.5.2. | Análisis de Vulnerabilidades con Uniscan. | 124 |
| 7.4.5.3. | Análisis de Vulnerabilidades con OWASP ZAP. | 129 |
| 7.5. | Instalación y Configuración | 134 |
| 7.5.5. | Descarga Archivos Fuentes | 134 |
| 7.5.6. | Instalación | 138 |
| 7.5.6.1. | Instalación de la Base de Datos. | 138 |
| 7.5.6.2. | Instalación Backend. | 142 |
| 7.5.6.3. | Instalación FrontEnd. | 145 |
| 8. | Anexos | 149 |
| 8.1. | Modelo Entidad Relación (MER) | 149 |
| 8.2. | Modelo EER | 151 |
| 8.3. | Lista de Chequeo de las Historias de Usuarios | 153 |
| 8.4. | SDL - Aplicación Enseña2 | 153 |
| 8.5. | Pruebas de Seguridad | 153 |
| 9. | Conclusiones | 154 |
| 10. | Referencias | 160 |

Lista de tablas

| | |
|---|-----|
| <i>Tabla 1. Estado del Arte (Fuente: Elaboración Propia)</i> | 23 |
| Tabla 2. Impacto del proyecto (Fuente: Elaboración Propia) | 29 |
| Tabla 3. Características o usos del software educativo (Fuente: Elaboración Propia) | 36 |
| Tabla 4. Historias de usuarios (Fuente: Elaboración Propia) | 54 |
| Tabla 5. Historias de usuarios - Épica Administrador (Fuente: Elaboración Propia)..... | 57 |
| Tabla 6. Historias de usuarios - Épica Aprendizaje (Fuente: Elaboración Propia) | 60 |
| Tabla 7. Pruebas Funcionales (Fuente: Elaboración Propia) | 101 |
| Tabla 8. Análisis AS-IS Sonarlint (Fuente: Elaboración Propia) | 109 |
| Tabla 9. Análisis TO-BE Sonarlint (Fuente: Elaboración Propia) | 112 |

Lista de figuras

| | |
|--|----|
| Figura 1. Logo Insor Educativo (INSOR s. f.)..... | 23 |
| Figura 2. Logo Gesco (Agencia de Noticias UN y Talaga 2019)..... | 23 |
| Figura 3. Logo Hablemos LSC (Es3Colombia 2020)..... | 23 |
| Figura 4. Logo Sep (J. A. 2018)..... | 24 |
| Figura 5. Logo Diccionario de Palabras en Lengua de Señas (Soporte UN CIER 2016)..... | 24 |
| Figura 6. Logo Ceibal LSU. (Ceibal LSU 2017)..... | 24 |
| Figura 7. Logo Centro de Relevo Colombia. (MINTIC y FENASCOL s. f.) | 25 |
| Figura 8. Logo IncluSeñas. (Arenas 2020) | 25 |
| Figura 9. Logo Kitsord. (Kitsord 2020)..... | 25 |
| Figura 10. Logo LSAPP. (LSApp y Posibillian Tech S.A. s. f.) | 26 |
| Figura 11. Logo Dilo en señas (Garza Gutiérrez 2015)..... | 26 |
| Figura 12. Logo Sign Language. (hindi Infoware 2020) | 26 |
| Figura 13. Logo Aprender lengua de señas (NMD STUDIO 2019)..... | 26 |
| Figura 14. Impactos del componente de Innovación (Fuente: Elaboración propia.) | 31 |
| Figura 15. Lenguaje de señas (EcuRed s. f.)..... | 33 |
| Figura 16. Educación Inclusiva Recuperado de (Unesco 2008:1)..... | 34 |
| Figura 17. Logo MySQL (MySQL 2020)..... | 41 |
| Figura 18. Logo Spring Boot (Cleventy 2020) | 42 |
| Figura 19. Logo Angular (Angular.io 2020)..... | 43 |
| Figura 20. Diagrama de arquitectura (Fuente: Elaboración propia.) | 44 |
| Figura 21. Diagrama de Hardware (Fuente: Elaboración Propia) | 46 |
| Figura 22. Diagrama de despliegue (Fuente: Elaboración propia.) | 66 |
| Figura 23. Diagrama de Caso de Uso General (Fuente: Elaboración propia.) | 68 |
| Figura 24. Diagrama de Caso de Uso Admón. (Fuente: Elaboración propia.) | 69 |
| Figura 25. Diagrama de Caso de Uso Usuario (Fuente: Elaboración propia.) | 70 |
| Figura 26. Diagrama de Secuencia Usuarios (Fuente: Elaboración propia.)..... | 73 |
| Figura 27. Diagrama de Secuencia Mod. Admón. (Fuente: Elaboración propia.)..... | 73 |
| Figura 28. Diagrama de Secuencia - Épica Enseñanza (Fuente: Elaboración propia)..... | 74 |
| Figura 29. Diagrama de Clases (Fuente: Elaboración propia.)..... | 76 |
| Figura 30. Diagrama de Arquitectura Alto Nivel (Fuente: Elaboración propia.) | 78 |
| Figura 31. Diagrama de Componentes Alto Nivel (Fuente: Elaboración propia.) | 79 |
| Figura 32. Diagrama de Componentes Nivel 1 (Fuente: Elaboración propia.) | 80 |
| Figura 33. Pull imagen Docker MySQL (Fuente: Elaboración Propia.)..... | 82 |
| Figura 34. Listado de imágenes Docker (Fuente: Elaboración Propia.)..... | 83 |
| Figura 35. Servicios Docker corriendo (Fuente: Elaboración Propia)..... | 84 |
| Figura 36. Prueba de conexión al servicio MySQL (Fuente: Elaboración Propia) | 85 |
| Figura 37. Uso de Workbench 8.0.23 (Fuente: Elaboración Propia)..... | 86 |
| Figura 38. IDE IntelliJ IDEA (Fuente: Elaboración Propia) | 87 |
| Figura 39. Empaquetamiento Enseñados App Backend (Fuente: Elaboración Propia)..... | 89 |
| Figura 40. Versiones de angular y sus librerías (Fuente: Elaboración Propia)..... | 92 |
| Figura 41. Configuración del proyecto Enseña2 FrontEnd (Fuente: Elaboración Propia)..... | 94 |

| | |
|--|-----|
| Figura 42. Paquetes app Enseña2 FrontEnd (Fuente: Elaboración Propia) | 95 |
| Figura 43. Paquetes Enseña2 FrontEnd (Fuente: Elaboración Propia)..... | 96 |
| Figura 44. Logo Postman (Fuente: https://getlogovector.com/postman-inc-logo-vector-svg/).... | 97 |
| Figura 45. Colletions Postman (Fuente: Elaboración Propia) | 98 |
| Figura 46. Consumo del servicio de Registro de Usuario (Fuente: Elaboración Propia) | 99 |
| Figura 47. Consumo del servicio de Registro de Avance de Usuario (Fuente: Elaboración Propia) | 99 |
| Figura 48. Consumo del servicio de Actualización de Estado de Tema (Fuente: Elaboración Propia)..... | 100 |
| Figura 49. E1- HU-002-C1 (Fuente: Elaboración Propia)..... | 101 |
| Figura 50. E1-HU-002-C2 (Fuente:Elaboración Propia)..... | 101 |
| Figura 51. E1-HU-002-C3 (Fuente: Elaboración Propia)..... | 101 |
| Figura 52. HU-002-C4 (Fuente: Elaboración Propia)..... | 102 |
| Figura 53. E1-HU-002-C5 (Fuente: Elaboración Propia)..... | 102 |
| Figura 54. E1-HU-002-C6 (Fuente: Elaboración Propia)..... | 102 |
| Figura 55. E2-HU-002-C1 (Fuente: Elaboración Propia)..... | 103 |
| Figura 56. E2-HU-002-C2 (Fuente: Elaboración Propia)..... | 103 |
| Figura 57. E2-HU-002-C4 (Fuente: Elaboración Propia)..... | 104 |
| Figura 58. E3-HU-001-C1 (Fuente: Elaboración Propia)..... | 104 |
| Figura 59. E2-HU-001-C2 (Fuente:Elaboración Propia)..... | 105 |
| Figura 60. Resultados Análisis Lighthouse (Fuente: Elaboración Propia) | 106 |
| Figura 61. Logo SonarLint (Fuente: https://www.sonarsource.com/products/sonarlint.) | 108 |
| Figura 62. Análisis AS-IS SonarLint (Fuente: Elaboración Propia)..... | 109 |
| Figura 63. Análisis TO-BE Sonarlint (Fuente: Elaboración Propia) | 111 |
| Figura 64. Code Smell Clase AvanceService (Fuente: Elaboración Propia)..... | 113 |
| Figura 65. Código Refactorizado Clase AvanceService (Fuente: Elaboración Propia) | 114 |
| Figura 66. Estadísticas de Líneas de código Sonarcloud (Fuente: Elaboración Propia) | 116 |
| Figura 67. Overview Sonar Cloud primera ejecución (Fuente: Elaboración Propia) | 117 |
| Figura 68. Overview Sonar Cloud ejecución final (Fuente: Elaboración Propia) | 118 |
| Figura 69. Reporte de Análisis de Nikto (Fuente: Elaboración Propia) | 122 |
| Figura 70. Ping realizado por Uniscan (Fuente: Elaboración Propia) | 124 |
| Figura 71. Traceroute realizado por Uniscan (Fuente: Elaboración Propia) | 125 |
| Figura 72. NSLookup realizado por Uniscan (Fuente: Elaboración Propia) | 125 |
| Figura 73. Nmap realizado por Uniscan (Fuente: Elaboración Propia)..... | 126 |
| Figura 74. Análisis de Puertos de Nmap (Fuente: Elaboración Propia) | 127 |
| Figura 75. Análisis de Puertos abiertos de Nmap (Fuente: Elaboración Propia)..... | 128 |
| Figura 76. Alertas de Análisis de OWASP ZAP (Fuente: Elaboración Propia)..... | 129 |
| Figura 77. Muestra de Alerta de Prioridad Media (Fuente: Elaboración Propia)..... | 130 |
| Figura 78. Muestra de Alerta de Prioridad baja (Fuente: Elaboración Propia) | 131 |
| Figura 79. Muestra de Alerta Informativa (Fuente: Elaboración Propia)..... | 132 |
| Figura 80. Ruta de Fuentes del aplicativo Enseña2 (Fuente: Elaboración Propia)..... | 135 |
| Figura 81. Archivo comprimido descargado (Fuente: Elaboración Propia)..... | 136 |
| Figura 82. Archivo Descomprimido (Fuente: Elaboración Propia)..... | 136 |

| | |
|--|-----|
| Figura 83. Carpetas de Fuentes (Fuente: Elaboración Propia) | 136 |
| Figura 84. Archivo war (Fuente: Elaboración Propia)..... | 137 |
| Figura 85. Sentencias SQL (Fuente: Elaboración Propia)..... | 137 |
| Figura 86. Carpeta Proyecto Web (Fuente: Elaboración Propia) | 138 |
| Figura 87. Script de esquema Enseña2db (Fuente: Elaboración Propia)..... | 139 |
| Figura 88. Schema Enseña2db (Fuente: Elaboración Propia) | 139 |
| Figura 89. Script de Data Enseña2db (Fuente: Elaboración Propia) | 140 |
| Figura 90. Consulta tabla Usuario (Fuente: Elaboración Propia)..... | 141 |
| Figura 91. Consulta tabla RolUsuario (Fuente: Elaboración Propia) | 141 |
| Figura 92. Ruta Tomcat (Fuente: Elaboración Propia)..... | 142 |
| Figura 93. Ejecución Tomcat (Fuente: Elaboración Propia) | 143 |
| Figura 94. Backend disponible (Fuente: Elaboración Propia)..... | 143 |
| Figura 95. Consumo del servicio de Autenticación (Fuente: Elaboración Propia)..... | 144 |
| Figura 96. Ruta htdoc XAMPP (Fuente: Elaboración Propia) | 145 |
| Figura 97. Aplicativo Enseña2 en línea (Fuente: Elaboración Propia)..... | 146 |
| Figura 98. Aplicativo Enseña2 Progresivo (Fuente: Elaboración Propia)..... | 147 |
| Figura 99. Acceso Directo Aplicativo Enseña2 (Fuente: Elaboración Propia) | 148 |
| Figura 100. Diagrama MER (Fuente: Elaboración Propia) | 150 |
| Figura 101. Diagrama ERR (Fuente: Elaboración Propia) | 152 |

1. Introducción

Los avances tecnológicos y el creciente auge del uso de las tecnologías de información y comunicaciones han promovido la inclusión de herramientas tecnológicas en la enseñanza de distintos idiomas para la población en general. Tal ha sido el crecimiento que se han adoptado métodos que facilitan la enseñanza de cualquier lenguaje a través del uso de las tecnologías (Larenas y Bruce 2011). Por otra parte, según (Díaz 2017), se reconoce la importancia de involucrar las actividades lúdicas en los sistemas educativos, específicamente en América Latina, por los aportes significativos que estos implementan ya que estas son un instrumento de conocimiento, un factor de socialización, regulador y compensador de la efectividad, instrumento del desarrollo de las estructuras del pensamiento en el niño y medio indispensable de organización, desarrollo y afirmación del infante.

Cabe destacar que las actividades lúdicas tienen como objetivo principal incitar el actuar y pensar rápido de los participantes puesto que, al incentivar armonía en un ambiente de aprendizaje mediante el uso de actividades divertidas y amenas, se promueven la apropiación de los temas en los cuales se enfatiza dichas actividades.

La enseñanza de la lengua de señas no se limita solo para la población no oyente dado que esta tiene alcance a cualquier persona que desea comunicarse por señas. Este tipo de lenguaje basado en movimientos de manos y gestos de cara puede llegar a ser útil en distintas actividades, por lo tanto, la enseñanza de este debe ser inculcada en la población oyente, promoviendo así la inclusión a la inversa hacia la población con discapacidad auditiva. En ese orden de ideas, el uso de actividades y juegos lúdicos para la enseñanza del lenguaje de señas facilitará su aprendizaje

en la población oyente y no oyente, impulsado a través del uso de las TIC, para incrementar la efectividad en la adquisición del conocimiento y con ello la sana convivencia, integración y adaptación entre ambas partes.

Este proyecto tiene como propósito ofrecer una herramienta digital y tecnológica que permita acercar a la población oyente hacia el aprendizaje del lenguaje de señas colombiano, por medio de una plataforma que pueda contener distintas actividades y juegos lúdicos que impulsaran el aprendizaje de diversos temas en lengua de señas, captando así la atención no solo de infantes sino de toda la población en general.

2. Definición del Problema

2.1. Descripción del Problema

Actualmente en Colombia son muy pocas las medidas implementadas para la inclusión de personas no oyentes a las actividades cotidianas, en el ámbito laboral, educativo y social, debido a la mala comunicación entre las personas oyentes y las personas con discapacidades auditivas, ya sea por el desconocimiento del lenguaje de señas y/o por el desinterés de iniciar una comunicación, esta población debe realizar un proceso de inclusión a la inversa lo cual ha causado limitantes para su desarrollo en la comunidad.

Ante esta realidad se ideó una práctica nombrada como inclusión a la inversa dentro de los referentes de la educación inclusiva, la cual trata de incluir las personas que no presentan ninguna discapacidad a las poblaciones con discapacidades como la sordera, con el objetivo de incentivar al aprendizaje del lenguaje de señas para permitir una inclusión social total y no a una inclusión selectiva, refiriendo a que este grupo de personas no deberán sobre esforzarse para encajar en la sociedad, si no la sociedad ya está preparada para convivir con ellas.

Sin embargo, esta práctica no presenta mucho incentivo por parte de la población oyente, ya sea por falta de interés o desconocimiento de esta práctica, partiendo del problema planteado, surge la necesidad de resolver el siguiente interrogante.

2.2. Formulación del Problema

¿De qué manera la implementación de un prototipo de aplicación web progresiva que a través de juegos y actividades lúdicas en lenguaje de señas pueda incentivar la práctica de la inclusión a la inversa en la población oyente de todas las edades?

3. Objetivo General

Incentivar la práctica denominada inclusión a la inversa a través de la elaboración de un prototipo de aplicación web progresiva lúdica que, a través de juegos y actividades en lenguaje de señas, permitan aprender este lenguaje de manera recreativa, apoyando esto a mejorar los niveles de convivencia, a fin de facilitar la socialización con la población no oyente.

4. Objetivos Específicos

- Diseñar y desarrollar un prototipo de aplicación web progresiva de apoyo para fomentar la inclusión social a la inversa a través de actividades lúdicas en lenguaje de señas.
- Identificar los elementos de las prácticas ágiles que puedan dar soporte a la ejecución del proyecto, y elaboración de la aplicación de inclusión social.
- Identificar la correcta arquitectura que permita un óptimo desarrollo, para la funcionalidad esperada de la aplicación Enseña2.
- Crear una estructura de aplicación de alto nivel, empleando el paradigma de programación orientado a objetos, que implemente en su diseño los principios para la alta cohesión, bajo acoplamiento, encapsulamiento, polimorfismo, abstracción y herencia.
- Implementar un componente de seguridad que permita el control de acceso a las operaciones del Backend, a través del uso de un token, y así mismo que también permita el cifrado y validación de la contraseña registrada por el usuario.

- Diseñar una interfaz gráfica adaptable para los dispositivos móviles más populares.

5. Marco Referencial

5.1. Estado del arte

El presente apartado describe los estudios previos en el área de las aplicaciones de software y uso de las tecnologías de la información para impulsar el aprendizaje del lenguaje de señas. Como primer tema se identificarán algunas páginas webs que apoyan a la comunidad no oyente y refuerzan el aprendizaje de esta lengua en Colombia, compuesta por un total de 3 sitios identificados. Como segundo tema se identificarán algunas plataformas orientadas al apoyo del aprendizaje de la lengua de señas, a través del uso de aplicativos orientados a dispositivos móviles o multi plataforma.

5.1.1. Investigación.

El lenguaje de señas o de signos (LS) “es la lengua natural de expresión y configuración gesto-espacial y percepción visual gracias a la cual las personas con discapacidad auditiva pueden comunicarse con su entorno social” (Vercher 2018). Según Vercher (2018), este tipo de lenguaje está basado en los movimientos y las expresiones realizadas con las manos, los ojos, el rostro, la boca y el cuerpo. Generalmente, las personas con discapacidad auditiva son los principales usuarios de este tipo de lenguaje, pero esta puede ser utilizada por la comunidad oyente; de hecho, su uso no se limita a una conversación, sino que puede ser implementada para distintas actividades.

La lengua de señas consta de un sistema de reglas gramaticales y pragmáticas cómo cualquier otro idioma, actuando de manera similar a otras lenguas, este lenguaje de señas no es universal, dado que esta varía de acuerdo con las comunidades de personas sordas que se constituyan en el mundo y su idioma nativo.

A lo largo de la historia humana, el lenguaje de señas ha sido empleado como medio de comunicación por diversas comunidades de oyentes y no oyentes. Existen registros históricos que relatan cómo distintas tribus de indígenas usaban una lengua de señas como medio de comunicación con otras tribus que hablaban un idioma distinto, incluso, este sistema se mantuvo aún mucho después de la conquista europea.

Según (Vercher 2018) en la Edad Media, los monjes usaban la lengua de señas principalmente en las abadías para comunicarse entre ellos ya que no podían comunicarse de manera oral. En el siglo XVI, Pedro Ponce de León, monje benedictino español, fue considerado como el primer maestro de sordos y utilizó un alfabeto manual basado en las lenguas de señas monásticas utilizadas por los monjes que hicieron un voto de silencio.

En 1620, Juan de Pablo Bonet recopiló e ilustró el alfabeto manual español en su obra “Reducción de las letras y arte para enseñar a hablar a los mudos”, considerada como el primer tratado moderno de fonética en la lengua de señas que instauró una metodología de enseñanza oral para los sordos (Rodríguez y G 2000). Dicha obra en el siglo XVIII sirvió de base al alfabeto de signos inventado por el francés Abad de L’Epee quien fue el creador de la primera escuela pública para sordos.

En Colombia actualmente, la lengua de señas colombiana o LSC es la lengua usada por la comunidad sorda colombiana y fue reconocida oficialmente como lengua en el año 1996 a través de la Ley 324 durante el gobierno del Dr. Ernesto Samper Pizano, en donde se crean algunas normas a favor de la población no oyente, mencionando lo siguiente: "El Estado Colombiano reconoce la Lengua Manual Colombiano, como idioma propio de la Comunidad Sorda del País" (art. 2) (SAMPER PIZANO, OCAMPO, y OLGA 1996). Según Rozo, esta lengua se caracteriza por ser visual y corporal, es decir que la comunicación se establece con el cuerpo en un espacio determinado.

La comunidad sorda colombiana, desde 1984 tomó la iniciativa por el estudio, divulgación y enseñanza de la lengua de señas, al igual que de la situación del sordo en el medio. Con el tiempo se estableció un grupo de investigación que trabajó fuertemente en el lenguaje manual colombiano, denominación que privilegiaba el papel de las manos en la comunicación desde la perspectiva de la comunidad sorda colombiana. En 1993, la Federación Nacional de Sordos de Colombia (Fenascol) publicó las cartillas del Lenguaje Manual Colombiano, como resultado de las primeras aproximaciones y de la labor de enseñanza de la lengua (Rozo Melo s. f.).

El interés por el estudio de la lengua de señas fue ampliándose hacia otras instancias, como las universidades, que se vincularon con gran interés a este propósito. La aptitud tomada en las investigaciones y el trabajo en equipo con otras organizaciones e instituciones reforzaron el estudio de la lengua de señas y la percepción sobre la misma. Así en el siguiente volumen, la cartilla número dos de Fenascol, se denomina Lengua de señas colombiana, con la siguiente aclaración: "De otra parte las investigaciones lingüísticas llevadas a cabo en varios países latinoamericanos

han permitido aclarar que la denominación correcta es SEÑAS, las cuales están conformadas por movimientos faciales, corporales y manuales." (Mejía, 1996.).

Luego, como resultado de un trabajo colaborativo entre el INSOR (Instituto Nacional para Sordos) y el Instituto Caro y Cuervo, con el apoyo de Fenascal, se realiza la publicación del Diccionario Básico de la lengua de señas colombiana.

La era digital ha brindado muchas herramientas que permiten que la enseñanza del lenguaje de señas sea impartida con mayor facilidad y alcance a un mayor público. Cabe mencionar que existen muchos ejemplos tanto a nivel nacional e internacional del trabajo realizado por diferentes organizaciones, empresas e instituciones gubernamentales y sociales en la enseñanza del lenguaje de señas a través del uso de las nuevas tecnologías de información y comunicaciones. A continuación, se presentan la siguiente tabla (Tabla 1) que contiene ejemplos de aplicaciones móviles y plataformas digitales que dan atención a la temática planteada dentro de este documento:

Tabla 1. Estado del Arte (Fuente: Elaboración Propia)

| Aplicativo | Descripción |
|---|--|
|  <p data-bbox="217 554 560 642">Figura 1. Logo Insor Educativo (INSOR s. f.)</p> | <p data-bbox="604 394 1351 489">En la plataforma web del Instituto Nacional para Sordos (INSOR) se encuentran una variedad de recursos educativos accesibles que van dirigidos a la comunidad sorda colombiana.</p> <p data-bbox="604 531 1351 737">Este portal cuenta con una sección denominada Recursos educativos, que contiene diversos módulos específicos catalogados en ciencias, los cuales se componen de clases y videos cortos de diferentes áreas del saber. También posee una sección Diccionario LSC, en donde se puede encontrar las definiciones en español y en Lengua de Señas Colombiana (LSC) de múltiples términos cotidianos.</p> |
|  <p data-bbox="201 1140 574 1234">Figura 2. Logo Gesco (Agencia de Noticias UN y Talaga 2019)</p> | <p data-bbox="604 814 1351 1056">El aplicativo Gesco es una plataforma concepto digital que permite difundir contenidos a través de cualquier plataforma, según (Agencia de Noticias UN y Talaga 2019) apoya la práctica y el aprendizaje de la Lengua de Señas Colombiana (LSC) entre las personas oyentes que tienen dificultad para acceder a cursos en la modalidad presencial, y desde luego entre la comunidad con discapacidad auditiva en general.</p> <p data-bbox="604 1098 1351 1371">La plataforma fue diseñada para promover una interacción social dinámica entre personas oyentes y no oyentes a través contenidos multimedia segmentados con información sencilla y simple. Gesco permite la personalización del aprendizaje según la necesidad del usuario, fraccionando la información para que la educación sea progresiva, es decir, desde conceptos básicos hasta los más avanzados, utilizando como componente principal la gamificación, es decir el uso de juegos para apoyar el tema del aprendizaje.</p> |
|  <p data-bbox="315 1507 461 1535">¡Hablemos LSC!</p> <p data-bbox="224 1560 555 1654">Figura 3. Logo Hablemos LSC (Es3Colombia 2020)</p> | <p data-bbox="604 1472 1351 1640">El aplicativo hablemos LSC según el autor (Es3Colombia 2020) es un diccionario multimedia de Lengua de señas- Español Colombia el cual está diseñado para plataformas Android. Esta aplicación agrupa su contenido en diversos temas prácticos usados con frecuencia en una conversación con este tipo de lenguaje.</p> |



Figura 4. Logo Sep (J. A. 2018)

El aplicativo SEP, es una herramienta para abordar el desarrollo de la relación palabra-concepto orientada al Profesional Sordo en formación, de forma específica en el área de Educación (Inicialmente), esto a partir de expresiones y términos académicos y científicos propios de las disciplinas de Educación Infantil y Educación Especial (J. A. 2018).

Está diseñado para plataformas Android y básicamente contiene un diccionario de términos y conceptos expresados por un modelo señante, de forma escrita y gráfica.



Figura 5. Logo Diccionario de Palabras en Lengua de Señas (Soporte UN CIER 2016)

El aplicativo es una plataforma web contenida en el sitio web del ministerio de educación Colombia aprende, el cual según el autor (Soporte UN CIER 2016) atiende a las particularidades lingüísticas y comunicativas de la población sorda de Colombia, mediante el uso de recursos audiovisuales, significados y ejemplos.

Este contiene palabras básicas del uso cotidiano en forma de diccionario Básico de Lenguaje de Señas Colombiano para usarse como herramienta en orientaciones educativas que promuevan la formación permanente en la comunidad educativa.



Figura 6. Logo Ceibal LSU. (Ceibal LSU 2017)

El aplicativo Ceibal LSU es una aplicación de tipo móvil desarrollada para plataformas Android dirigida a la comunidad oyente uruguaya, este aplicativo de tipo juego el cual ofrece la enseñanza a través de vídeos que demuestran cómo se traducen las palabras al lenguaje de señas uruguaya y propone una serie de pruebas para confirmar que se comprendió la traducción de las palabras de forma correcta.



Figura 7. Logo Centro de Relevo Colombia. (MINTIC y FENASCOL s. f.)

El aplicativo Centro de Relevo es un sitio web promovido por el Ministerio de Tecnologías de la Información y las Comunicaciones-MINTIC y la Federación Nacional de Sordos de Colombia, Fenascal, el cual busca favorecer a la población no oyente del país, en sus necesidades comunicativas básicas, a través de las TIC.

Según el autor (MINTIC y FENASCOL s. f.) el Centro de Relevo es un medio por el cual las personas no oyentes pueden comunicarse con cualquier persona oyente en todo el país, solicitar el servicio de interpretación cuando requieran ser atendidos en las diferentes instituciones o entidades del país, acceder al conocimiento y uso de las TIC, siendo no sólo consumidores sino productores de información, y consta de intérpretes que refuerzan conocimientos en Lengua de Señas Colombiana constantemente a través de diversos cursos de formación.



Figura 8. Logo IncluSeñas. (Arenas 2020)

IncluSeñas es una aplicación móvil diseñada para plataforma Android, gratuita y sin fines de lucro que facilita el aprendizaje básico en Lengua de Señas Chilena (LSCH), Lengua de Señas Colombiana (LSC) y Lengua de Signos Española (LSE) de manera sencilla y lúdica, buscando así ayudar a la inclusión de las personas no oyentes en la sociedad. Su funcionalidad consta de un diccionario de palabras que posee cortos videos explicativos de la traducción en el lenguaje de señas.



Figura 9. Logo Kitsord. (Kitsord 2020)

El aplicativo Kitsord es una aplicación gratuita para aprender lenguaje de señas, desarrollada con fines no lucrativos, que tiene como objetivo o finalidad, colaborar en romper la barrera social y de comunicación existente entre las personas con discapacidad auditiva y las que no la padecen.

Está desarrollada para plataformas Android y contiene material didáctico y multimedia, tales como, fotografías y videos, que apoyan el aprendizaje del lenguaje de señas ASL, LSE y LENSEGUA.



Figura 10. Logo LSAPP. (LSApp y Posibillian Tech S.A. s. f.)

El aplicativo LSApp es una aplicación gratuita, disponible en appstore y Google store, desarrollada para promover el aprendizaje de la Lengua de Señas Argentina (LSA).

Según el autor (LSApp y Posibillian Tech S.A. s. f.) entre sus funcionalidades se puede destacar un buscador de señas, que pasa de español a LSA, juegos para facilitar la práctica de LSA y consejos para comunicarse adecuadamente con personas Sordas.



Figura 11. Logo Dilo en señas (Garza Gutiérrez 2015)

Dilo es un aplicativo desarrollado para plataformas Android utilizado para aprender Lengua de Señas Mexicana (LSM). Está dirigido principalmente para niños sordos pequeños, según el autor (Garza Gutiérrez 2015) este aplicativo cuenta con 89 señas en 7 categorías: Familia, Alimentos, Juguetes, Animales, Colores, Números y Abecedario. El juego consiste en relacionar videos con imágenes.



Figura 12. Logo Sign Language. (hindi Infoware 2020)

El aplicativo Sign Lenguaje, es una aplicación desarrollada para la enseñanza del lenguaje de señas americano. Dentro del aplicativo se encuentra organizado en varias secciones de diversos temas que comienzan con conceptos básicos hasta un nivel avanzado, este aplicativo según el autor (hindi Infoware 2020), cuenta con los lenguajes de señas en los idiomas Czech, British, Estonian, French, German, Austrian, Italian, Japanese, Latvian, Lithuanian, Polish, Portuguese, Brazilian, Romanian, Russian, Spanish, Swedish, Turkish.



Figura 13. Logo Aprender lengua de señas (NMD STUDIO 2019)

El aplicativo Aprender lengua de señas, es una aplicación desarrollada para plataformas Android con el objetivo de enseñar el lenguaje de señas español, dentro de este aplicativo según el autor (NMD STUDIO 2019), se encontrará gestos para alfabetos, números y también algunas palabras cotidianas. Esta aplicación también puede servir fácilmente como una guía diaria para el lenguaje de señas y puede consultarla cuando desee guardar las tarjetas sin conexión.

Esta aplicación también puede servir fácilmente una guía diaria para el lenguaje de señas y permite guardar información en el dispositivo móvil para que puedan acceder a este sin necesidad de tener alguna conexión activa.

Una vez explorada toda la información, donde se realizó la búsqueda de la historia del lenguaje de señas y las aplicaciones que implementan este tema, se identificó que el lenguaje de señas ha sido utilizado y promovido a través de la historia para la comunicación no solo de personas que presentan problemas auditivos, sino también para la comunicación en general, puesto que se encuentra en las diferentes etapas del desarrollo humano y en las comunicaciones asertivas, sin importar el idioma o nacionalidad, por ende podemos encontrar que cada nación presenta un modelo adaptativo del lenguaje de señas según su región o lengua, orientado específicamente al uso e inclusión de poblaciones que presentan algunas discapacidades, la cual comúnmente se encuentra arraigada a la sordera.

Con el boom de las nuevas tecnologías de la comunicación, podemos encontrar diferentes aplicaciones y plataformas que tienen como funcionalidad ayudar al aprendizaje de estas lenguas, o la inclusión de esta población no oyente, como pudimos observar anteriormente en aplicaciones como INSOR, Centro de Relevo y diccionario de palabras de señas, las cuales están enfocadas en ayudar a la población no oyente prestando servicios de apoyo y acompañamiento, sin dejar atrás la enseñanza. También pudimos observar que se encuentran un gran número de aplicaciones enfocadas a enseñar el lenguaje de señas a manera de diccionario, aplicaciones citadas anteriormente en este documento tales como SEP, Ceiba Isu, Inluseñas, kidsord, lsapp, dilo en señas, etc. Y de manera más enfocada pudimos encontrar la aplicación Gesco desarrollada por el ingeniero industrial Jorge Steban Talaga, la cual se encuentra enfocada en enseñar el lenguaje de señas mediante el uso de juegos, a manera de método de enseñanza de la gamificación del tema, método utilizado actualmente en plataformas como Duolingo.

En conclusión, la idea planteada en este proyecto presenta un componente innovador, ya que además de utilizar los juegos y las actividades lúdicas como métodos de enseñanza de manera similar a la app Gesco, la idea es utilizar un prototipo aplicación web progresiva multiplataforma, con el fin de albergar actividades y juegos, no solo para enseñar el lenguaje de señas, sino para permitir la interacción de las personas oyentes con la población no oyente fomentando la inclusión a la inversa, a través de la lúdica, tratando de llegar a personas de todas las edades.

5.2. Impacto

Luego de conocer el estado actual del arte en la temática planteada dentro de este proyecto, se realizó un análisis de los impactos que puedan generar la realización y ejecución de este. A continuación, se presenta la siguiente tabla (Tabla 2) en donde se indican los principales factores impactados:

Tabla 2. Impacto del proyecto (Fuente: Elaboración Propia)

| Aspecto | Impacto |
|-------------|---|
| Social | El principal impacto que generará el presente proyecto se verá reflejado en el aspecto social, ya que el objetivo de este está enfocado en brindar una herramienta con propósitos sociales, beneficiando significativamente a la comunidad con discapacidad auditiva, promoviendo con ello la inclusión social e integración de la comunidad oyente con esta misma, e impulsando el aprendizaje del lenguaje de señas colombiano en la cultura actual del país. |
| Educativo | Otro de los grandes aspectos impactados por el presente proyecto es el educativo, dado que una de las principales características de este será la implementación de diversas actividades y juegos didácticos para todas las edades, en especial para niños, estimulando con ello el aprendizaje del lenguaje de señas colombiano desde temprana edad. |
| Tecnológico | El proyecto se plasma en un entorno electrónico y digital que sumado a la situación de aislamiento actual ofrece entretenimiento a la familia en general gracias a un prototipo aplicación web progresiva con actividades y juegos didácticos, promoviendo el aprendizaje grupal y la interacción con las nuevas tecnologías de información y comunicaciones. |

5.3. Componente de Innovación

De acuerdo la investigación de referentes de aplicaciones nacionales e internacionales que definían como su tema principal la implementación, el uso o el aprendizaje del lenguaje de señas tales como Hablemos LSC, SEP, Dilo en Señas y Aprender lengua de Señas (España), se identificó que ofrecen múltiples contenidos multimedia para aprender este lenguaje, por otro lado la aplicación prototipo colombiana Gesco elaborada por el estudiante de diseño industrial de la universidad nacional de Colombia (UNAL) Jorge Steban Talaga, al ver esta problemática se orientó a proponer, “una plataforma que promueve las dinámicas de interacción social, segmenta los contenidos multimedia sin recargarlos de información innecesaria, permite la personalización según la necesidad de aprendizaje del usuario y fracciona la información de manera que la educación sea progresiva, es decir, desde conceptos básicos hasta los más avanzados.” (Agencia de Noticias UN y Talaga 2019).

De acuerdo con el contexto anterior, la propuesta planteada en este proyecto es elaborar un prototipo de aplicación web progresiva multiplataforma, que promueva la inclusión social, mediante el concepto de inclusión a la inversa entre la población oyente, hacia la población no oyente, mediante la interacción a través de la plataforma propuesta que contenga diversas actividades y juegos lúdicos, en base al conocimiento y aprendizaje del lenguaje de señas colombiano o LSC, permitiendo a través del uso de esta plataforma como componente tecnológico, explorar el desarrollo del usuario mediante la sección de Aprendizaje por medio del seguimiento de las actividades educativas como vocabulario, letras y palabras de uso cotidiano como componente educativo, aprendiendo de esto a compartir e incentivar el reconocimiento y apoyo a

poblaciones vulnerables a través del aprendizaje mediante el uso de esta plataforma virtual como componente social.

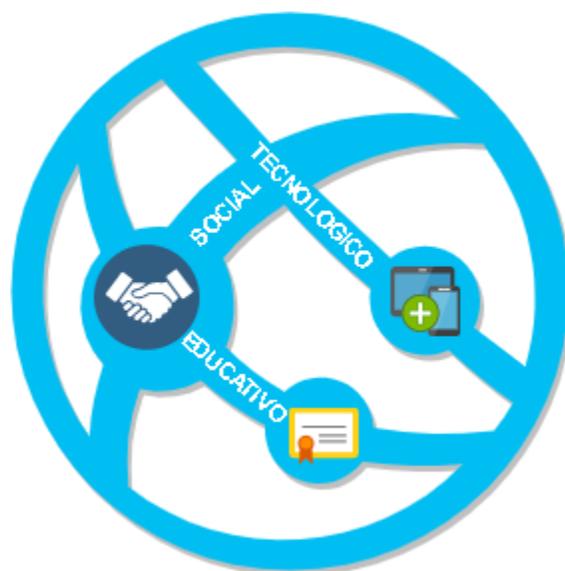


Figura 14. Impactos del componente de Innovación (Fuente: Elaboración propia.)

5.4. Marco Teórico

En este apartado se definirá los conceptos de lenguaje de señas, actividades y juegos lúdicos en el proceso de aprendizaje, aplicaciones orientadas a la enseñanza y el término de inclusión a la inversa, conceptos que describirán la fundamentación para la elaboración de un prototipo de software que permita utilizar juegos didácticos como medio de inclusión a la inversa en poblaciones con problemas auditivos, además de los lenguajes, técnicas y procesos que se piensa utilizar.

5.4.1. Antecedentes de la Investigación.

5.4.1.1. *El lenguaje de señas.*

El lenguaje de señas o lenguaje de signos es una lengua natural del ser humano que permite expresarse mediante el uso de la configuración gesto espacial y percepción visual, es decir permite la comunicación mediante el uso de gestos y la percepción obtenida del movimiento de estos gestos. Este lenguaje de señas es la manera natural de comunicación que emplean las personas no oyentes, lo cual en palabras de (Pérez de Arado 2011), el lenguaje de gestos o señas es una lengua que, como cualquier otra, posee y cumple todas las leyes lingüísticas y se aprende dentro de la comunidad de usuarios a quienes facilita resolver todas las necesidades comunicativas y no comunicativas propias del ser humano.

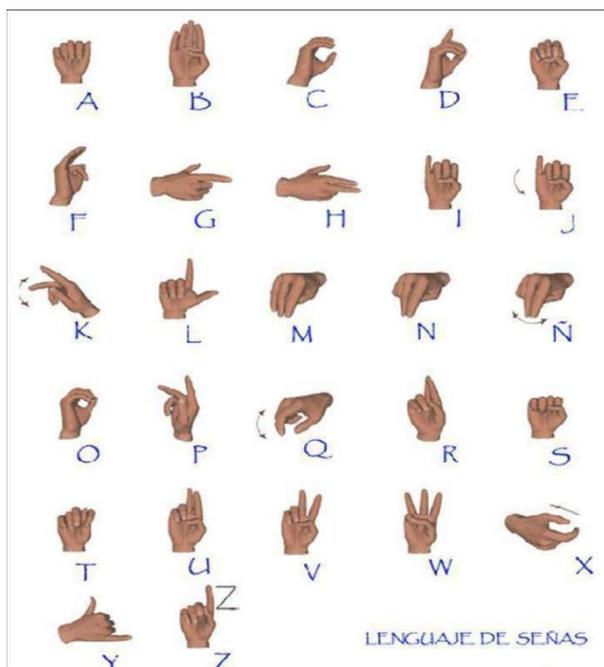


Figura 15. Lenguaje de señas (EcuRed s. f.)

5.4.1.2. Educación inclusiva.

La educación inclusiva según (Unesco 2006), se ve como el proceso de identificar y responder a la diversidad de las necesidades de todos los estudiantes a través de la mayor participación en el aprendizaje, las culturas, las comunidades, y reduciendo la exclusión en la educación, puesto que se basa en que cada individuo tiene necesidades de aprendizaje distintos, de acuerdo a esto cada grupo poblacional de según sus características necesitan un tipo o modelo determinado de enseñanza, partiendo de que la educación es un derecho fundamental el cual le permite al individuo competir al mismo nivel en la sociedad y le garantiza que disfrute en igualdad de condiciones a los demás. Por ende, la educación inclusiva determina que se debe adaptar la

manera de impartir el conocimiento de acuerdo con las necesidades de un individuo o población, y se le garantice el derecho fundamental a la igualdad y a la educación de acuerdo características especiales que esté presente, ya sea por problemas de aprendizaje, discapacidades etc., puedan afectar su normal desarrollo del aprendizaje.

En resumidas cuentas, según (Parra Dussan 2010:82) “La educación inclusiva se puede resumir en las transformaciones de la educación general y de las instituciones educativas para que sean capaces de dar respuesta equitativa y de calidad a la diversidad...”



Figura 16. Educación Inclusiva Recuperado de (Unesco 2008:1)

5.4.1.3. *Software orientado a la enseñanza.*

El avance de la tecnología se está implementando en el sector educativo, ayudando a optimizar los resultados obtenidos en el aprendizaje, partiendo de la premisa de la transformación digital donde es posible optimizar procesos, métodos y estrategias que apoyados con la innovación permiten aprender de una manera más interactiva y dinámica. Según (Muenta 2019), todos los niveles de enseñanza pueden beneficiarse de la tecnología para mejorar el proceso de aprendizaje, solo es necesario encontrar los recursos más adecuados y benéficos para cada una de las etapas.

Para orientar el software al contexto de la enseñanza, es necesario adquirir el termino de software educativo, el cual es una aplicación o programa orientado a la educación, de acuerdo con (Rovira Salvador 2018), el software educativo es un programa o conjunto de programas que están específicamente diseñados para facilitar y potenciar la adquisición de conocimientos exclusivamente académicos. Es decir, este debe ser su principal propósito y debe de estar señalado de manera explícita, permitiendo a los alumnos adquirir conocimiento de manera interactiva lo cual aumenta la posibilidad de reforzar las actividades realizadas o expuestas en clase; no obstante, este software permite al alumno no solo trabajar en las aulas, sino que también le permite acceder desde cualquier parte donde cuente con los medios necesario para acceder a este.

Las características que debe tener un software desde el ámbito educativo y psicológico según (Rovira Salvador 2018), es tener la posibilidad de ser usado en cualquier contexto de la educación, debe ser una herramienta interactiva, debe tener la capacidad de adaptarse a las necesidades del alumno (inclusión), debe ser fácil de usar y finalmente debe permitir al alumno crear sus propios conocimientos o resultados. También, este software deberá estar categorizado

de manera que se pueda identificar si el aprendizaje estará dado a través de ejercicios y práctica, simulación, tutoriales o juegos, y la función que este puede tener, de acuerdo con (Rovira Salvador 2018), este software debe cumplir con algunas características:

Tabla 3. Características o usos del software educativo (Fuente: Elaboración Propia)

| Características | Definición |
|-----------------|---|
| Innovación | El software debe ser innovador ya que este debe tener la capacidad de estar en constante evolución, aplicando las mejores tecnologías y teorías en la enseñanza y el aprendizaje. |
| Informativo | El software informático debe ser informativo, deberá exponer de manera clara y concisa los temas y actividades, además exponer y albergar la información que se le va a transmitir al alumno, apoyándolo de manera dinámica en el uso de la plataforma. |
| Motivador | El software deberá motivar a los alumnos para que se sientan más atraídos a la hora de su uso. |
| Instrucción | El software deberá tener la capacidad de explicar cualquier tipo de contenido. |
| Expresión | El software educativo según (Rovira Salvador 2018) deberá ser ideal para potenciar la expresividad de los alumnos, los cuales pueden aportar su propio punto de vista de los conocimientos y compartirlo con el resto. |
| Investigación | El software deberá permitir de manera directa al alumno elaborar sus propias hipótesis y procedimientos de investigación. |
| Lúdico | El software deberá ejecutar la enseñanza y el aprendizaje de manera lúdica y divertida que refuerce las ganas de aprender de los alumnos. |

5.4.1.4. *Aplicación Web Progresiva.*

Una Aplicación Web Progresiva o PWA (Progressive Web App) es una solución tecnológica basada en la Web, la cual puede ser desarrollada con una gran variedad de tecnologías orientadas al desarrollo de aplicativos FrontEnd, en donde su principal idea es emular las funciones propias, de las aplicaciones nativas, pero también partiendo como base la seguridad. (MDN, s. f.).

Para que una aplicación pueda llegar a considerarse como una web progresiva debe cumplir ciertos criterios o principios claves. A continuación, se da a conocer las características que poseen las aplicaciones web progresivas:

- *Detectable*, contenido fácil de encontrar a través de motores de búsqueda.
- *Instalable*, disponible en la pantalla inicial del dispositivo.
- *Enlazable*, compatible a través de una URL.
- *Independiente de la red*, puede funcionar sin necesidad de conexión de red.
- *Progresiva*, puede ser utilizada en un nivel básico en un navegador web anterior o funcionar completamente en uno más reciente.
- *Reconectable*, permite enviar notificaciones.
- *Adaptable*, puede ser accedido y usado desde cualquier dispositivo que cuente con pantalla y navegador.
- *Segura*, garantiza que las conexiones entre el usuario, la aplicación y el servidor estén protegidas contra terceros que traten de acceder a datos sensibles.

5.4.1.5. *Inclusión a la inversa*

El concepto de inclusión inversa o inclusión a la inversa está dado a promover la participación de las personas sin ningún tipo de deficiencia o discapacidad en las actividades diseñadas para las personas más vulnerables, permitiendo así la interacción de los dos grupos de poblaciones, adquiriendo de manera similar experiencia, conocimiento y apoyo para la generación de relaciones sociales, según (Valencia, Perlaza, y Ortiz 2019), esta actividad permite la familiaridad en el trato de cada una de las poblaciones, donde se reconocen que existen diferencias, pero que éstas no impiden relacionarse y compartir, siendo críticos ante una situación de conflicto, escuchando y utilizando la empatía como una forma de construcción en los procesos de mediación, generando una cultura de no rechazo, promoviendo una cultura de apoyo y de respeto hacia la diversidad de manera que se evidencia con normalidad la coexistencia con personas que presentan discapacidades.

5.4.1.6. *Los juegos en la enseñanza.*

El juego es una actividad innata del ser humano desde sus primeras etapas de crecimiento a manera de diversión y entretenimiento, el cual incentiva al individuo a entablar relaciones con otros individuos y su medio, permitiendo así adquirir conocimientos, ya sean de los medios con los que se está llevando la actividad, o de los individuos con los que se está ejerciendo esta, conociendo así características buenas o malas de la actividad que se está llevando, del medio, y adquiriendo experiencia de las situaciones y de la socialización que este conlleva.

Actualmente el juego se está implementando como herramienta de enseñanza ya que este incentiva al alumno a participar del aprendizaje, generar sus propias hipótesis, buscar la manera de resolver los problemas y todo de una manera lúdica y divertida, provocando que la absorción del conocimiento sea más placentera, según (Amaya y Maradiaga 2007:107), los juegos contribuyen al desarrollo de la acción, de la decisión, de la interpretación y de la socialización del individuo, de acuerdo con lo anterior se plantea el concepto de Gamificación, el cual según (Werbach & Hunter, 2012), nace de la idea de conseguir buenos resultados en dinámicas de trabajo, implementando mecánicas y técnicas típicas de los juegos en contextos fuera de ellos, apoyando a facilitar el aprendizaje a través de los juegos.

5.4.2. Bases o Fundamentos teóricos.

En este apartado se definirán las tecnologías utilizadas dentro del proyecto en la cual su fin será elaborar un prototipo de aplicación progresiva orientada al aprendizaje del lenguaje de señas a través del uso de una plataforma contenedora de juegos didácticos, cuyo fin sea promover el proceso de inclusión inversa, la cual incluirá: los conceptos claves, la arquitectura y otros factores para el desarrollo de esta actividad:

5.4.2.1. *Arquitectura propuesta.*

Para la elaboración del software o aplicativo nombrado Enseña2 se planea realizar un prototipo de aplicación web progresiva mediante el uso del patrón de diseño MVC (Modelo – Vista - Controlador), aplicado de acuerdo con la siguiente estructura:

5.4.2.1.1. Bases de datos:

Se emplea como sistema de gestión de base de datos MYSQL, el cual es un gestor de bases de datos relacionales licenciado bajo licencia Dual por Oracle, este sistema nos permite de manera gratuita desarrollar bases de datos relacionales mediante el uso de alguno de sus motores de bases de datos, en el caso de este aplicativo implementaremos la versión de MySQL Server 8.0.19.



Figura 17. Logo MySQL (MySQL 2020)

5.4.2.1.2. Backend:

Para el desarrollo del Backend y capa media se emplea el uso de la tecnología Spring Boot 2, la cual es una herramienta que tiene como fin simplificar el desarrollo de aplicaciones basadas en el framework Spring Core, el cual según (Cuervas 2019) “simplifica y facilita la construcción de aplicaciones JEE” en Java, añadiendo ventajas como lo son la inyección de dependencias, desarrollo sencillo de Pojos, simplificando los acceso a datos gracias a los ORM, y minimizando el código repetitivo, apoyándonos en tecnologías expuestas por Spring Boot tales como:

- **Spring Data:** Tecnología que según (Spring.io s. f.) facilitará el uso de las capas de acceso a datos mediante la implementación de Spring Data JPA, permitiendo la fácil conexión a la base de datos e implementación de repositorios JPA.
- **Spring Session:** tecnología que según (Spring.io s. f.) proporciona una API e implementaciones para administrar la información de la sesión de un usuario.

En el desarrollo este aplicativo se emplearán estas herramientas, permitiendo el uso de Spring Boot como herramienta para agilizar el desarrollo del Backend del aplicativo y el negocio, Spring Data para facilitar el uso de la capa de acceso a datos y facilitar la conexión con la base de

dato MySQL mediante el uso del repositorio JPA, Spring Session para administrar la información de sesión del usuario, y finalmente el uso de la tecnología Spring Rest Controller o controlador REST para exponer la información por medio del uso del protocolo Http mediante las operaciones POST y GET.



Figura 18. Logo Spring Boot (Cleventy 2020)

5.4.2.1.3. FrontEnd:

Para el desarrollo del FrontEnd se emplea inicialmente el uso de la tecnología Angular, la cual es un framework de código abierto para JavaScript creado por Google, que nos permite según (TORRES s. f.) crear aplicaciones SPA (Single Page Application). Lo cual nos dirá que nuestra aplicación cargará los datos sin interrumpir nuestra navegación con la aplicación, es decir, no hay necesidad de ir al servidor para procesar la información. Entre sus características principales podemos encontrar que su estructura está orientada al MVC, permite hacer uso de inyección de dependencias, permite usar lenguajes como TypeScript, JavaScript, o JQuery.

En el desarrollo de este aplicativo se emplea inicialmente angular en su versión 8 puesto que actualmente es la versión más estable para desarrollar la vista o la capa FrontEnd, el cual es un framework JavaScript, gratuito y Open Source, y que fue creado por Google para facilitar la creación de aplicaciones web de tipo SPA (Single Page Application).



Figura 19. Logo Angular (Angular.io 2020)

Acompañado al uso de angular se implementará el desarrollo de un Service Worker el cual es un archivo JavaScript compuesto de eventListeners que está pendiente de los eventos ejecutados, en otras palabras, los Service Workers según (Pérez 2020), son un servicio intermedio entre nuestro navegador e internet que actúa a modo de proxy y que intercepta toda la comunicación que se produce entre el dispositivo y la red. Este archivo se instala con la primera carga obtenida de la aplicación (hablemos de archivos HTML, JavaScript y el CSS) a través de una instrucción contenida en el archivo JavaScript, interceptando a partir de su instalación la petición de la página web, validando si esta información está almacenada en caché, corriendo en background.

En caso de contener la información en caché, el Service Worker retorna la data sin necesidad de ir a internet, de lo contrario que no cuente con esta información el Service Worker ira a internet cargará la información solicitada en caché, retornara la solicitud del usuario y la próxima vez que se realice esta petición ya no se deberá ir a internet para obtener la respuesta.

La característica principal y fundamental del uso del Service Worker en este proyecto está dado a mantener la disponibilidad aun sin conexión a internet, permitiendo escuchar eventos hasta que se establezca nuevamente la conexión, logrando así almacenar la información para poder

implementar esta aplicación progresiva como si fuese una app nativa, corriendo localmente en el dispositivo.

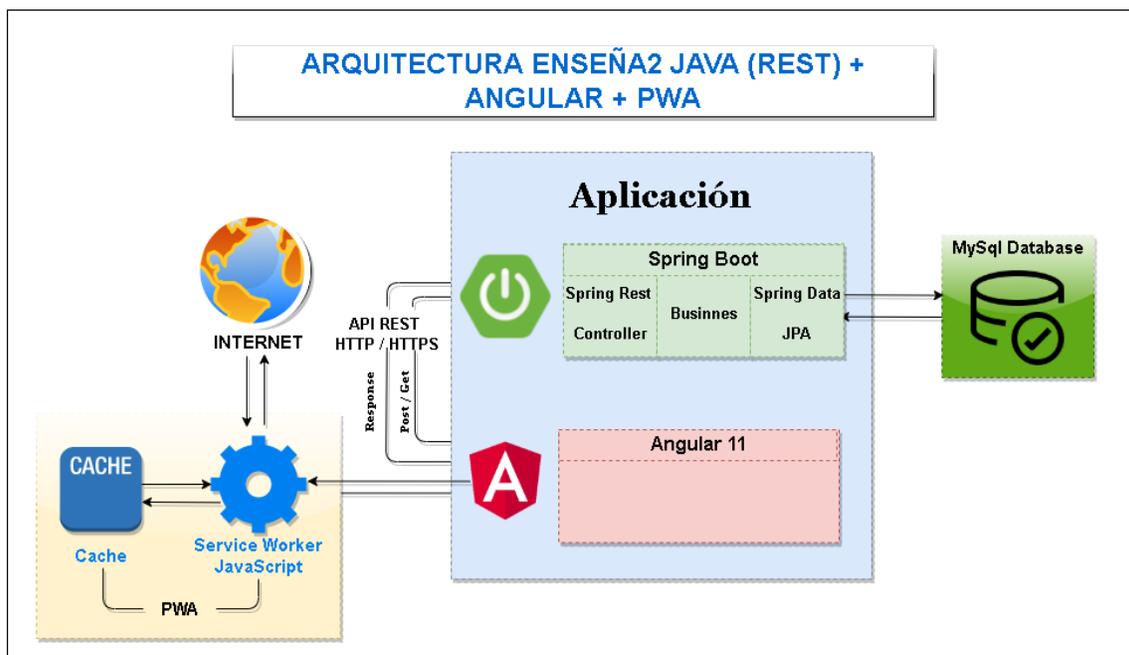


Figura 20. Diagrama de arquitectura (Fuente: Elaboración propia.)

5.4.2.2. Hardware y Software.

Para la elaboración del aplicativo nombrado Enseña2 es necesario la correcta distribución entre hardware y software utilizado para la implementación y despliegue del aplicativo, de acuerdo con la siguiente estructura:

5.4.2.2.1. Servidor Base de datos:

Para la implementación de base de datos es necesario un servidor Linux dedicado a base de datos con el software MySQL Server versión 8.0.23, el cual cuenta con las siguientes características mínimas de hardware:

- Procesador: Intel Dual Core 2GHz, AMD Ryzen 3 3.1GHz o Xeon 3GHz o superior
- Cores: Dual Core o superior.
- RAM: 4GB o superior.
- OS: Ubuntu 14.04 o superior, CentOS 6.x

5.4.2.2.2. Servidor de Aplicaciones Java:

Para la implementación del servidor de aplicativos (Spring Boot) es necesario un servidor Linux dedicado el cual cuenta con el siguiente software: Tomcat 9.0.43, JDK 8u261 o superior (Soporta hasta JDK 15.0.1), Spring Framework 5.0.7 Release, Maven 3.2+ y las siguientes características mínimas a nivel de hardware:

- Procesador: Intel Core I5 o Equivalente.
- HDD: 200 GB o superior
- RAM: 4GB o superior.
- OS: Ubuntu 14.04 o superior, CentOS 6.x

5.4.2.2.3. Servidor Web:

Para la implementación del servidor Web (apache) es necesario un servidor dedicado el cual cuente con el siguiente software: NodeJs 10.9.0 o superior, y con las siguientes características mínimas de hardware:

- Procesador: Intel Core I3 o Superior.
- HDD: 200 GB o superior
- RAM: 8GB o superior.
- OS: Ubuntu 14.04 o superior, CentOS 6.x

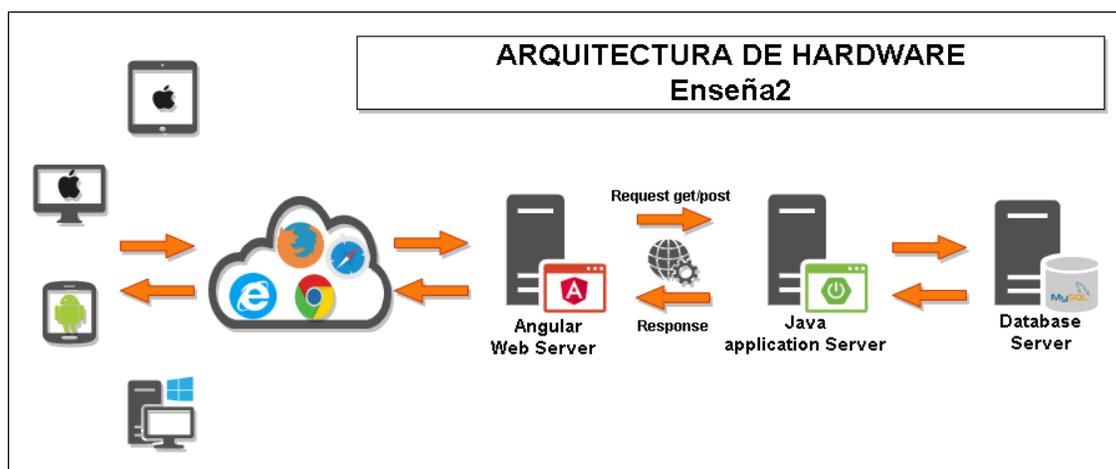


Figura 21. Diagrama de Hardware (Fuente: Elaboración Propia)

5.4.3. Definición de términos básicos.

En este apartado se definirá los términos utilizados dentro del proyecto, conceptos empleados en el análisis, identificación, formalización del problema y desarrollo del problema, términos que se emplean para la elaboración del prototipo de aplicación progresiva orientada al aprendizaje del lenguaje de señas a través del uso de juegos didácticos:

- **Inclusión inversa o inclusión a la inversa:** La inclusión inversa busca la participación de todo tipo de población en actividades diseñadas para personas con discapacidad o algún tipo de vulnerabilidad, permite que el ejercicio resulte apropiado para todos.
- **No oyente:** Persona con dificultad diagnosticada para oír cualquier tipo de sonido, las personas no oyentes suelen comunicarse por el lenguaje de señas.
- **Lenguaje de Señas:** Es el lenguaje de expresión gesto-espacial, que se utiliza como canal de comunicación por personas no oyentes o con discapacidad.
- **Actividad lúdica:** Son las actividades que fomentan la imaginación, la creatividad, la concentración, el entretenimiento y el desarrollo de conocimiento o adquisición de conceptos.
- **Prácticas ágiles:** Es un marco de trabajo diseñado para la gestión y el seguimiento de proyectos, se caracterizan por manejar prácticas iterativas e incrementales.

- **Cohesión:** Es una medida que determina que el componente esté realizando únicamente la tarea asignada, en el diseño de software se recomienda que la cohesión sea alta para que el diseño tenga éxito.
- **Acoplamiento:** Es una medida que determina la dependencia de un componente con los demás, en el diseño de software se recomienda que el acoplamiento sea bajo para que el diseño tenga éxito.
- **Encapsulamiento:** Es un principio que permite ocultar el estado interno de un objeto para que toda la interacción se realice por los métodos que lo componen.
- **Polimorfismo:** Es una propiedad que permite enviar mensajes sintéticamente iguales a objetos distintos, estos deben tener la capacidad de responder al mensaje.
- **Abstracción:** Es un principio que permite aislar características relevantes de su contexto original para definir nuevos tipos de entidades.
- **Herencia:** Es un principio que permite a los objetos heredar propiedades y comportamientos de las clases a las que pertenece.
- **Software:** Conjunto de pasos organizados con lógica implementada para ejecutar una tarea específica.

- **Bases de datos:** Es un programa que permite alojar una gran cantidad de datos estructurados, este es el encargado de administrar dicha información.
- **Backend:** Es el componente lógico que alimenta la interfaz gráfica visualizada por el cliente, garantizando la funcionalidad de los programas.
- **Middleware:** Es un software que asiste programas, aplicaciones o paquetes para que este se pueda comunicar con otros sistemas.
- **FrontEnd:** Es el componente gráfico que visualiza el usuario y sobre el que puede interactuar con el sistema, este también es conocido como la capa de presentación.

6. Metodología

Para la elaboración de este proyecto se definió utilizar el marco de trabajo Scrum, dado que es una metodología que incentiva el agilismo, este marco de trabajo está representado bajo 3 artefactos que están diseñados para maximizar el orden y la transparencia de la información, estos artefactos están definidos como: el Product Backlog, como primer artefacto es una lista ordenada, elaborada a través de los requerimientos generados y ordenados por el Product Owner; el segundo artefacto es el Sprint backlog, que son los elementos seleccionados para ser ejecutados en el Sprint en curso; y como tercero tenemos el Incremento, el cual es la suma de todos los elementos del Sprint Backlog terminados (Que cumplen la definición de hecho).

El marco de trabajo Scrum se ejecuta por un equipo de trabajo compuesto por el Product Owner, el Scrum Master, y el Equipo de desarrollo. Para la elaboración de este ejercicio, se conformó un equipo de trabajo que consta de un Product Owner el cual será representado por Jimmy Carrillo Bernal (Traductor del lenguaje de señas), por un Scrum Máster (Cristian Rodríguez Bernal) y finalmente por el equipo de desarrollo compuesto por (Yeinner Zambrano Santiago & Cristian Rodríguez Bernal).

Para la correcta ejecución de la fase de desarrollo del aplicativo, el proyecto contó con un total de 6 Sprints, los cuales fueron organizados a manera de permitir el correcto desarrollo de las ceremonias de Scrum para cumplir el control de sus procesos, tomando la semana número uno para realizar la actividad de Agile Inception, la cual es una serie de dinámicas dirigidas a enfocar el equipo conformado de Scrum hacia un mismo objetivo, permitiendo reducir esta según (Jmbeas 2013) muchas de las incertidumbres, ayudando a explicitar los riesgos más evidentes y permitiendo

conocer en común las expectativas de todos, adicional se utilizará el tiempo definido en esta semana para el alistamiento del ambiente de desarrollo.

A partir de la semana número dos se trabajarán Sprints de dos semanas para un total de 6 Sprints en los que se elaborara el desarrollo, pruebas y documentación de aplicativo. Acorde a lo anterior se celebran las siguientes ceremonias dentro de cada iteración de los Sprints:

- **Sprint Planning:** Con una duración de 4h por Sprint a través de video llamada, en la cual se reunió el equipo completo de Scrum para determinar qué ítems del Product backlog se pueden entregar en este Sprint y las estrategias para poder cumplir la definición de terminado.
- **Daily Scrum:** Con una duración de 15 minutos en el día laboral con la reunión del Scrum Master y el Equipo de desarrollo a través de videollamada, donde se indican las tareas realizadas el día anterior, las tareas que se realizarán en el día actual y si se ha tenido algún inconveniente con el avance del desarrollo y sus tareas realizadas del Sprint Backlog.
- **Sprint Review:** Con una duración de 2h por Sprint, en la cual se reúne el equipo Scrum para identificar las tareas realizadas en el Sprint a través de videollamada, en donde se indica que problemas se presentaron y se identifican las tareas terminadas y faltantes del Product Backlog.

- **Sprint Retrospective:** Con una duración de 1.5h por Sprint a través de videollamada, en el cual se identifica las posibles mejoras del Sprint anterior, las actividades que salieron bien y se elabora un plan o camino para implementar estas mejoras.

7. Proceso de Software

En este apartado encontraremos todos los puntos arquitecturalmente relevantes, partiendo desde lo más básico como lo son las historias de usuario elaboradas juntamente con el Product Owner del proyecto, siguiendo con los diseños del sistema tales como los casos de uso, los diagramas de clases, diagramas de secuencia, diagramas de despliegue para llegar finalmente a los diagramas de arquitectura de alto nivel y de componentes hasta un primer nivel de visualización:

7.1. Requerimientos Funcionales y No Funcionales

Dado que el presente proyecto se ejecutó bajo el marco de trabajo ágil SCRUM, los requerimientos funcionales y no funcionales fueron reemplazados por historias de usuarios, en donde se identificaron las épicas a partir de la agrupación de los casos de usos relevantes de la aplicación, partiendo de estas se generaron las historias de usuario de acuerdo a el rol de los usuarios, sus necesidades y la forma en que estos serán solventados, estableciendo también los criterios de aceptación para el cumplimiento de estas.

Para la redacción de las historias de usuarios, en conjunto con el product owner se especificó un total de tres épicas, las cuales permitieron agrupar las historias de usuarios individuales en cada una de estas. Cabe mencionar que para el diseño de las historias de usuarios se tuvieron en cuenta los Casos de Usos definidos de acuerdo con la especificación de P.O y que se presentan en el punto 7.2.2 del presente documento, a continuación, se presenta las siguientes tablas en las cuales se dan a conocer las historias de usuarios:

Tabla 4. Historias de usuarios (Fuente: Elaboración Propia)

| ID Historia | Historia | Criterios de Aceptación |
|----------------------|---|--|
| <p>HU-001</p> | <p>El usuario del sistema quiere poder registrarse y el formulario de registro solo debe solicitar los siguiente campos: Nombre, Apellidos, Correo Electrónico, Teléfono Celular y Contraseña en donde esta última deberá ser cifrada para mayor seguridad, Estos datos deberán mantener el formato que corresponde al tipo de dato que almacenan; de igual manera, el usuario podrá actualizar los datos anteriores; todo esto se requiere para que el usuario pueda crear y mantener actualizado y seguro los datos de perfil dentro del sistema.</p> | <ol style="list-style-type: none"> 1. El formulario de registro solo debe contener los siguientes campos: Nombre, Apellido, Correo Electrónico, Teléfono Celular y Contraseña. 2. Los campos Nombre y Apellido solo deberán admitir letras en mayúscula y minúscula y demás acentos ortográficos del lenguaje español. 3. El Correo electrónico deberá verificar que sea un correo válido y que mantenga la siguiente estructura: micorreo@email.com. 4. El campo de Teléfono celular de usuario solo debe permitir números y debe tener un formato móvil estándar colombiano. 5. El campo contraseña deberá tener mínimo 8 caracteres y máximo 12 caracteres. La contraseña no debe tener espacios y deberá solicitar al menos una letra en mayúscula, una letra en minúscula, 1 número y 1 carácter especial. 6. El usuario podrá actualizar sus datos dentro del sistema y los campos deberán tener las siguientes validaciones: Nombre y Apellido solo deberán admitir letras en mayúscula y minúscula y demás acentos ortográficos del lenguaje español, El campo de Teléfono celular de usuario solo debe permitir números y debe tener un formato móvil estándar de colombiano. |

HU-002

El usuario del sistema quiere que en la ventana de Inicio de Sesión el campo Contraseña esté protegida y permita visualizar o no la contraseña y que debajo del botón Login, dé la opción de recuperar la contraseña en el cual se pueda acceder y abrir una nueva ventana para restablecer las credenciales de acceso.

-
7. El sistema deberá implementar el algoritmo de cifrado BCrypt de Spring Boot para el registro de la contraseña.
 8. El sistema deberá tener un tiempo de respuesta en los registros y/o actualizaciones de usuarios no mayor a 0,05 segundos.
-
1. El sistema deberá ocultar la contraseña.
 2. El campo Contraseña en el inicio de sesión deberá tener el botón de ver password el cual permita visualizar la contraseña digitada u ocultarla.
 3. La nueva contraseña deberá tener mínimo 8 caracteres y máximo 12 caracteres. La contraseña no debe tener espacios y deberá solicitar al menos una letra en mayúscula, una letra en minúscula, 1 número y 1 carácter especial.
 4. Deberá haber otro campo de confirmación de contraseña en el cual se pueda verificar la contraseña registrada.
 5. El sistema deberá rechazar el cambio de la contraseña cuando esta sea similar a la anterior mostrando un mensaje de advertencia indicando que la contraseña debe ser una nueva y especificando las características que debe tener la misma.
 6. El sistema deberá implementar el algoritmo de cifrado BCrypt de Spring Boot para el registro de la contraseña.
 7. El tiempo de respuesta del update de la contraseña en la BD no deberá ser mayor a 0,05 segundos.
-

HU-003

El usuario del sistema quiere poder iniciar sesión en cualquier momento del día desde cualquier dispositivo y en cualquier navegador y que cuando desee cerrar sesión debe ser por un botón Logout, de igual manera el sistema deberá contar con certificados de seguridad web, el sistema deberá ejecutarse en dispositivos con capacidades mínimas del sistema, todo esto para tener accesibilidad al sistema, iniciar sesión, usar los servicios de este y poder finalizar sesión cuando lo desee.

1. El sistema deberá estar disponible 24/7.
2. La interfaz de conexión debe ser navegable en las últimas versiones de Google Chrome y Mozilla Firefox u otros navegadores que soporten Java.
3. El sistema deberá correr en equipos o dispositivos móviles que cuenten con al menos 1gb de memoria RAM, 8 Gb de almacenamiento en disco y procesadores con mínimo 1,6 GHz de velocidad de procesamiento.
4. El sistema deberá poder ser accedido desde cualquier dispositivo o equipo con un Sistema Operativo de Microsoft Windows o Android en cualquier versión de estos siempre y cuando tengan un navegador web que soporte la tecnología HTTPS y Java.

HU-004

El usuario del sistema quiere poder enviar un formulario PQR con una solicitud, queja o reclamo y que dicho formulario solicite los siguientes datos: Nombre y Apellido, Correo Electrónico, Teléfono Celular, Asunto y Descripción, y estos campos deben mantener un formato de acuerdo con el tipo de dato que almacenan, para poder enviar una solicitud ya sea a un traductor LSC o al equipo de desarrollo de la plataforma.

1. El Formulario PQR deberá contener los siguientes campos: Tipo Solicitud, Nombre y Apellido del solicitante, Correo Electrónico, Teléfono Celular, Asunto y Descripción.
2. El Nombre y Apellido del usuario deberá cargarse en el campo Nombre de Usuario del formulario.
3. El Correo Electrónico del usuario deberá cargarse en el campo correo electrónico del formulario.
4. El Teléfono celular del usuario debe cargarse en el campo Teléfono del formulario.
5. El campo Asunto deberá ser una caja de texto que solo admite hasta 150 caracteres.

6. El campo Descripción Deberá ser una caja de texto que admite hasta 1000 caracteres.
7. Cuando el usuario envíe el formulario PQR este deberá almacenarse en la BD.

Tabla 5. Historias de usuarios - Épica Administrador (Fuente: Elaboración Propia)

| Épica | | Administrador |
|---------------------|--|---|
| Descripción: | <p>El Administrador del sistema es un usuario que tendrá la capacidad de administrar los 3 módulos principales del sistema: Usuarios, Juegos y Aprendizaje. El administrador podrá actualizar la información de contacto y perfil de Usuario, de igual manera podrá habilitar/deshabilitar usuarios del sistema; el administrador podrá gestionar el módulo Administración de Juegos habilitando o deshabilitando los juegos disponibles en las temáticas establecidas dentro del sistema; y, el administrador del sistema podrá gestionar el módulo Administrar Temas de Aprendizaje en el cual habilitará o deshabilitará los temas que se encuentren en las lecciones del sistema de acuerdo los temas: Números, Letras y Palabras.</p> | |
| ID Historia | Historia | Criterios de Aceptación |
| HU-001 | <p>El administrador del sistema quiere que en el módulo de Administrar Usuarios este la opción de actualizar información de Usuario y los campos que se podrán editar deben ser: Correo Electrónico y Contraseña; también debe estar una opción para habilitar/deshabilitar usuarios del sistema; para mantener la información de los usuarios actualizada.</p> | <ol style="list-style-type: none"> 1. Para actualizar usuarios el sistema deberá solicitar correo del usuario, y una vez el administrador lo digite el sistema deberá mostrar una ventana con los datos del usuario a actualizar. 2. Si el cambio es sobre el campo Correo Electrónico de Usuario el sistema deberá verificar que sea un correo válido y que mantenga la siguiente estructura: micorreo@email.com. 3. Los Update sobre la BD de las actualizaciones que realice el |

administrador por medio del sistema no deberán ser superior a 0,05 segundos.

HU-002

El administrador del sistema quiere que en el módulo Aprendizaje esté la opción de configurar temas ya sea para habilitar o deshabilitar los temas que pueden ser: Números, Letras y Palabras; una vez sea deshabilitado un tema este debe bloquearse y bloquear las lecciones, evaluaciones y juegos disponibles que estén relacionados al tema deshabilitado o el caso contrario; todo esto se requiere para tener control sobre las lecciones ofrecidas por tema y que estarán disponibles para el público.

1. Cuando el administrador ingrese a la opción Configurar Temas deberá visualizar las opciones Número, Letras y Palabras.
2. Al lado de cada tema deberá haber un Switch button que permite al administrador habilitar o deshabilitar los temas.
3. El sistema deberá actualizar los estados de las lecciones en la BD en un tiempo no mayor a 0,05 segundos y debe ser habilitado/deshabilitado del sistema en un tiempo no mayor a 2 minutos.
4. El proceso de habilitación/deshabilitación de un tema no debe afectar el avance del aprendizaje de los usuarios y no debe tardar más de 1 segundo en efectuarse los cambios sobre el sistema y tampoco más de 2 minutos en ser aplicados en los dispositivos de los usuarios.

HU-003

El administrador del sistema quiere que en el módulo Aprendizaje este la opción de configurar Juegos y en esta se puedan habilitar o deshabilitar los juegos disponibles en los Temas establecidos dentro del sistema, y una vez sea deshabilitado un juego este debe bloquearse y ocultarse del listado de juegos por temas disponibles que tienen los usuarios o el caso contrario; todo esto se requiere para tener control sobre los juegos que estarán disponibles para el público.

1. Cuando el administrador ingrese a la opción Configurar Juegos, el sistema deberá tener un listado de los juegos disponibles en la plataforma.
 2. Al lado de cada juego deberá haber un Switch button que permita al administrador habilitar o deshabilitar los juegos.
 3. El sistema deberá actualizar los estados de los juegos en la BD en un tiempo no mayor a 0,05 segundos y debe ser habilitado/deshabilitado del sistema en un tiempo no mayor a 2 minutos.
-

-
4. Los juegos deben estar enlazados a las Temas registrados en el módulo de Aprendizaje.
 5. El proceso de habilitación/deshabilitación de un juego no debe tardar más de 1 segundo en efectuarse los cambios sobre el sistema y tampoco más de 2 minutos en ser aplicados en los dispositivos de los usuarios.
-

HU-004

El administrador del sistema quiere que en el módulo Aprendizaje, las opciones de lección que están dentro de cada Tema se pueda habilitar/deshabilitar; todo esto se requiere para tener control sobre las enseñanzas y evaluaciones brindadas por tema y que estarán disponibles para el público.

1. Cuando el administrador ingrese a la opción Configurar Lecciones, el sistema deberá tener un listado de las lecciones disponibles categorizadas por Temas.
 2. Al lado de cada lección deberá haber un Switch button que permita al administrador habilitar o deshabilitar las lecciones.
 3. El sistema deberá actualizar los estados de las lecciones en la BD en un tiempo no mayor a 0,05 segundos y debe ser habilitado/deshabilitado del sistema en un tiempo no mayor a 2 minutos.
 4. Las lecciones deben estar enlazados a los temas registrados en el módulo de Aprendizaje.
 5. Una vez se habilite/deshabilite una lección el sistema deberá habilitar/deshabilitar la lección en conjunto con la evaluación registrada de la misma.
 6. El proceso de habilitación/deshabilitación de una lección no debe afectar el avance del aprendizaje de los usuarios y no debe tardar más de 1 segundo en efectuarse los cambios sobre el sistema y tampoco más de
-

2 minutos en ser aplicados en los dispositivos de los usuarios.

Tabla 6. Historias de usuarios - Épica Aprendizaje (Fuente: Elaboración Propia)

| Épica | Aprendizaje | |
|---------------------|--|--|
| Descripción: | <p>Cuando el usuario ingrese al menú de aprendizaje, el sistema desplegará un listado con las siguientes opciones: Números, Letras y Palabras y que cada una muestre las lecciones disponibles para cada tema, estas lecciones deberán tener imágenes que las caracterice y una breve descripción y un botón de acceso a la lección para poder ver el avance logrado en el aprendizaje del lenguaje de señas por medio de juegos lúdicos. Por cada una de las lecciones por tema habrá una evaluación que medirán el avance alcanzado por el usuario. En la opción Consultar Avance de aprendizaje el sistema mostrará el avance en porcentaje por cada una de las lecciones registradas en los siguientes temas: Números, Letras y Palabras, mostrando cuáles son las lecciones de los temas que el usuario debe completar. Por último, el sistema deberá permitir registrar una hora por día y un tiempo límite para seguir con el aprendizaje en el aplicativo y deberá ser almacenada en la BD en un tiempo menor a 0,05 segundos.</p> | |
| ID Historia | Historia | Criterios de Aceptación |
| HU-001 | <p>El usuario del sistema quiere que el menú de aprendizaje despliega un listado con los siguientes temas: Números, Letras y Palabras y cada una debe mostrar las lecciones disponibles para cada tema, estas lecciones deberán tener imágenes que las caracterice y un botón de acceso a la lección para poder ver el contenido de estas.</p> | <ol style="list-style-type: none"> 1. Los temas disponibles en el menú de aprendizaje son: Números, Letras y Palabras. 2. Cuando el usuario acceda a un tema, el sistema deberá desplegar el listado con las lecciones disponibles para dicho tema. 3. El usuario podrá acceder a una lección y el sistema mostrará su contenido. 4. El sistema deberá presentar el contenido de la lección escogida por el usuario. 5. Las lecciones deberán contar con un checkbox que indique cuáles temas ya fueron cumplidos por el usuario. |

| | |
|--|---|
| | <ol style="list-style-type: none"> 6. Cada imagen de los temas deberá tener un botón para ingresar al listado de lecciones del tema escogido. |
| <p>HU-002</p> <p>El usuario del sistema quiere que cuando termine alguna lección el sistema habilite un examen lúdico que permita evaluar su aprendizaje y aumentar el avance que tiene en el sistema.</p> | <ol style="list-style-type: none"> 1. El sistema deberá desplegar un examen siempre y cuando el usuario finalice la lección. 2. Cada evaluación deberá tener recursos multimedia. 3. La evaluación deberá desplegar un listado de preguntas. 4. Los recursos multimedia deberán ser archivos de bajo peso y que mantenga una buena calidad. 5. Sí la calificación por el examen que realice el usuario supera el 50% dará como culminado la lección y registrará el avance del usuario por temática. |
| <p>HU-003</p> <p>El usuario del sistema quiere que en la opción Consultar Avance de aprendizaje muestre el avance en porcentaje por cada una de las lecciones registradas en los siguientes temas: Números, Letras y Palabras, de igual manera la plataforma deberá mostrar cuales son las lecciones de los temas que el usuario debe completar para que el usuario pueda estar al día de lo que ha logrado dentro del sistema.</p> | <ol style="list-style-type: none"> 1. El sistema deberá mostrar el porcentaje de progreso de lo aprendido en el uso del sistema. 2. Una lección será finalizada siempre y cuando el usuario complete todo el recorrido del contenido de la lección para que se registre el avance del usuario. |
| <p>HU-004</p> <p>El usuario del sistema quiere poder entrar en las opciones de juegos dentro de los temas ubicados en el módulo Aprendizaje y ver los juegos disponibles; los juegos disponibles deben poder ser ejecutados.</p> | <ol style="list-style-type: none"> 1. El sistema deberá mostrar un listado de los juegos disponibles o habilitados. En el caso que un juego esté deshabilitado, deberá ocultarse del listado de juegos. 2. Los juegos en la lista de juegos deben mostrar una breve descripción de su |

contenido y las instrucciones para poder jugarlo.

3. cada juego deberá tener un botón de “Iniciar” el cual cuando se presione debe abrir una nueva ventana en la cual ejecute el juego.

HU-005

El usuario del sistema quiere que haya un menú de opciones en el cual se pueda controlar de manera externa el juego ejecutado. Estas opciones deben ser: “Reiniciar” y “Finalizar” el juego ejecutado, y una vez finalice el juego el sistema deberá cerrarlo.

-
1. El sistema deberá tener un menú con los siguientes botones: “Reiniciar” y “Finalizar”.
 2. El botón reiniciar deberá cerrar y volver a ejecutar el juego seleccionado en una nueva ventana.
 3. El botón finalizar deberá cerrar la ventana del juego ejecutado.
 4. El sistema deberá mostrar en el cuadro de mensaje dos botones con las opciones de “Continuar” y “Finalizar”.
 5. El sistema deberá mostrar un cuadro de mensaje indicando que el usuario está por finalizar el juego y deberá preguntar si desea continuar.
 6. Cuando el usuario Finaliza el juego, el sistema deberá dirigirlo a la ventana del listado de juegos.
 7. Cuando se reinicia un juego el sistema no deberá tardar más de 2 segundos para cerrar la ventana del juego y volverla abrir.
 8. Cuando el juego finaliza el sistema no deberá tardar más de 3 segundos en cerrar el juego.
-

7.2. Diseño y Arquitectura

En este apartado trabajamos específicamente los diagramas a nivel de diseño que componen el sistema, como lo son: el diagrama de despliegue, con el cual identificamos los requisitos mínimos y el hardware necesario para el correcto despliegue del sistema Enseña2, el diagrama de casos de uso, en el cual a través de modelados de caso de uso podremos identificar los actores del sistema y sus actividades en el sistema. El diagrama de secuencia, en el cual podremos observar el ciclo de vida de las acciones ejecutadas por los actores del sistema, el diagrama de clases, el cual especifica a nivel de clases el esqueleto o estructura del sistema y sus componentes a través de clases y finalmente, el diagrama de arquitectura de alto nivel, el cual representa una visión general de los sistemas y su respectiva interacción y distribución a nivel arquitectónico.

7.2.1. Diagrama de Despliegue

De acuerdo con las características requeridas de hardware para la exposición de la aplicación progresiva Enseña2, es necesario la implementación de tres servidores para el despliegue del sistema como se plantea a continuación:

El primer servidor, está encargado de la funcionalidad web, mediante el uso de un sistema operativo Linux, el cual cuenta con un servidor para aplicaciones Web HTTP Apache 2.4.46, en donde se alojarán la aplicación web o FrontEnd desarrollado con el framework de JavaScript Angular Js 11, y que a su vez cuenta con un módulo capturador de eventos o el service worker que trabaja a manera de proxy.

El segundo servidor está encargado del Backend, mediante el uso de un sistema operativo Linux, que cuenta con el servidor de aplicaciones Apache Tomcat 9.0.43, en donde se aloja el Backend del sistema, compuesto por un aplicativo Spring Boot 2 (Spring Data JPA, Spring Sesion y Spring Rest Controller) bajo el Framework 5.0.7 Release y el uso de las librerías de Maven en su versión 3.2. Este servidor se comunica con el servidor Web a través del protocolo TCP/IP.

El tercer el servidor está encargado de alojar la Base de Datos, mediante el uso de un sistema operativo Linux y que a su vez cuenta con el servidor de base de datos MySQL Server en su versión 8.0.23, este servidor se comunica con el servidor de aplicación a través del protocolo TCP/IP por el puerto 3306, de acuerdo con lo anterior, a continuación, se presenta el diagrama de despliegue el cual permite observar la distribución del hardware del sistema:

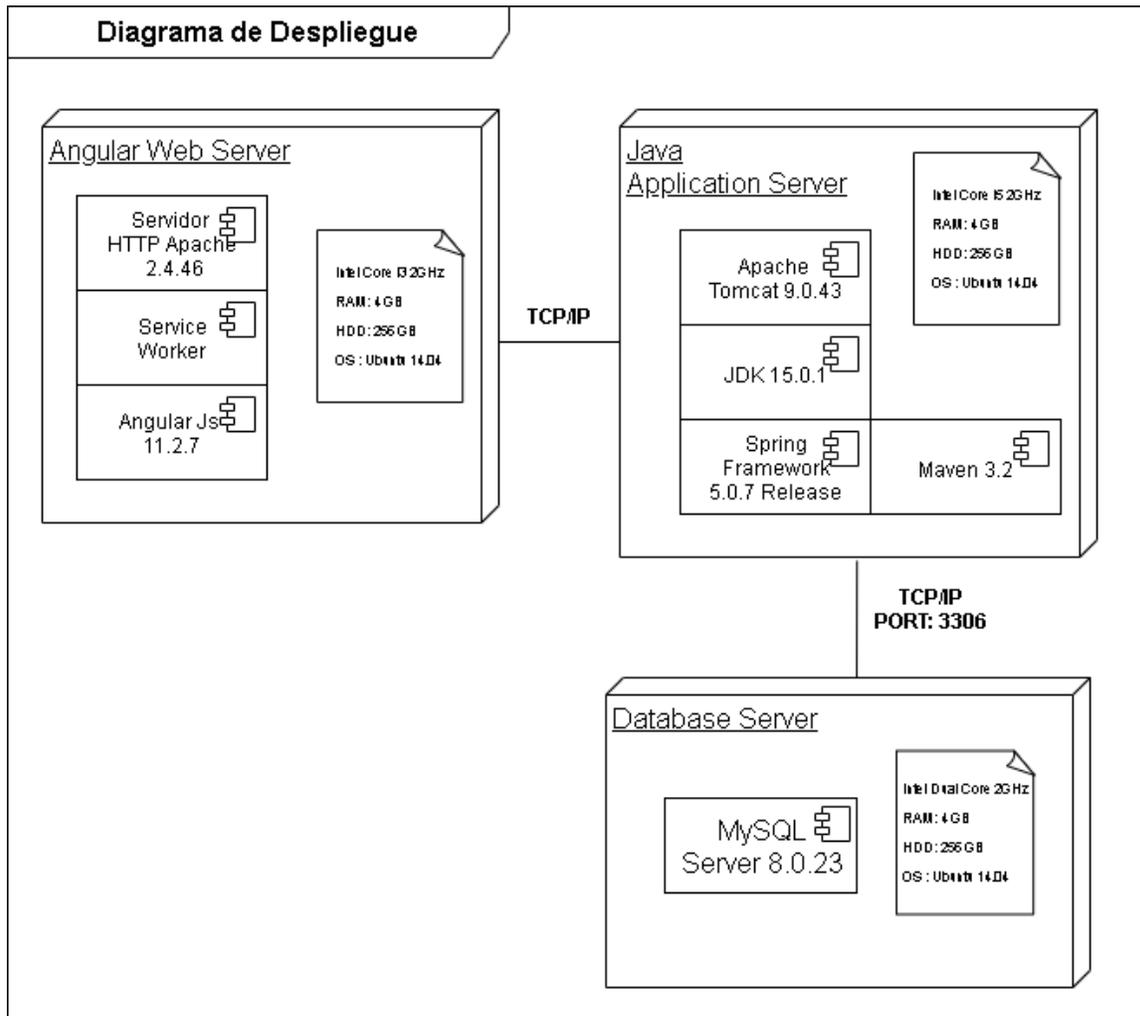


Figura 22. Diagrama de despliegue (Fuente: Elaboración propia.)

7.2.2. Casos de Usos Arquitecturalmente Relevantes

Para el diseño del sistema se identificaron los actores que intervienen en el ciclo de vida de la aplicación progresiva Enseña2, de acuerdo con esto, los actores se encuentran catalogados de la siguiente manera: un actor que representa a los usuarios finales y un actor que representa al usuario administrador del sistema. Partiendo desde el punto de vista de los actores se modelaron los siguientes casos de usos los cuales permiten observar el comportamiento funcional del sistema y la implicación de los actores con cada uno de estos casos de uso, de acuerdo con los siguientes diagramas:

El diagrama de caso de uso general representa una visión general de los actores y su interacción con los respectivos casos de uso del sistema, los cuales se encuentran organizados jerárquicamente, a continuación, veremos el diagrama de caso de uso general:

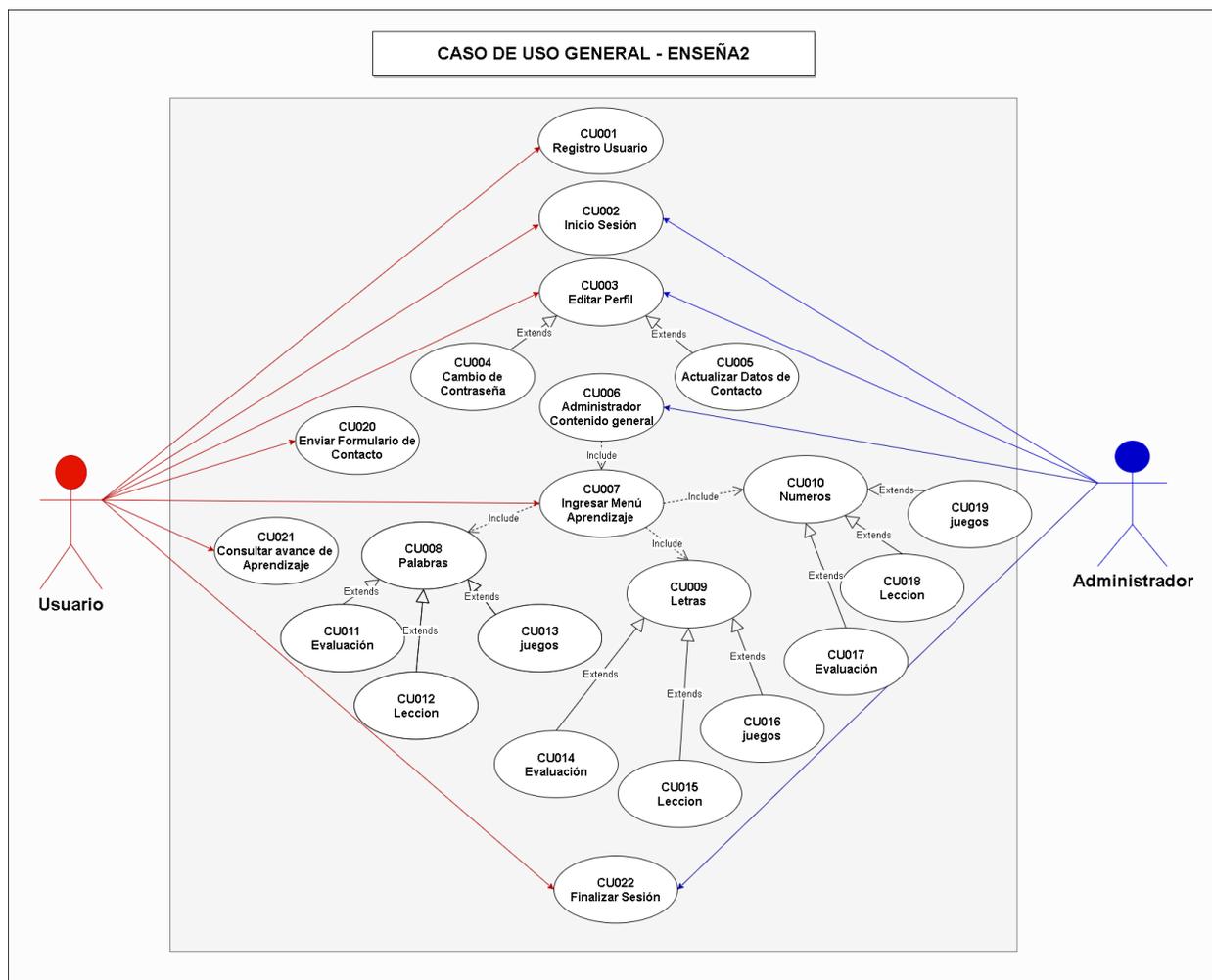


Figura 23. Diagrama de Caso de Uso General (Fuente: Elaboración propia.)

El diagrama de caso de uso específico por actor representa una visión individual de cada uno de los actores y su interacción con los respectivos casos de uso del sistema de acuerdo con el flujo con el que pueden interactuar en el aplicativo, los cuales se encuentran organizados jerárquicamente, a continuación, se presentan los casos de uso específico por actor:

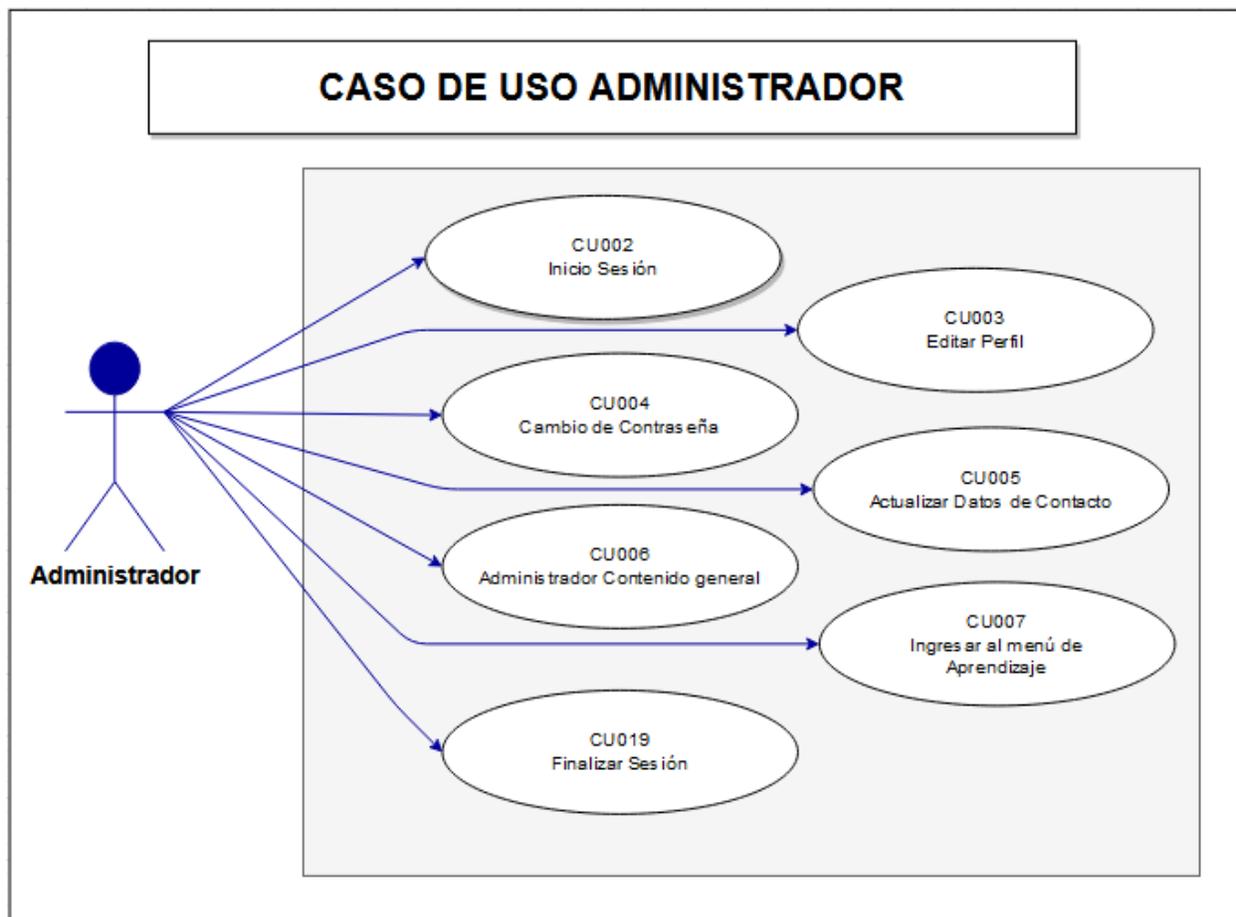


Figura 24. Diagrama de Caso de Uso Admón. (Fuente: Elaboración propia.)

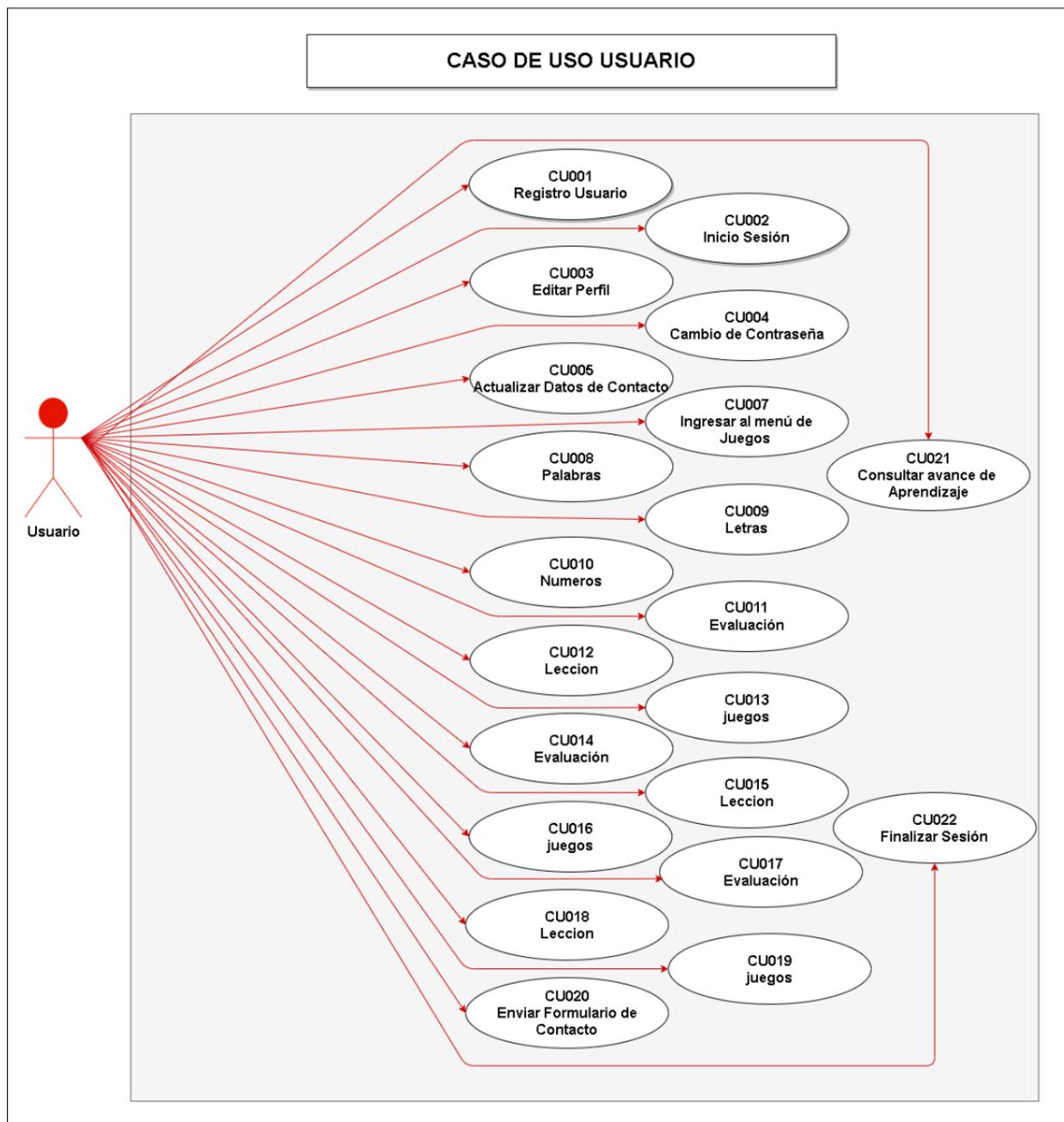


Figura 25. Diagrama de Caso de Uso Usuario (Fuente: Elaboración propia.)

7.2.3. Diagrama de Secuencia

Con el fin de mostrar la secuencia de las acciones ejecutadas por los actores en el aplicativo y que se encuentran definidos en el comportamiento de los casos de uso previamente vistos en el punto anterior, se presenta a continuación los diagramas de secuencias que se encuentran distribuidos de acuerdo a las épicas (Agrupación de casos de uso) definidas en las historias de usuarios, estos diagramas representan los eventos en orden cronológico dentro del sistema y la respuesta contemplada dentro del mismo. A continuación, podremos observar los diagramas de secuencias:

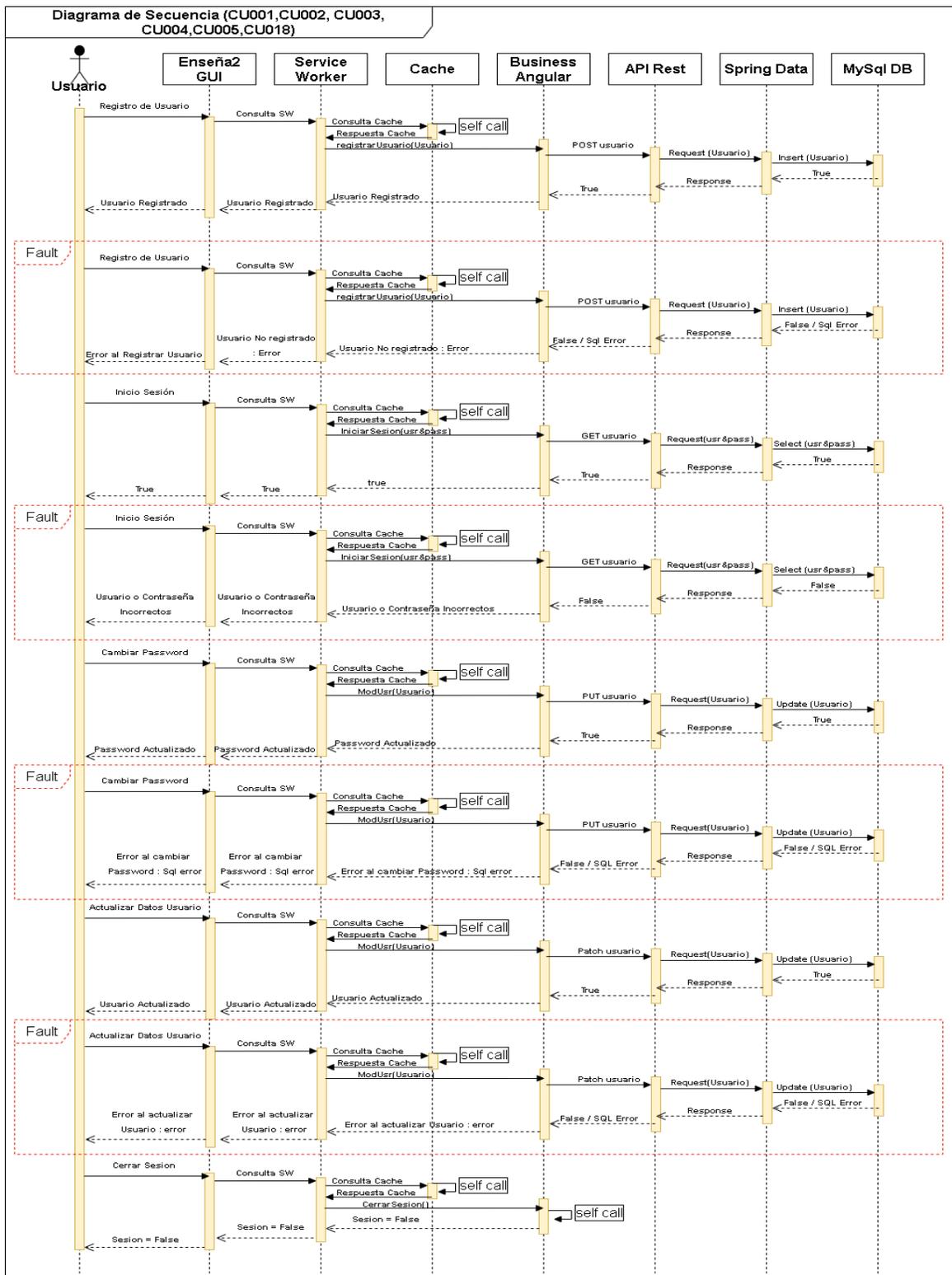


Figura 26. Diagrama de Secuencia Usuarios (Fuente: Elaboración propia.)

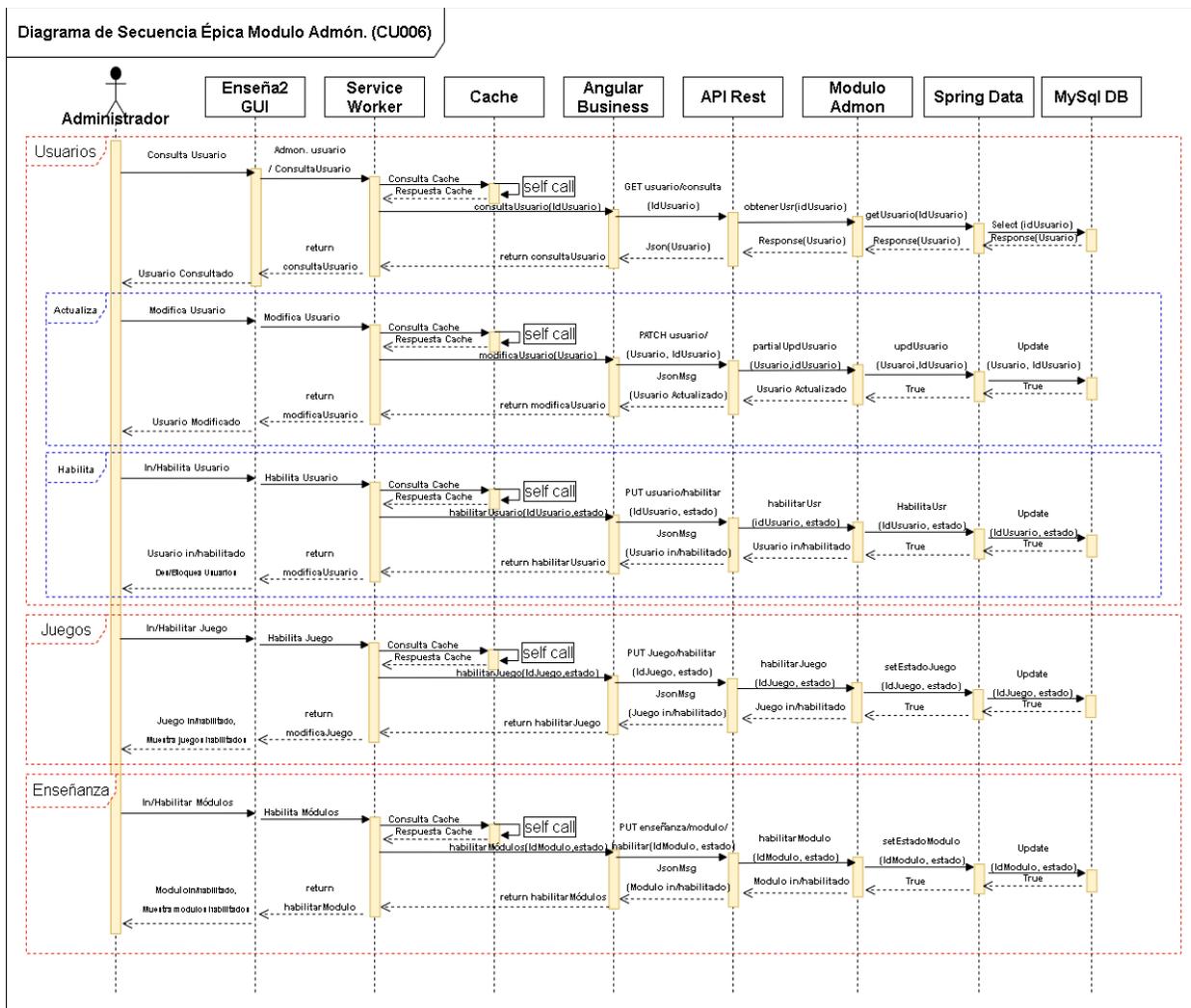


Figura 27. Diagrama de Secuencia Mod. Admón. (Fuente: Elaboración propia.)

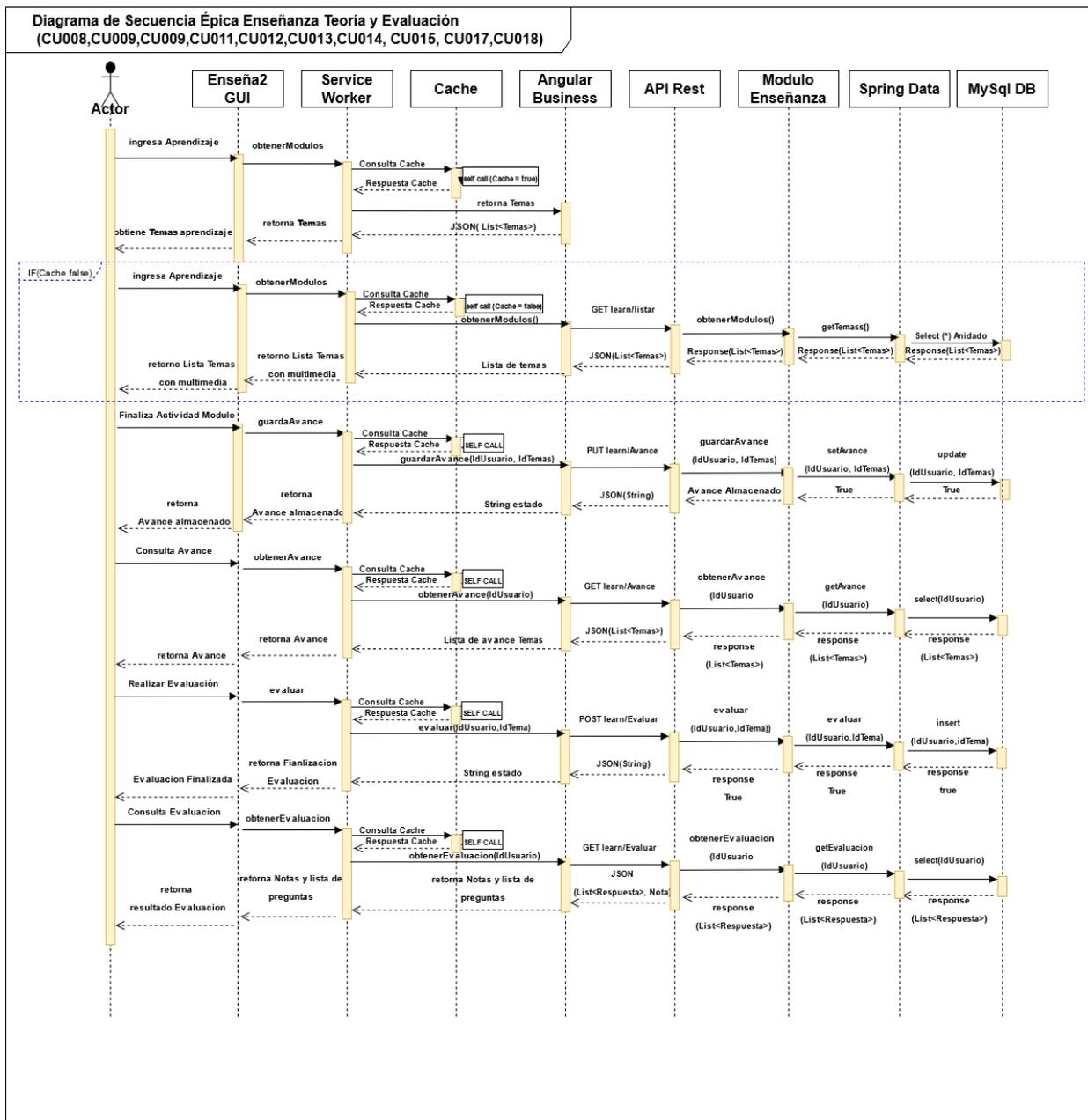


Figura 28. Diagrama de Secuencia - Épica Enseñanza (Fuente: Elaboración propia)

7.2.4. Diagrama de Clases

Para representar gráficamente la estructura general del sistema a nivel de clases, se presenta un diagrama general de las clases que componen el aplicativo Enseña2, en donde podremos observar cada una de estas clases con sus estados y comportamiento, adicional podremos observar las interacciones entre ellas, mediante una previa revisión a través del análisis del modelo entidad relación propuesto (ver Anexo 1), tratando de cumplir con los principios de alta cohesión, bajo acoplamiento, encapsulamiento, polimorfismo, abstracción y herencia, facilitando el análisis y diseño del sistema desarrollado, a continuación, presentamos el diagrama de clases general:

7.2.5. Arquitectura de Alto Nivel

En este apartado encontraremos la arquitectura a alto nivel definida en el diseño para la aplicación progresiva Enseña2, la cual consta de un diseño de aplicación basado en el estilo arquitectural MVC o Modelo, Vista y Controlador (ver Figura 28), distribuido en componentes (ver Figura 29), mediante el uso de una capa de presentación, una capa de negocio y una capa de datos.

De acuerdo a lo anterior, la capa de presentación está compuesta por un aplicativo elaborado en Angular más un componente Service Worker; la capa de negocio está constituida por un módulo Api Rest y un módulo de negocio, el cual a su vez es contenedor de la lógica funcional del aplicativo mediante la distribución en los sub módulos de juegos, administración y enseñanza, desarrollados bajo el framework Spring Boot; Finalmente, la capa de datos la cual alberga la conexión a una Base de Datos MySQL, mediante el uso de Spring Data JPA.

El diagrama de arquitectura general representa una visión general de los sistemas y su respectiva interacción y distribución a nivel arquitectónico, a continuación, veremos el diagrama de arquitectura general:

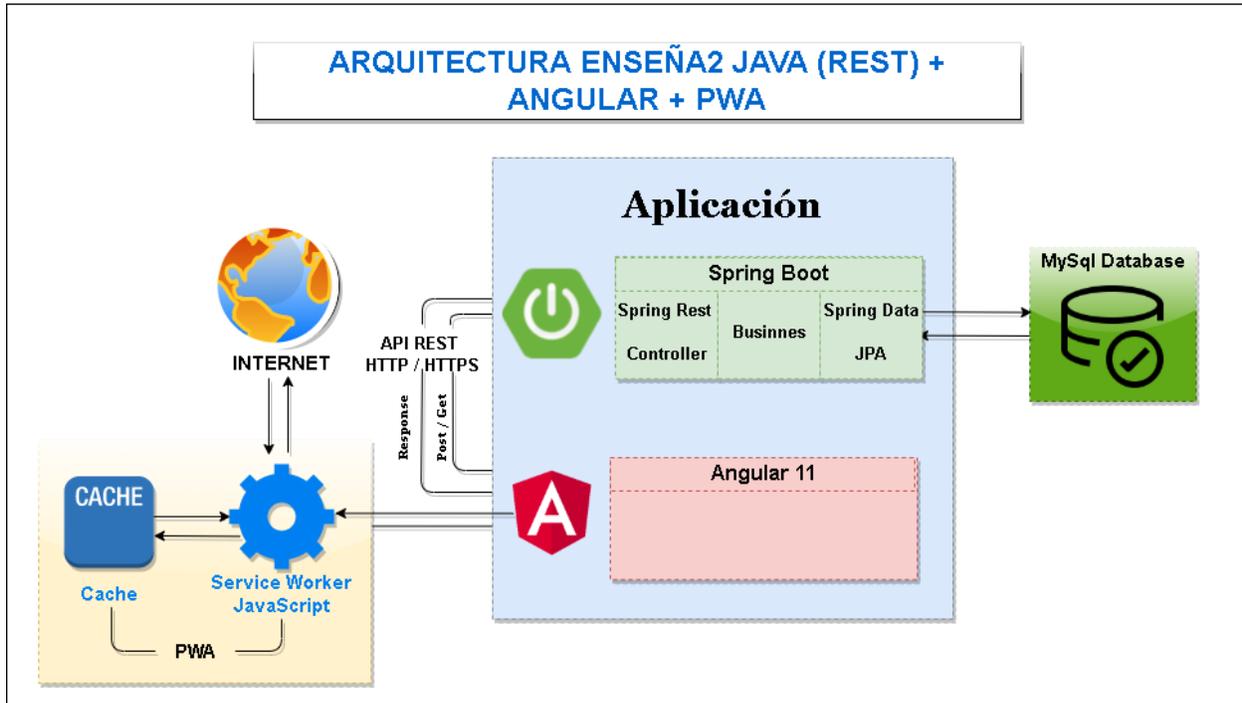


Figura 30. Diagrama de Arquitectura Alto Nivel (Fuente: Elaboración propia.)

El diagrama de componentes representa una visión general de los componentes del sistema y su distribución por capas, a continuación, veremos el diagrama de componentes general:

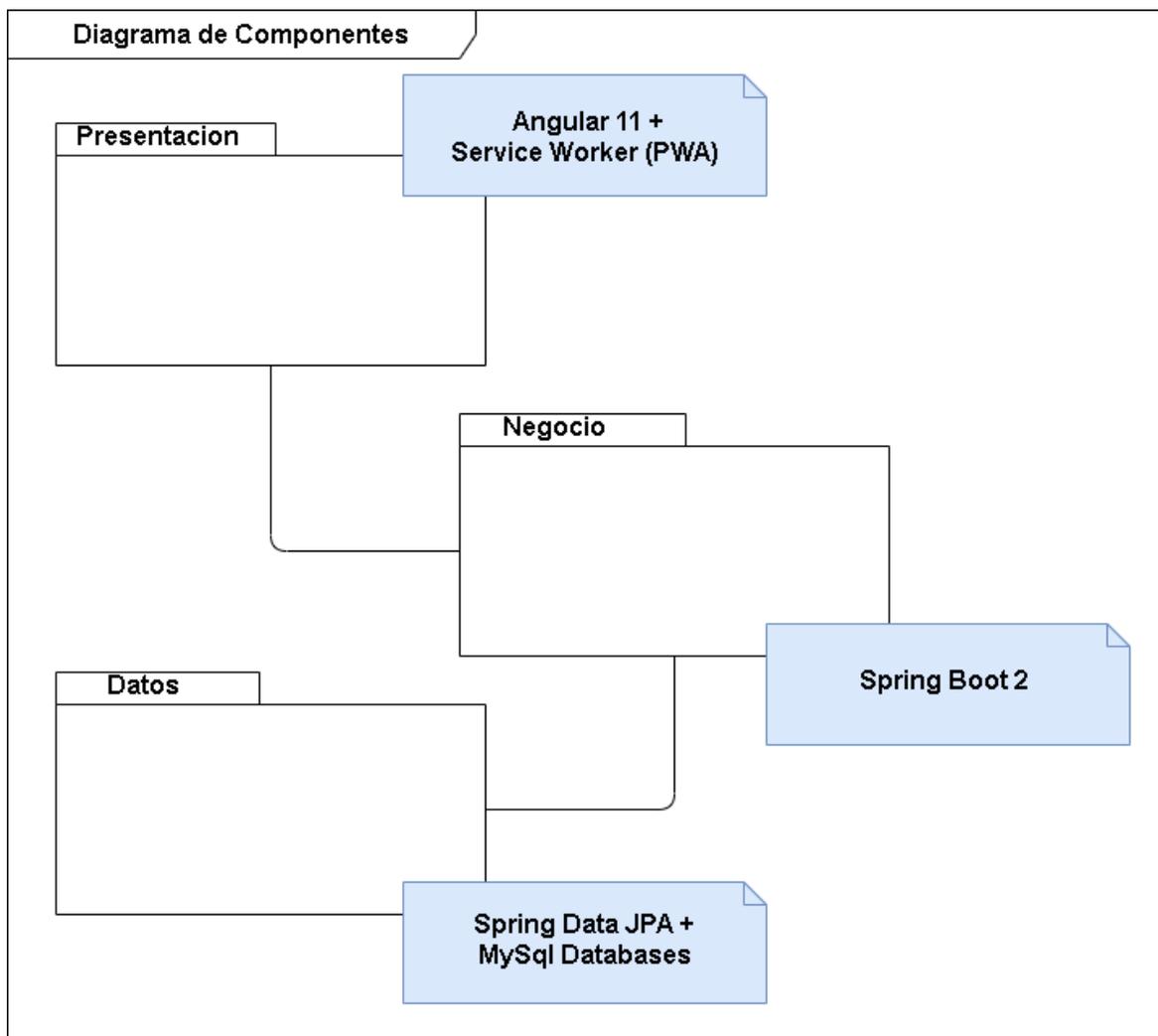


Figura 31. Diagrama de Componentes Alto Nivel (Fuente: Elaboración propia.)

El diagrama de componentes de primer nivel representa una visión general de los módulos contenidos dentro de los componentes previamente expuestos en la figura (Figura 29), y sus respectivas interfaz ofrecida e interfaz requerida, a continuación, veremos el diagrama de componentes de primer nivel:

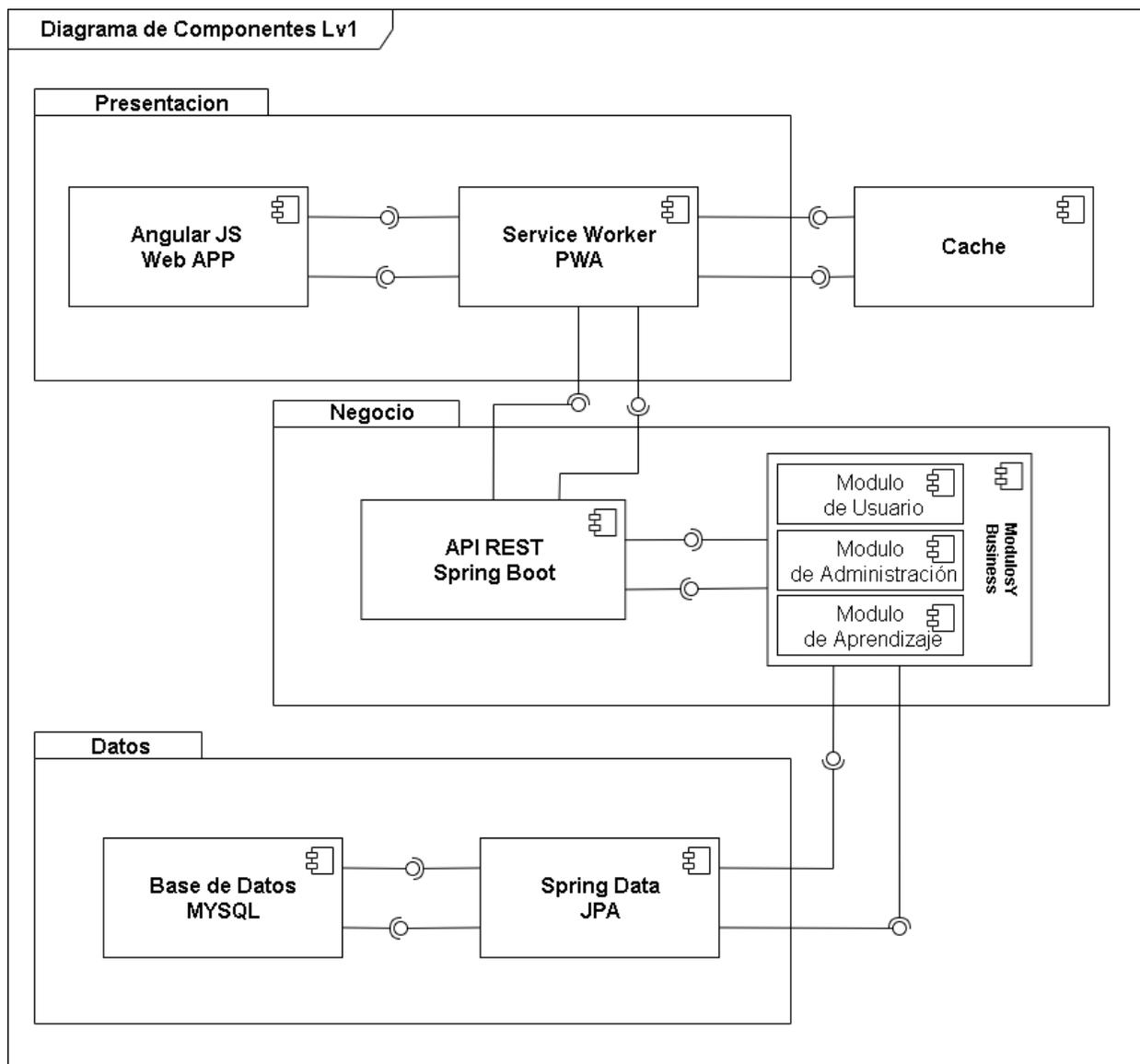


Figura 32. Diagrama de Componentes Nivel 1 (Fuente: Elaboración propia.)

7.3. Construcción

En este apartado podemos observar a manera de alto nivel, como se realizó la construcción y/o desarrollo del prototipo de la aplicación web progresiva denominada Enseña2, la instalación de los ambientes, y la configuración de estos. El prototipo Enseña2 consta de tres componentes para su correcto funcionamiento; siendo el primer componente la adecuación, configuración y todo lo relacionado con la base de datos, que para este ejercicio se utilizó el motor de base de datos MySQL en su versión 8.0.23.

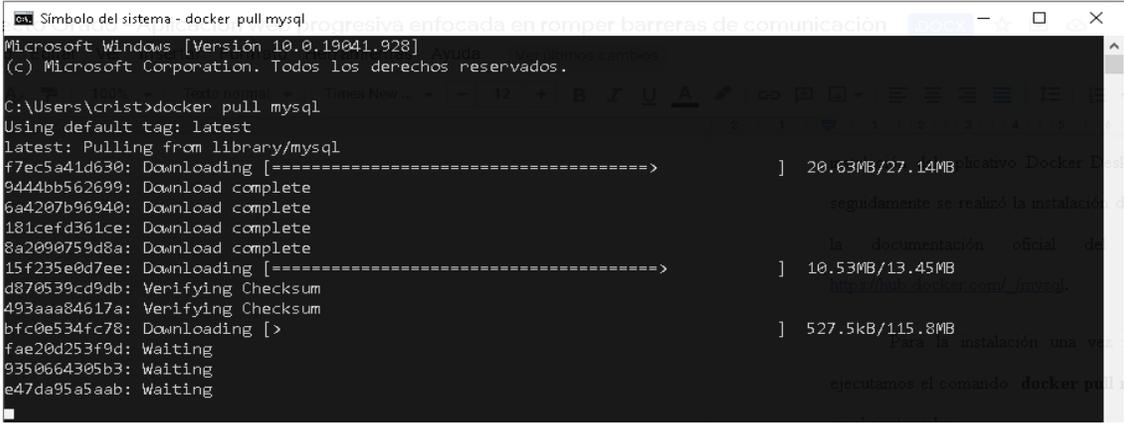
Como segundo componente tenemos un prototipo de aplicativo Backend el cual se encuentra desarrollado bajo la tecnología Java, implementando el JDK 15.0.1, en donde utilizaremos la versión 11 para el desarrollo de este aplicativo, como framework de desarrollo se implementó Spring Boot en su versión 2.4.3, bajo la tecnología de empaquetamiento Maven.

Como tercer componente tenemos el aplicativo FrontEnd el cual se compone de un prototipo de aplicación elaborada en TypeScript en su versión 4.1.5, y como framework de desarrollo se utilizó, Angular en la versión 11.2.7, Adicionalmente se utilizó para el componente de FrontEnd la implementación de un service worker bajo la tecnología PWA para hacer nuestro prototipo una aplicación progresiva.

7.3.1. Construcción componente de Bases de Datos

Para la implementación del ambiente de desarrollo de la base de datos en MySQL, se utilizó el contenedor oficial de MySQL 8.0.23 en Docker con su versión 20.10.5, el cual fue accedido bajo la instalación del aplicativo Docker Desktop, el cual se descargó de la página web www.docker.com, seguidamente se realizó la instalación del contenedor de MySQL utilizando las instrucciones obtenidas de la documentación oficial del contenedor de MySQL ubicada en la dirección https://hub.docker.com/_/mysql.

Para la instalación una vez instalado el software Docker desktop, a través de consola ejecutamos el comando **Docker pull MySQL**, este comando nos hace una descarga de la imagen oficial en el contenedor, de acuerdo con la siguiente imagen:

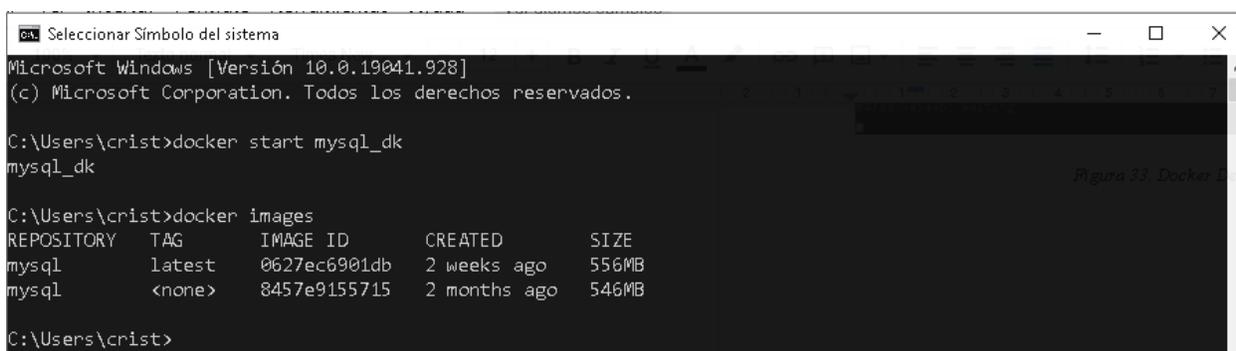
A screenshot of a Windows command prompt window titled "Símbolo del sistema - docker pull mysql". The window shows the output of the command "docker pull mysql". The output indicates that the latest version of the MySQL image is being pulled from the Docker library. The download progress is shown for several layers, with the total size of the image being 20.63MB. The download is complete, and the checksum is verified. The command prompt shows the user's current directory as "C:\Users\crist" and the command "docker pull mysql" is executed. The output shows the following steps: "Using default tag: latest", "latest: Pulling from library/mysql", "f7ec5a41d630: Downloading [=====] 20.63MB/27.14MB", "9444bb562699: Download complete", "6a4207b96940: Download complete", "181cefd361ce: Download complete", "8a2090759d8a: Download complete", "15f235e0d7ee: Downloading [=====] 10.53MB/13.45MB", "d870539cd9db: Verifying Checksum", "493aaa84617a: Verifying Checksum", "bfc0e534fc78: Downloading [>] 527.5kB/115.8MB", "fae20d253f9d: Waiting", "9350664305b3: Waiting", "e47da95a5aab: Waiting".

```
Símbolo del sistema - docker pull mysql
Microsoft Windows [Versión 10.0.19041.928]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\crist>docker pull mysql
Using default tag: latest
latest: Pulling from library/mysql
f7ec5a41d630: Downloading [=====] 20.63MB/27.14MB
9444bb562699: Download complete
6a4207b96940: Download complete
181cefd361ce: Download complete
8a2090759d8a: Download complete
15f235e0d7ee: Downloading [=====] 10.53MB/13.45MB
d870539cd9db: Verifying Checksum
493aaa84617a: Verifying Checksum
bfc0e534fc78: Downloading [>] 527.5kB/115.8MB
fae20d253f9d: Waiting
9350664305b3: Waiting
e47da95a5aab: Waiting
```

Figura 33. Pull imagen Docker MySQL (Fuente: Elaboración Propia.)

Una vez realizada la descarga validamos que la imagen se encuentre listada dentro de las imágenes para uso en los contenedores mediante el uso del comando Docker images de acuerdo con la siguiente imagen:



```

Microsoft Windows [Versión 10.0.19041.928]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\cris>docker start mysql_dk
mysql_dk

C:\Users\cris>docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
mysql         latest   0627ec6901db   2 weeks ago   556MB
mysql         <none>   8457e9155715   2 months ago  546MB

C:\Users\cris>

```

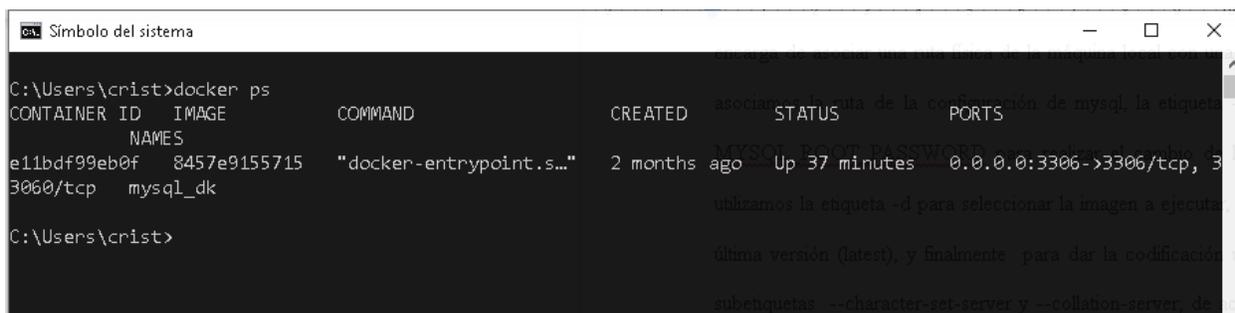
Figura 34. Listado de imágenes Docker (Fuente: Elaboración Propia.)

Posteriormente creamos la instancia del contenedor de MySQL en Docker utilizando una sentencia de configuración construida con el comando Docker run, el cual es utilizado para la ejecución de la imagen del contenedor, la etiqueta -p 3306:3306 es la configuración del puerto en donde el primer valor es el puerto local y el segundo valor es el puerto de la máquina virtual, la etiqueta --name es el apodo que se le va a dar al contenedor de la imagen para evitar tratar con un nombre hash, la etiqueta -v, se encarga de asociar una ruta física de la máquina local con una ruta de la máquina virtual, en este caso asociamos la ruta de la configuración de MySQL, la etiqueta -e le utilizamos conjunto con la palabra MYSQL_ROOT_PASSWORD para realizar el cambio de la contraseña del usuario, seguidamente utilizamos la etiqueta -d para seleccionar la imagen a ejecutar como demonio, en nuestro caso dejamos por defecto la última versión (latest), y finalmente para dar la codificación utf8 por defecto mediante el uso de las subetiquetas --character-set-server y --collation-server; de acuerdo a esto esta sentencia se encarga de configurar

el ambiente del servidor de base de datos, en la máquina local hospedadora, enlazando esta con la máquina virtual y sus carpetas:

```
Docker run -p 3306:3306 --name mysql_dk -v  
C:\Users\cris\OneDrive\Documentos\Docker:/var/lib/mysql -e  
MYSQL_ROOT_PASSWORD= **** -d mysql:latest --character-set-server=utf8 --  
collation-server=utf8_unicode_ci
```

Una vez ejecutado el comando anterior para validar que nuestro contenedor de MySQL se encuentra bien configurado ejecutamos el comando Docker ps, en el cual podemos ver las máquinas que se encuentran configuradas y su respectiva configuración básica



```

C:\Users\cris>docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS
e11bdf99eb0f   8457e9155715                        "docker-entrypoint.s..." 2 months ago  Up 37 minutes  0.0.0.0:3306->3306/tcp, 33060/tcp
mysql_dk
C:\Users\cris>

```

Figura 35. Servicios Docker corriendo (Fuente: Elaboración Propia)

Seguido a esto dentro del contenedor de MySQL, levantamos como el servicio de MySQL como servicio por defecto utilizando el comando **Docker exec -it mysql_dk MySQL -uroot -p**, el cual ejecuta dentro de nuestra instancia de Docker llamada “mysql_dk” el servicio de MySQL, y ya con esto podemos iniciar la prueba de la base de datos.

```

C:\Users\cris...> docker exec -it mysql_dk mysql -uroot -p
3060/tcp  mysql_dk

C:\Users\cris...> docker exec -it mysql_dk mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.23 MySQL Community Server - GPL

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

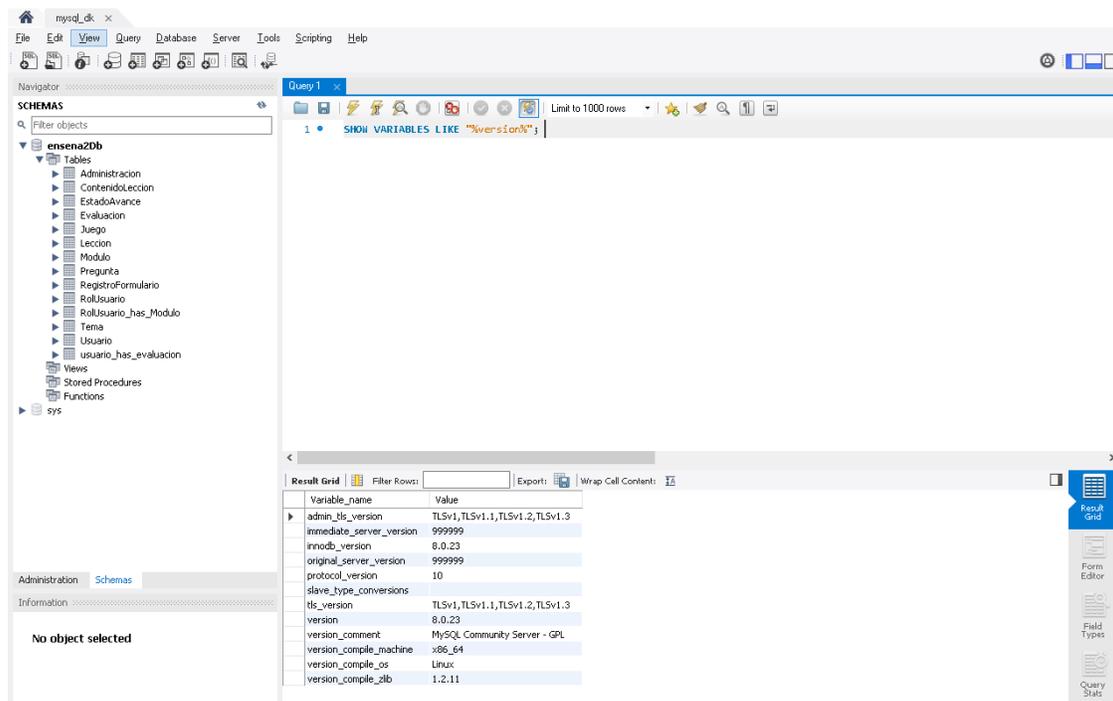
mysql> show databases
+-----+
| Database |
+-----+
| ensena2Db |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.00 sec)

mysql>

```

Figura 36. Prueba de conexión al servicio MySQL (Fuente: Elaboración Propia)

Para el desarrollo de la base de datos y sus tablas contenidas, se utilizó el software MySQL Workbench en su versión 8.0.23 conectado mediante interfaz gráfica a nuestro servidor de base de datos previamente creado, utilizando como hostname nuestro localhost en el puerto 127.0.0.1, el puerto 3306 el cual se encuentra emulado con la máquina virtual, y nuestros datos de usuario de MySQL, ya finalmente en este ambiente creado como servidor de base de datos, procedemos al desarrollo de nuestra base de datos denominada ensena2Db.



The screenshot displays the MySQL Workbench 8.0.23 interface. The left sidebar shows the 'SCHEMAS' tree with the 'ensena2db' database selected. The main window shows a query editor with the command 'SHOW VARIABLES LIKE "%version%"'. The 'Result Grid' at the bottom displays the output of this query, listing various system variables and their values.

| Variable_name | Value |
|--------------------------|-------------------------------|
| admin_tls_version | TLSv1,TLSv1.1,TLSv1.2,TLSv1.3 |
| admin_tls_version | TLSv1,TLSv1.1,TLSv1.2,TLSv1.3 |
| immediate_server_version | 999999 |
| innodb_version | 8.0.23 |
| original_server_version | 999999 |
| protocol_version | 10 |
| slave_type_conversions | |
| tls_version | TLSv1,TLSv1.1,TLSv1.2,TLSv1.3 |
| version | 8.0.23 |
| version_comment | MySQL Community Server - GPL |
| version_compile_machine | x86_64 |
| version_compile_os | Linux |
| version_compile_zlib | 1.2.11 |

Figura 37. Uso de Workbench 8.0.23 (Fuente: Elaboración Propia)

7.3.2. Construcción componente Backend

Para la adecuada configuración del ambiente de desarrollo del Backend en java se implementó el JDK 15.0.1, utilizando la versión 11 de Java, en donde se utilizó como IDE de desarrollo, IntelliJ IDEA 2021.1.1 (Ultimate Edition), el cual se manejó bajo una licencia de tipo educativa generada por Jet Brains, quienes son la casa propietaria de la distribución de este IDE, el desarrollo del Backend se compone de una aplicación MVC en spring boot, trabajando con el framework version 2.4.3 conjunto a la tecnología de empaquetamiento Maven y sus respectivas configuraciones a través del archivo pom.xml.

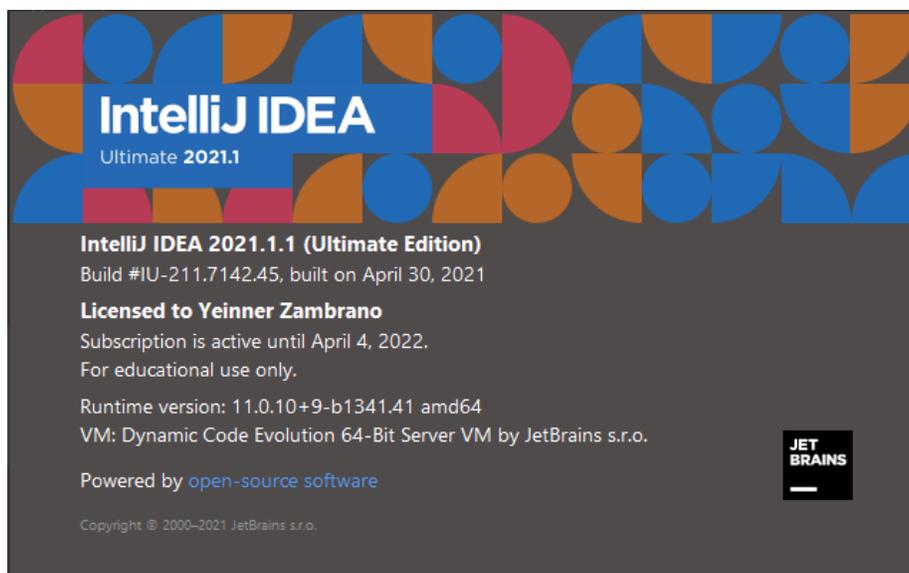


Figura 38. IDE IntelliJ IDEA (Fuente: Elaboración Propia)

La construcción del Backend fue realizada e idealizada teniendo en cuenta el uso del patrón de diseño arquitectónico MVC (Modelo Vista Controlador). Para el desarrollo de este fue necesario redistribuir las responsabilidades de los módulos planteados en el diseño mediante el

código a través de la exposición de servicios webs mediante la configuración de controladores que exponen servicios de tipo REST o la transferencia de estado representacional a través de los métodos GET, POST, PATCH, PUT y DELETE del protocolo HTTP/HTTPS, según sea el caso del servicio a orquestar.

La arquitectura y empaquetamiento de las operaciones desarrolladas en el servicio `enseña2` Spring Boot App, se componen de un paquete denominado `utils`, en el cual se establecen las clases que tienen función de utilidades y constantes dentro del aplicativo, encontramos el paquete `entity`, el cual alberga las clases entidad que persisten mediante el uso de las tecnologías JPA e Hibernate a la base de datos elaborada en MySQL, el paquete `repository` contiene las interfaces de las clases `entity` que extienden de la clase `CrudRepository` para poder generar persistencia en base de datos, el paquete `models`, contiene las clases de tipo `request` y `response` de las operaciones que se van a exponer mediante el uso de los controladores, el paquete `security`, contiene la clases de configuración de Spring Security y la configuración e implementación de JWT, el paquete `service` compone las clases de servicio, las cuales manejan toda la lógica del negocio, el paquete `controller`, contiene las clases de tipo `RestController`, en las cuales sus métodos están encargados del nombramiento y configuración de las operaciones que expone el servicio y finalmente el paquete `proxy`, está compuesto por interfaces que cumplen la función de `proxy delegate`, evitando la conexión directa de los controladores, la capa de servicios y la persistencia, esta configuración anteriormente nombrada la podemos observar en la siguiente imagen:

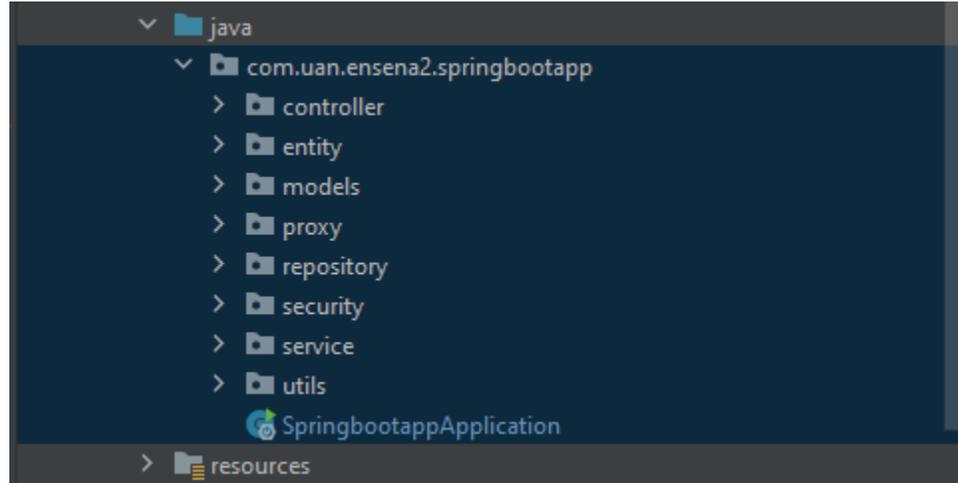


Figura 39. Empaquetamiento Enseñados App Backend (Fuente: Elaboración Propia)

De acuerdo con lo anteriormente mencionado, A continuación, se especifica que contiene cada uno de los package que están dentro del código fuente del Backend:

- *Controller*: Contiene los controladores que reciben las peticiones REST a través de los métodos GET, POST, PATCH, PUT y DELETE.
- *Entity*: En este package se encuentran todas las entidades que representan interés para el sistema desarrollado y sobre los cuales se realiza la persistencia con la base de datos.
- *Models*: Dentro de este se encuentran aquellas clases usadas para representar las solicitudes y las respuestas que reciben los controladores.
- *Proxy*: Dentro de este package están las clases que sirven como intermediarios en las peticiones que reciben los controladores hacia los servicios.
- *Repository*: Contiene las clases encargadas de gestionar todas las acciones de persistencia de cara a una tabla de la base de datos.

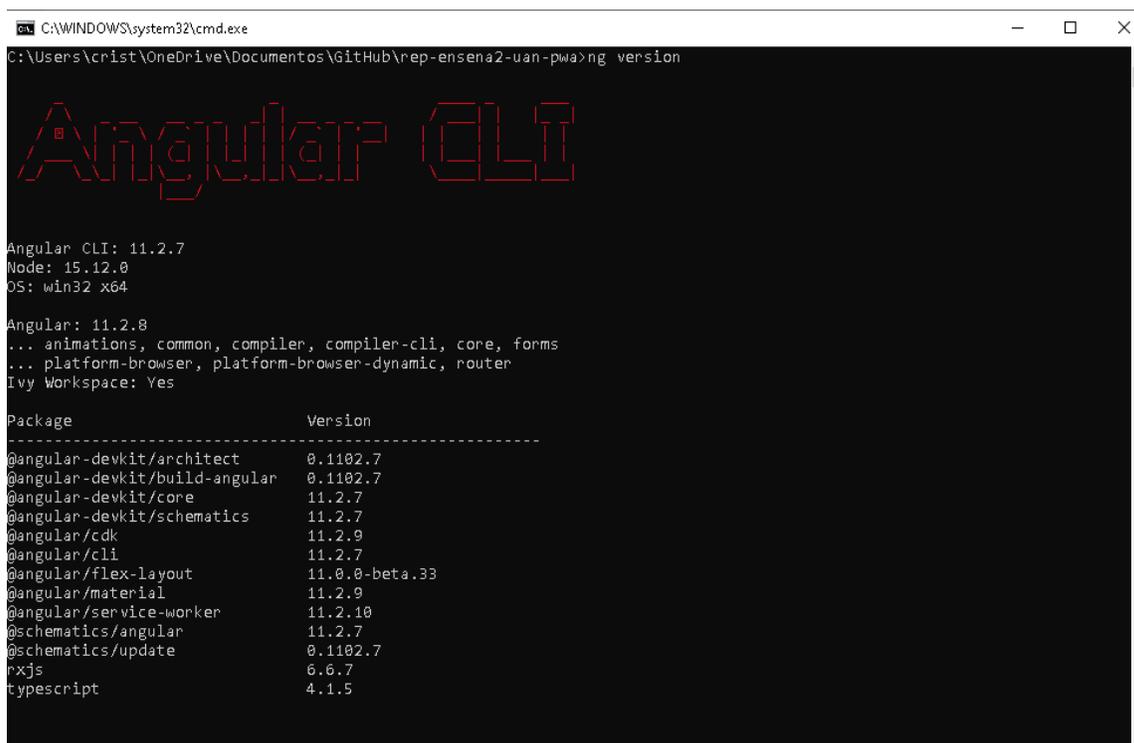
- *Security*: En este package se encuentran las clases que proveen seguridad en el consumo de los servicios del sistema.
- *Service*: Dentro de este se encuentran todas las clases que presentan los métodos los cuales reciben las solicitudes que envían los controladores y responde a la vista.
- *Utils*: en este package se encuentran las clases que representan todas las utilidades necesarias para la correcta ejecución de los métodos que se encuentran en los servicios.

La ejecución de las operaciones, constan del siguiente flujo de acuerdo al uso de las clases en el aplicativo enseñá2 App Backend, el cual inicia cuando se realiza una petición al servicio mediante el uso del nombre de la operación, la cual se configura en la etiqueta del verbo de acuerdo a la funcionalidad que se va a realizar, este método, se encarga de realizar la conexión entre el patrón proxy y la clase de lógica o clase service, está a su vez se encarga de realizar la conexión con las interfaces que extienden de CrudRepository y persisten mediante el uso de las clases de entidad.

El paradigma usado fue el de Programación Orientada a Objetos (POO) creando diversas clases. Dichas clases fueron distribuidas en distintos package de acuerdo con la responsabilidad y ejecución que estos realizan.

7.3.3. Construcción componente FrontEnd

En el desarrollo del componente FrontEnd, se realizó un prototipo aplicación web progresiva o PWA, la cual tiene como componente base, el desarrollo de un aplicativo en Angular 11, exactamente en la versión 11.2.7, acompañado de la versión 15.12.0 de node. Para el correcto uso de esta tecnología se hizo uso de paquetes externos asociados al framework angular para dar pie a el objetivo del proyecto, utilizando librerías como @angular/flex-layout en su versión 11.0.0, la cual tiene como función ayudar en el ajuste de los gaps en los layouts de las páginas y apoyar el correcto funcionamiento de la parte responsiva del aplicativo, se da uso de la librería @angular/material en su versión 11.2.9, la cual tiene componentes predefinidos en el framework, evitando así la reconstrucción de algunos componentes, simplemente importando su código se puede acceder a estos componentes visuales y sus funcionalidades, y el uso de la librería @angular/service-worker en su versión 11.2.10, librería utilizada para dar transformación a un aplicativo angular a manera de una aplicación progresiva o PWA, permitiendo el uso de service workers, el manejo del cache y la instalación de esta aplicación como nativa en cualquier dispositivo desde donde esta sea accedida, es de recordar que angular 11 es un framework de la tecnología typescript la cual es usada en su versión 4.1.5 para el desarrollo de este aplicativo.



```

C:\WINDOWS\system32\cmd.exe
C:\Users\crist\OneDrive\Documents\GitHub\rep-ensena2-uan-pwa>ng version

Angular CLI
Angular CLI: 11.2.7
Node: 15.12.0
OS: win32 x64

Angular: 11.2.8
... animations, common, compiler, compiler-cli, core, forms
... platform-browser, platform-browser-dynamic, router
Ivy Workspace: Yes

Package                    Version
-----
@angular-devkit/architect 0.1102.7
@angular-devkit/build-angular 0.1102.7
@angular-devkit/core       11.2.7
@angular-devkit/schematics 11.2.7
@angular/cdk               11.2.9
@angular/cli               11.2.7
@angular/flex-layout       11.0.0-beta.33
@angular/material          11.2.9
@angular/service-worker    11.2.10
@schematics/angular        11.2.7
@schematics/update         0.1102.7
rxjs                      6.6.7
typescript                 4.1.5

```

Figura 40. Versiones de angular y sus librerías (Fuente: Elaboración Propia)

Acompañado del lenguaje angular, se dio uso a tecnologías de estilo CSS como hoja de estilos mediante el uso del framework Bootstrap en su versión 4.6.0, permitiendo que nuestra aplicación sea responsiva a través del uso del sistema de grillas, también se da uso de la tecnología basada en JavaScript JQuery en su versión 3.5.1 y la tecnología HTML.

A nivel de distribución de paquetes la aplicación FrontEnd se encuentra contenida en un proyecto angular el cual en su raíz contiene los archivos de configuración, dentro de este proyecto podemos encontrar el directorio de recursos denominado src el cual en su raíz contiene los archivos de inicialización del aplicativo FrontEnd, como lo son el favicon.ico que corresponde al icono de la página web, el archivo index.html el cual inicializa los componentes a través de HTML, , el

archivo main.ts el cual inicializa el componente de configuración de Bootstrap en typescript, el archivo manifests.webmanifest, el cual maneja a configuración visual y de iconos del sitio para convertirlo en una aplicación progresiva , el archivo polyfills.ts el cual funciona como un archivo de importación de angular, el archivo styles.css que compone el estilo general de la aplicación y finalmente el test.ts el cual contiene funciones de prueba para el aplicativo.

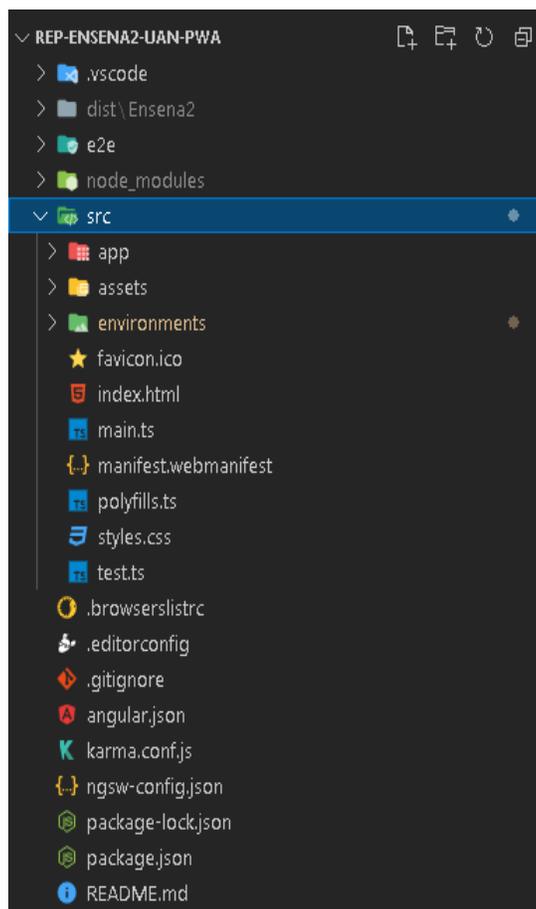


Figura 41. Configuración del proyecto Enseña2 FrontEnd (Fuente: Elaboración Propia)

De acuerdo con lo anteriormente mencionado, A continuación, se especifica que contiene cada uno de los package que están dentro del código fuente del FrontEnd en la ruta src, iniciando con el paquete denominado como app, el cual en su interior se compone de la siguiente estructura:

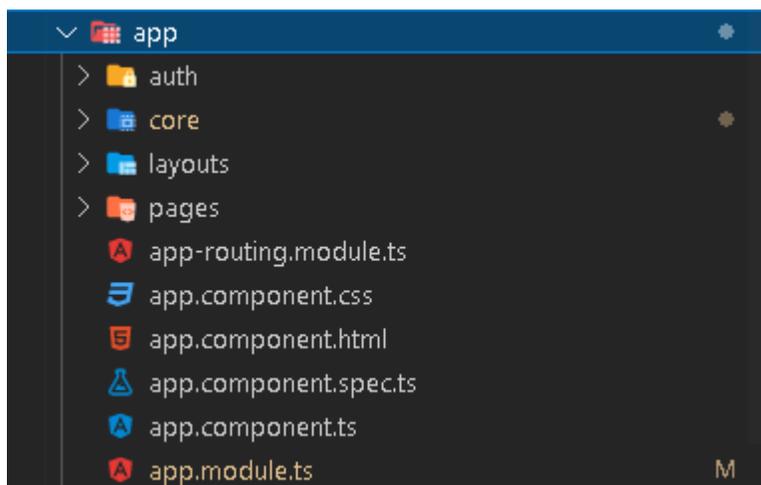


Figura 42. Paquetes app Enseña2 FrontEnd (Fuente: Elaboración Propia)

- *auth*: En este package se encuentran todas los módulos y componentes que se encargan de la autenticación y el registro del aplicativo, con sus respectivas hojas de estilo y archivos HTML.
- *core*: En este paquete podemos encontrar el sub paquete guards, en el cual se alojan los archivos .guard de la aplicación que son los encargados del acceso y la seguridad en la paginación, encontramos el paquete models, en el cual se alojan los archivos .models o entidades representadas en clases typescript y encontramos el paquete service en donde encontramos alojados los archivos .service, los cuales están encargados del consumo de las Apis del Backend y su respectivas funcionalidad de negocio.
- *layouts*: En este paquete podemos encontrar los componentes y módulos generales del aplicativo, como lo es el navbar, el footer y demás complementos que son transversales en todas las vistas del sistema.

- *pages*: En este paquete podemos encontrar los componentes y los modelos de las páginas del aplicativo, estos componentes, archivos HTML y hojas de estilo se encuentran a diferencia de los layouts de manera particular, es decir tienen su propia configuración de acuerdo con su funcionalidad.

El paquete complementario denominado *assets*, contiene en su interior los archivos multimedia del aplicativo FrontEnd, archivos tales como lo son las fuentes, los iconos para cada una de sus representaciones a nivel responsivo y progresivo, las imágenes etc, mientras que el paquete denominado *environments*, contiene dos archivos de configuración de los ambientes uno para pruebas denominado *environment.ts* el cual tiene la configuración del endpoint de pruebas y el archivo de producción denominado *environment.prod.ts*, el cual contiene la configuración del entorno productivo, de acuerdo a la siguiente imagen:

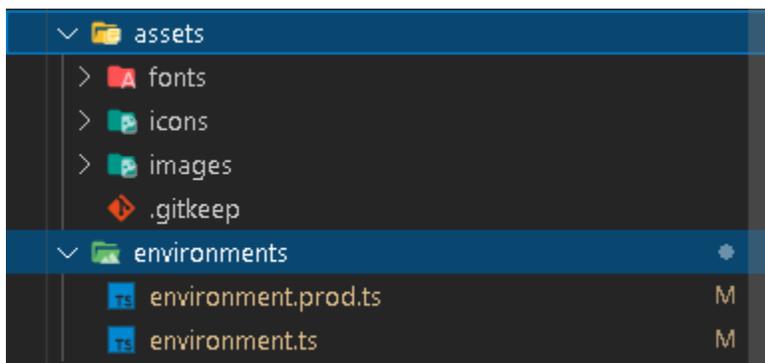


Figura 43. Paquetes Enseña2 FrontEnd (Fuente: Elaboración Propia)

7.4. Pruebas

7.4.1. Pruebas Funcionales

Con el fin de garantizar que el sistema desarrollado haya cumplido con los criterios de aceptación definidos dentro de cada una de las historias de usuarios se realizaron diversas pruebas partiendo desde dos frentes distintos: el Backend y el FrontEnd. Dichas Historias de Usuarios pueden ser consultadas en el apartado 7.1. del presente documento. Estas pruebas se realizaron tomando en cuenta diversos escenarios mitigando con ello errores en la lógica del código fuente que puedan afectar la puesta en marcha del sistema.

Para el desarrollo de las pruebas del Backend se usó la aplicación Postman en su versión 8.3.0. el cual facilita el envío de peticiones con los métodos GET, POST, PATCH y DELETE del protocolo HTTP REST sin requerir el desarrollo de un cliente.



Figura 44. Logo Postman (Fuente: <https://getlogovector.com/postman-inc-logo-vector-svg/>)

Se diseñaron 2 colecciones dentro de postman los cuales contienen los parámetros o request en formato JSON. Estas peticiones fueron enviadas a los servicios desarrollados en el Backend para su correcta ejecución con el fin de garantizar el adecuado funcionamiento de cada uno de estos. A continuación, se presenta una imagen con las colecciones de request de pruebas diseñadas en Postman:

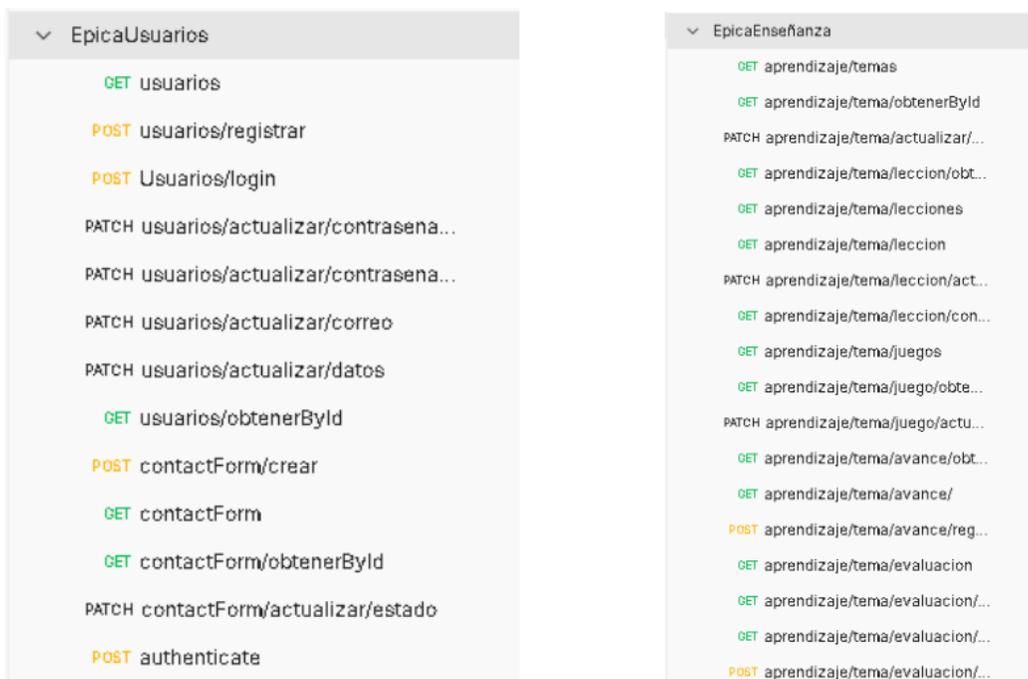


Figura 45. Colleions Postman (Fuente: Elaboración Propia)

Estos request tienen el tipo de método que requiere el controlador ya sea del tipo GET, POST, PATCH o DELETE, además, estas peticiones envían los parámetros o request a una URL previamente definida en cada una de las operaciones de los controladores desarrollados en el Backend. Cada uno de los request tienen definidos parámetros o un body con una rama en formato JSON el cual contiene los valores esperados por las operaciones definidas en los controladores.

A fin de validar que haya comunicación entre el Backend y la Base de datos, y con el objetivo de garantizar que los métodos desarrollados estén alineados con los criterios de aceptación, se realizaron diversas validaciones a través del consumo de los servicios por medio de las peticiones enviadas desde Postman. A continuación, se presentan como evidencia tres consumos de los servicios desarrollados en el Backend en las siguientes imágenes:

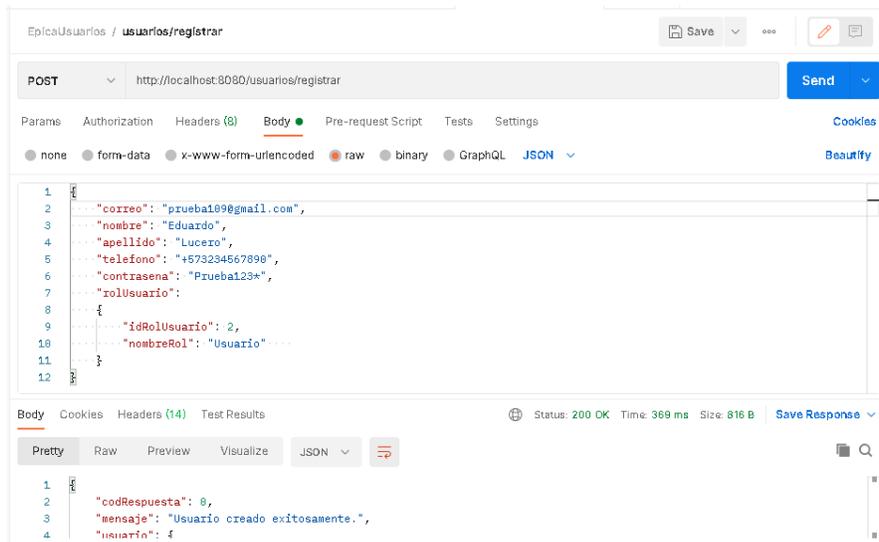


Figura 46. Consumo del servicio de Registro de Usuario (Fuente: Elaboración Propia)

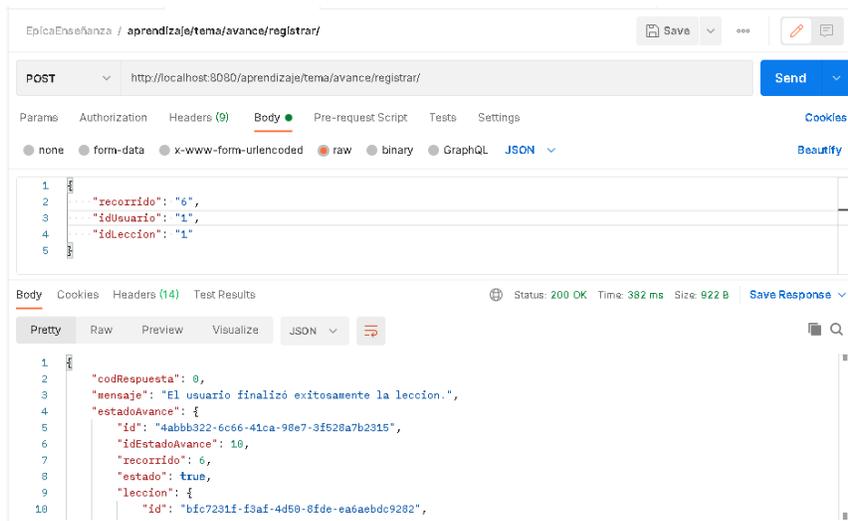


Figura 47. Consumo del servicio de Registro de Avance de Usuario (Fuente: Elaboración Propia)

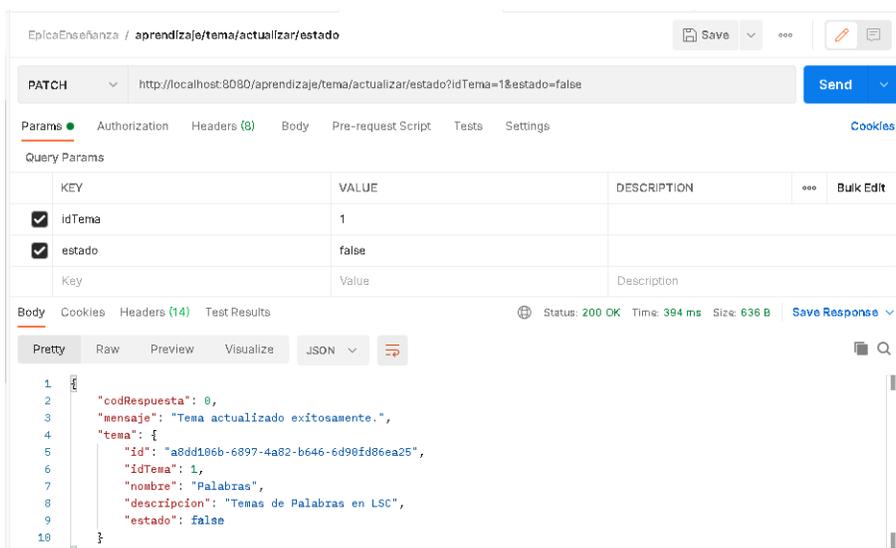
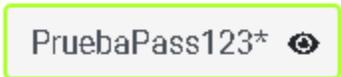


Figura 48. Consumo del servicio de Actualización de Estado de Tema (Fuente: Elaboración Propia)

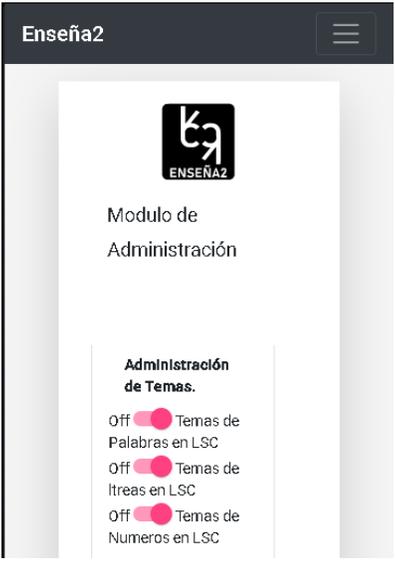
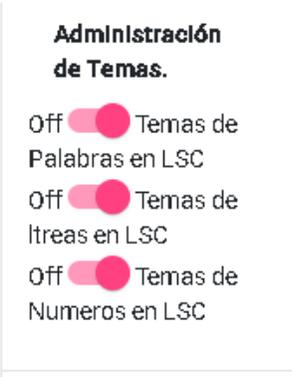
Como se observa en las imágenes anteriores, una vez se realiza el envío de las peticiones desde postman hacia el Backend, este procesa los datos de acuerdo con la operación a la cual va dirigida y realiza la consulta, registro o modificación en la Base de datos conforme a lo requerido para luego retornar un resultado que es enviado desde Backend en un response en formato JSON como respuesta a los datos recibidos.

Con el fin de garantizar que el sistema cumpla con los criterios de aceptación establecidos en las historias de usuarios se realizaron las validaciones respectivas en el FrontEnd por medio de un servidor Localhost el cual fue iniciado a través de la terminal de comando de la aplicación Visual Studio Code. A continuación, se presenta una tabla en la cual tomamos como referencia tres historias de usuarios, 1 por épica respectivamente, con el objetivo de mostrar los resultados de las pruebas realizadas sobre los criterios de aceptación más relevantes de dichas historias, con el fin de dar a conocer las validaciones funcionales del sistema:

Tabla 7. Pruebas Funcionales (Fuente: Elaboración Propia)

| Épica ID Historia | Criterios de Aceptación | Prueba Funcional |
|------------------------------|--|--|
| | 1. El sistema deberá ocultar la contraseña. |  <p>Figura 49. E1- HU-002-C1 (Fuente: Elaboración Propia)</p> |
| Épica de Usuario (HU-002) | 1. El campo Contraseña en el inicio de sesión deberá tener el botón de ver password el cual permita visualizar la contraseña digitada u ocultarla. |  <p>Figura 50. E1-HU-002-C2 (Fuente:Elaboración Propia)</p> |
| | 1. La nueva contraseña deberá tener mínimo 8 caracteres y máximo 12 caracteres. La contraseña no debe tener espacios y deberá solicitar al menos una letra en mayúscula, una letra en minúscula, 1 número y 1 carácter especial. |  <p>Figura 51. E1-HU-002-C3 (Fuente: Elaboración Propia)</p> |

| | |
|---|--|
| <p>1. Deberá haber otro campo de confirmación de contraseña en el cual se pueda verificar la contraseña registrada.</p> |  <p>Figura 52. HU-002-C4 (Fuente: Elaboración Propia)</p> |
| <p>1. El sistema deberá rechazar el cambio de la contraseña cuando esta sea similar a la anterior mostrando un mensaje de advertencia indicando que la contraseña debe ser una nueva y especificando las características que debe tener la misma.</p> |  <p>Figura 53. E1-HU-002-C5 (Fuente: Elaboración Propia)</p> |
| <p>1. El sistema deberá implementar el algoritmo de cifrado BCrypt de Spring Boot para el registro de la contraseña.</p> |  <p>Figura 54. E1-HU-002-C6 (Fuente: Elaboración Propia)</p> |

| | | |
|--|--|---|
| <p>Epica de Administración (HU-002)</p> | <p>1. Cuando el administrador ingrese a la opción Configurar Temas deberá visualizar las opciones Número, Letras y Palabras.</p> |  <p><i>Figura 55. E2-HU-002-C1 (Fuente: Elaboración Propia)</i></p> |
| | <p>1. Al lado de cada tema deberá haber un Switch button que permite al administrador habilitar o deshabilitar los temas.</p> |  <p><i>Figura 56. E2-HU-002-C2 (Fuente: Elaboración Propia)</i></p> |

| | | |
|---|---|---|
| | <p>1. El proceso de habilitación/deshabilitación de un tema no debe afectar el avance del aprendizaje de los usuarios y no debe tardar más de 1 segundo en efectuarse los cambios sobre el sistema y tampoco más de 2 minutos en ser aplicados en los dispositivos de los usuarios.</p> | <div data-bbox="987 296 1276 659" style="border: 1px solid gray; padding: 10px;"> <p style="text-align: center;">Administración de Temas.</p> <p>Off <input type="checkbox"/> Temas de Palabras en LSC</p> <p>Off <input type="checkbox"/> Temas de Letras en LSC</p> <p>Off <input type="checkbox"/> Temas de Numeros en LSC</p> </div> <p style="text-align: center;"><i>Figura 57. E2-HU-002-C4 (Fuente: Elaboración Propia)</i></p> |
| <p style="text-align: center;">Épica de Aprendizaje (HU-001)</p> | <p>1. Los temas disponibles en el menú de aprendizaje son: Números, Letras y Palabras.</p> | <div data-bbox="850 884 1414 1199" style="border: 1px solid gray; padding: 10px;"> <p style="text-align: center;">SELECCIONA UN TEMA:</p> <div style="display: flex; justify-content: space-around;"> <div style="border: 1px solid gray; padding: 5px; text-align: center;">  <p style="font-size: 8px;">Módulo de Aprendizaje de Palabras Es un módulo de aprendizaje de palabras de palabras en Colombia LSC.</p> </div> <div style="border: 1px solid gray; padding: 5px; text-align: center;">  <p style="font-size: 8px;">Módulo de Aprendizaje de Letras Es un módulo de aprendizaje de letras de letras en Colombia LSC.</p> </div> <div style="border: 1px solid gray; padding: 5px; text-align: center;">  <p style="font-size: 8px;">Módulo de Aprendizaje de Números Es un módulo de aprendizaje de números de números en Colombia LSC.</p> </div> </div> </div> <p style="text-align: center;"><i>Figura 58. E3-HU-001-C1 (Fuente: Elaboración Propia)</i></p> |

| | | |
|--|---|---|
| | <p>1. Cuando el usuario acceda a un tema, el sistema deberá desplegar el listado con las lecciones disponibles para dicho tema.</p> |  <p>Figura 59. E2-HU-001-C2 (Fuente:Elaboración Propia)</p> |
|--|---|---|

Los demás resultados de todas las pruebas funcionales de cara a los criterios de aceptación establecidos dentro de las historias de usuarios se encuentran debidamente documentados en el anexo “Lista de Chequeo de las Historias de Usuario” el cual es un archivo en Excel cargado en una ruta de drive y que se encuentra en el apartado 8.3 del presente documento.

7.4.2. Pruebas Lighthouse

Lighthouse es una herramienta automatizada de código abierto integrada en Google Chrome y que sirve para medir la calidad de las páginas web auditando el rendimiento, la accesibilidad y la optimización de motores de búsqueda de páginas web.

A continuación, se presentan el resultado de las pruebas realizadas del FrontEnd de la aplicación Enseña2:

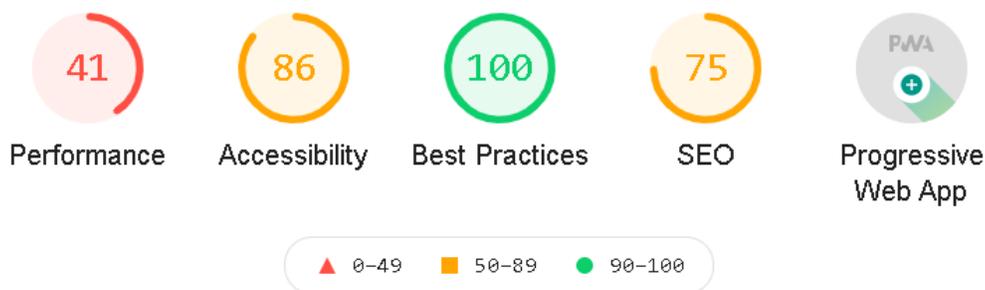


Figura 60. Resultados Análisis Lighthouse (Fuente: Elaboración Propia)

Los resultados anteriores indica lo siguiente:

- *Performance*: El Performance tiene una calificación baja debido al cargue de archivos multimedia que supera el peso recomendado para la aplicación, CSS y JavaScript no usados, esto debido a que la aplicación es un prototipo el cual aún se encuentra en fase de desarrollo.

- *Accessibility*: Estas comprobaciones destacan oportunidades para mejorar la accesibilidad de la aplicación web prototipo Enseña2. En el análisis realizado por Lighthouse solo se pudo detectar automáticamente un subconjunto de problemas de accesibilidad, por lo cual sugirió realizar pruebas manuales. Adicionalmente, señala que los colores de fondo y primer plano no tienen una relación de contraste suficiente afectando algunos textos de bajo contraste ya que pueden llegar a ser difíciles o casi imposible de leer para algunos usuarios.
- *Best Practice*: La aplicación web prototipo Enseña2 cuenta con muy buenas prácticas en el desarrollo.
- *SEO*: Estas comprobaciones garantizan que la aplicación web prototipo Enseña2 está optimizada para la clasificación de los resultados de los motores de búsqueda. Lighthouse sugiere incluir meta descriptions para que en los resultados de búsqueda se resumen de manera concisa el contenido de la aplicación.
- *Progressive Web App*: Estas comprobaciones validaron que la aplicación Enseña2 cumple con los aspectos de una aplicación web progresiva ya que cuenta con un manifiesto de la aplicación web y el service worker que son requisitos de instalación.

7.4.3. Pruebas Sonarlint

Para poder garantizar que el código desarrollado en el Backend es un código limpio y libre de incidencias y de code smells fue necesario realizar algunas pruebas de calidad usando la herramienta Sonarlint en su versión 4.15.0.30553. Esta herramienta es una extensión o plugin usada en diferentes IDE de desarrollo y que facilita encontrar posibles incidencias en el código de manera automática al mismo tiempo que se codifica de forma sencilla, incluso brindando una breve explicación y una posible solución a estas.



Figura 61. Logo SonarLint (Fuente: <https://www.sonarsource.com/products/sonarlint>.)

Al realizar el análisis del Backend del proyecto con la herramienta Sonarlint, se logró identificar que dentro de los 98 archivos existente en el proyecto hay un total de 169 mensajes de alerta en 34 archivos distintos cómo se observa en la siguiente imagen:

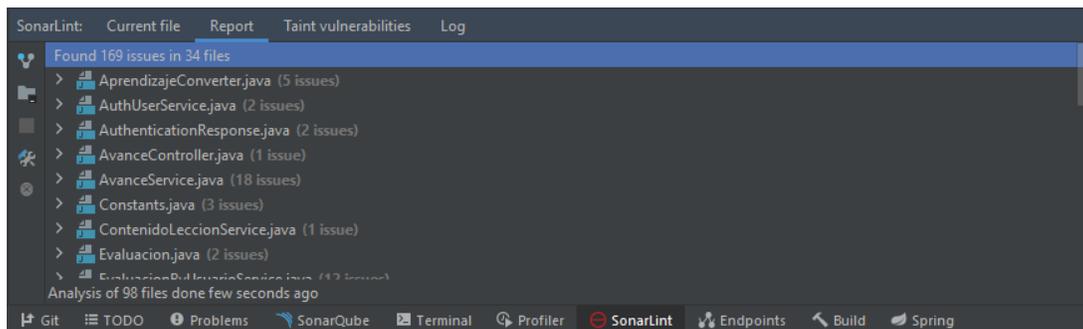


Figura 62. Análisis AS-IS SonarLint (Fuente: Elaboración Propia)

A continuación, se da a conocer una tabla que contiene el listado de las clases en las cuales la herramienta Sonarlint arrojó uno o varios mensajes de alerta:

Tabla 8. Análisis AS-IS Sonarlint (Fuente: Elaboración Propia)

| Clase | Cantidad de Mensajes de Alerta |
|----------------------------|--------------------------------|
| AprendizajeConverter | 5 |
| AuthUserService | 2 |
| AuthenticationResponse | 2 |
| AvanceController | 1 |
| AvanceService | 18 |
| Contants | 3 |
| ContenidoLeccionService | 1 |
| Evaluacion | 2 |
| EvaluacionByUsuarioService | 12 |
| EvaluacionService | 3 |
| FormService | 7 |
| GeneradorBcrypt | 4 |

| | |
|----------------------------|----|
| IRolUsuario_Has_Modulo | 1 |
| IRolUsuario_Has_Evaluacion | 1 |
| Juego | 2 |
| JuegoService | 5 |
| JwtTokenFilter | 3 |
| JwtTokenUtil | 6 |
| Leccion | 2 |
| LeccionService | 10 |
| Pregunta | 1 |
| PreguntaService | 3 |
| RolUsuario | 1 |
| RolUsuario_Has_Modulo | 2 |
| SecurityConfig | 6 |
| SpringBootApplication | 2 |
| Tema | 2 |
| TemaService | 7 |
| Usuario | 1 |
| UsuarioController | 4 |
| UsuarioConverter | 5 |
| UsuarioService | 29 |
| Usuario_Has_Evaluacion | 2 |
| Utilidades | 14 |

Partiendo de este análisis se logró realizar un proceso de depuración, validando cada una de las recomendaciones dadas por Sonarlint en donde se tuvo en cuenta el nivel de criticidad de dichos mensajes. Cabe destacar que no todas las indicaciones dadas por dicha herramienta deben ser atendidas ya que esto podría afectar significativamente la lógica del código y su adecuado funcionamiento. En conformidad a esta última afirmación, y al revisar detalladamente cada una de las alertas arrojadas por la herramienta, se pudo reducir el 58,58% del número inicial reportado, es decir, 99 mejoras realizadas al código, quedando un total de 70 mensajes de alertas en 24 archivos como se puede observar en la imagen siguiente:

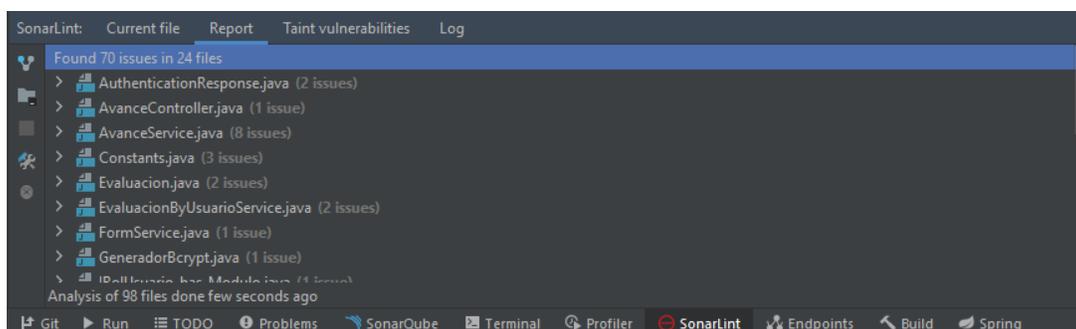


Figura 63. Análisis TO-BE Sonarlint (Fuente: Elaboración Propia)

En la siguiente tabla se brinda más detalle de cuáles son las clases en las que Sonarlint arrojó uno o varios mensajes de alerta posterior al proceso de limpieza de código realizado:

Tabla 9. Análisis TO-BE Sonarlint (Fuente: Elaboración Propia)

| Clase | Cantidad de Mensajes de Alerta |
|----------------------------|--------------------------------|
| AuthenticationResponse | 2 |
| AvanceController | 1 |
| AvanceService | 8 |
| Contants | 3 |
| Evaluación | 2 |
| EvaluacionByUsuarioService | 2 |
| FormService | 1 |
| GeneradorBcrypt | 1 |
| IRolUsuario_Has_Modulo | 1 |
| IRolUsuario_Has_Evaluacion | 1 |
| Juego | 1 |
| JwtTokenFilter | 3 |
| JwtTokenUtil | 3 |
| Lección | 1 |
| Pregunta | 1 |
| RolUsuario | 1 |
| RolUsuario_Has_Modulo | 2 |
| SecurityConfig | 2 |
| SpringBootApplication | 1 |
| Tema | 1 |
| UsuarioController | 4 |
| UsuarioService | 13 |
| Usuario_Has_Evaluacion | 2 |
| Utilidades | 13 |

A continuación, mostraremos un ejemplo de code smells identificado en el proceso de análisis del código con Sonarlint y cuál fue el resultado luego de refactorizar dicho código.

```
if(recorrido == contenidoUpd.size()){
    estado = true;
    mensaje = "El usuario finalizó exitosamente la leccion.";
    band = 1;
}else if(recorrido > contenidoUpd.size()){
    mensaje = "El recorrido no debe superar el limite del contenido de la lección.";
    band = 0;
}else if(recorrido < contenidoUpd.size()){
    estado = false;
    mensaje = "El usuario no ha finalizado exitosamente la leccion.";
    band = 1;
}
```

Figura 64. Code Smell Clase AvanceService (Fuente: Elaboración Propia)

La anterior imagen nos muestra algunas condicionales que se repetía varias veces en el código de la clase AvanceService. De acuerdo con esto, se procedió a extraer el código en algunos nuevos métodos y se reemplazó el código anterior por un llamado a estos.

```
band = recorridoValido(recorrido, contenido.size());  
mensaje = recorridoMensaje(recorrido, contenido.size());  
estado = recorridoEstado(recorrido, contenido.size());
```

```
private int recorridoValido(int recorrido, int contenidoSize){  
    if(recorrido <= contenidoSize){  
        return 1;  
    }else{  
        return 0;  
    }  
}  
  
private boolean recorridoEstado(int recorrido, int contenidoSize){  
    if(recorrido == contenidoSize){  
        return true;  
    }else{  
        return false;  
    }  
}  
  
private String recorridoMensaje(int recorrido, int contenidoSize){  
    String mensaje = "";  
    if(recorrido == contenidoSize){  
        mensaje = "El usuario finalizó exitosamente la lección.";  
    }else if(recorrido > contenidoSize){  
        mensaje = "El recorrido no debe superar el límite del contenido de la lección.";  
    }else if(recorrido < contenidoSize){  
        mensaje = "El usuario no ha finalizado exitosamente la lección.";  
    }  
    return mensaje;  
}
```

Figura 65. Código Refactorizado Clase AvanceService (Fuente: Elaboración Propia)

7.4.4. Pruebas Sonarcloud

En la verificación de pruebas tanto de calidad como seguridad, se realiza el análisis del código del Backend del aplicativo prototipo Enseña2, a través del uso de la herramienta Sonar Cloud, la cual es una plataforma de solución de calidad que detecta vulnerabilidades a nivel de código, tales como code smells, bugs o errores en proyectos privados como públicos, en el caso de del prototipo de aplicación progresiva Enseña2 se utilizó la versión gratuita para proyectos públicos, conectando nuestro repositorio GitHub con el código del Backend, con Sonar Cloud mediante el uso de un secret token y la creación de un archivo de configuración build en el proyecto con extensión .xml, el cual contiene la configuración para dar acceso a la acción de análisis de código mediante el uso de GitHub y finalmente exponiendo el resultado en Sonar Cloud.

El análisis del código del Backend del prototipo de aplicación progresiva Enseñados se realizó a un total de 2.827 Líneas de código distribuidas en un total de 98 archivos java compuesto por clases e interfaces de acuerdo con la siguiente tabla generada por la ejecución de la revisión de código generada por GitHub y expuesta por la plataforma Sonar Cloud.

| | Lines of Code | Bugs | Vulnerabilities | Code Smells | Security Hotspots | Coverage | Duplications |
|--|---------------|------|-----------------|-------------|-------------------|----------|--------------|
| main/java/com/uan/ensenaa2/springbootapp | 2,827 | 0 | 0 | 74 | 0 | 0.0% | 2.2% |
| controller | 362 | 0 | 0 | 5 | 0 | 0.0% | 0.0% |
| entity | 455 | 0 | 0 | 13 | 0 | 0.0% | 3.4% |
| models | 273 | 0 | 0 | 4 | 0 | 0.0% | 4.8% |
| proxy | 84 | 0 | 0 | 0 | 0 | — | 0.0% |
| repository | 132 | 0 | 0 | 2 | 0 | — | 0.0% |
| security | 120 | 0 | 0 | 5 | 0 | 0.0% | 0.0% |
| service | 1,206 | 0 | 0 | 25 | 0 | 0.0% | 2.7% |
| utils | 185 | 0 | 0 | 20 | 0 | 0.0% | 0.0% |
| SpringbootappApplication.java | 10 | 0 | 0 | 0 | 0 | 0.0% | 0.0% |

Figura 66. Estadísticas de Líneas de código Sonarcloud (Fuente: Elaboración Propia)

De acuerdo con el análisis ejecutado por Sonar Cloud se generó el primer informe de calidad del código del aplicativo Enseña2 (SpringBoot), el cual nos arrojó información del código tal como que se contaba con 2 bus, 19 Security HotSpots, 172 code smells, y 8 Bloques duplicados como se puede observar en la siguiente imagen:

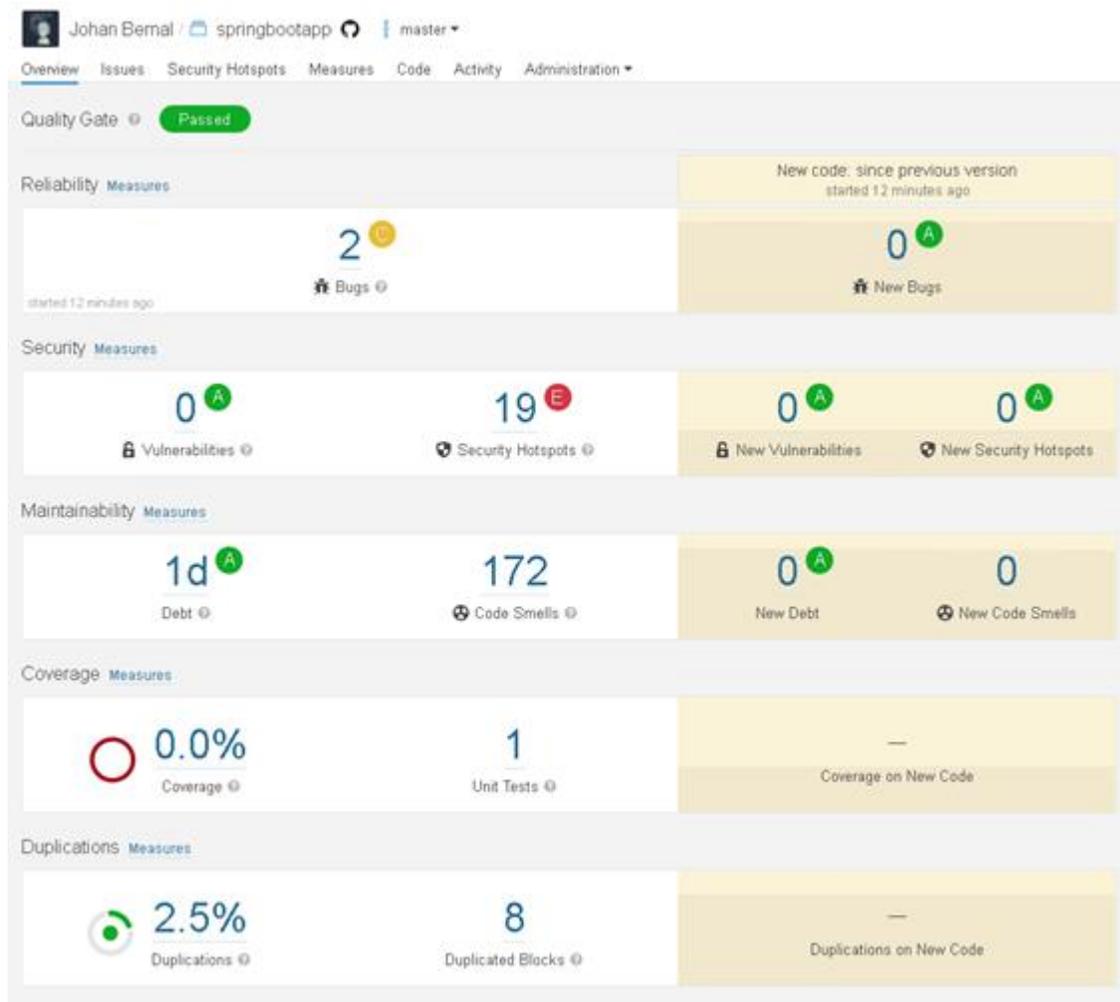


Figura 67. Overview Sonar Cloud primera ejecución (Fuente: Elaboración Propia)

De acuerdo con lo anterior, se realizó la tarea de dar solución a estas fallas presentadas a nivel de seguridad y calidad, mediante el uso de las ayudas proporcionadas tanto por Sonar Cloud, como también por Sonarlint, permitiendo así reducir principalmente el número de bugs a cero, el número de Security HotSpots a cero, reducir el número de code smells de 172 a 75, reducir el número de bloques duplicados de 8 a 6, permitiendo esto una reducción de duplicidad de 2.5% a 2.1%. de acuerdo con la siguiente imagen generada por Sonar Cloud:

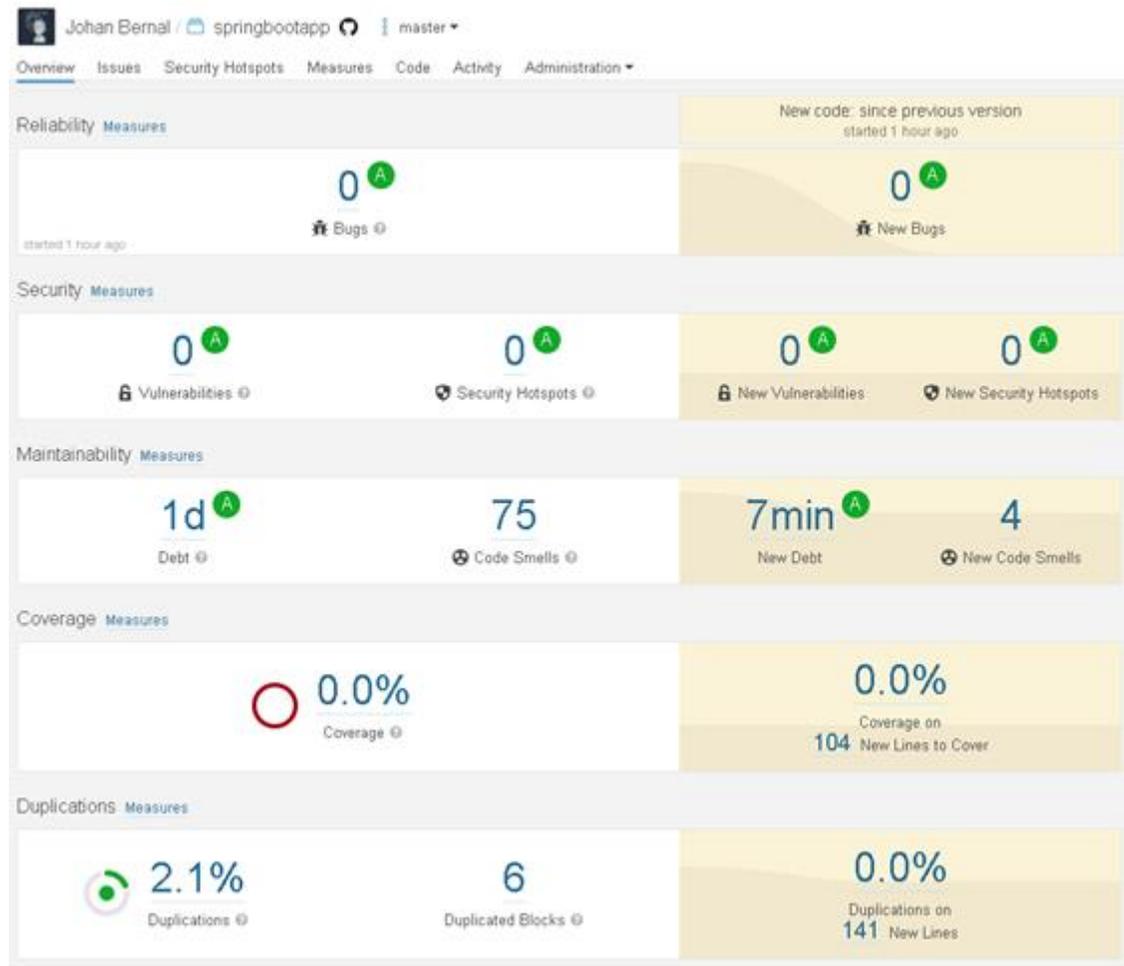


Figura 68. Overview Sonar Cloud ejecución final (Fuente: Elaboración Propia)

Como se puede observar se dio una reducción significativa del 100% en los bugs generados por el Backend Enseña2, una reducción de 100% en los Security Hotspots, reduciendo el 25% a nivel de bloques duplicados, y un 56.40% los code smells presentados en el código del aplicativo. Lo cual es bastante significativo para ser un prototipo de aplicación que aún se encuentra en construcción con clases a medias y en elaboración, denotando así 1 día de deuda técnica en el caso de querer dar solución al 100% de los incidentes.

7.4.5. Pruebas de Seguridad y Vulnerabilidad

En la actualidad, las aplicaciones web se encuentran expuestas a un alto número de amenazas y vulnerabilidades, de las cuales podemos destacar algunas como el mal manejo de las aplicaciones web por parte de sus usuarios, los ataques a la seguridad, la calidad baja en el desarrollo del software o la falta de un adecuado marco de desarrollo bien estructurado. Partiendo de esta realidad, se hizo necesario tomar medidas que permitieran evaluar el desarrollo del prototipo de aplicación web progresiva denominada Enseña2, con el fin de evitar que dichos ataques aprovechen las posibles vulnerabilidades que pueda tener el sistema desarrollado y evaluar estas. Para ello, se desarrollaron pruebas de seguridad con el objetivo de identificar aquellas vulnerabilidades que puedan implicar un nivel de riesgo en la implementación y despliegue del aplicativo Enseña2.

De acuerdo a lo anterior, las herramientas usadas en las pruebas de seguridad implementadas para encontrar las vulnerabilidades conocidas que están categorizadas dentro del top 10 de OWASP fueron: Nikto, Uniscan y OWASP ZAP, estas herramientas son usadas como método de escaneo de las vulnerabilidades que puede poseer la aplicación web a nivel de red, a nivel de configuración y seguridad, para ello se realizó la implementación de una máquina virtual con Kali Linux para hacer uso de estas herramientas. A continuación, podremos identificar el resultado obtenido de los análisis realizados por las herramientas mencionadas anteriormente:

7.4.5.1. *Análisis de Vulnerabilidades con Nikto.*

Nikto es un escáner web útil para realizar pruebas de servidores web/URL y que permite verificar la configuración que puede tener un servidor y exponer archivos de índice y archivos de respaldo que puedan encontrarse en el servidor, entre otras cosas. Por lo tanto, se usa para realizar un análisis de la aplicación web progresiva Enseña2.

El resultado de dicho análisis dejó en evidencia que el sistema desarrollado de la aplicación Enseña2 cuenta con 6 vulnerabilidades como se puede observar en la siguiente imagen:

| 192.168.0.15 / 192.168.0.15 port 4200 | |
|--|--|
| Target IP | 192.168.0.15 |
| Target hostname | 192.168.0.15 |
| Target Port | 4200 |
| HTTP Server | |
| Site Link (Name) | http://192.168.0.15:4200/home/ |
| Site Link (IP) | http://192.168.0.15:4200/home/ |
| URI | /home/ |
| HTTP Method | GET |
| Description | Retrieved x-powered-by header: Express |
| Test Links | http://192.168.0.15:4200/home/ http://192.168.0.15:4200/home/ |
| OSVDB Entries | OSVDB-0 |
| URI | /home/ |
| HTTP Method | GET |
| Description | Retrieved access-control-allow-origin header: * |
| Test Links | http://192.168.0.15:4200/home/ http://192.168.0.15:4200/home/ |
| OSVDB Entries | OSVDB-0 |
| URI | /home/ |
| HTTP Method | GET |
| Description | The anti-clickjacking X-Frame-Options header is not present. |
| Test Links | http://192.168.0.15:4200/home/ http://192.168.0.15:4200/home/ |
| OSVDB Entries | OSVDB-0 |
| URI | /home/ |
| HTTP Method | GET |
| Description | The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS |
| Test Links | http://192.168.0.15:4200/home/ http://192.168.0.15:4200/home/ |
| OSVDB Entries | OSVDB-0 |
| URI | /home/ |
| HTTP Method | OPTIONS |
| Description | Allowed HTTP Methods: GET, HEAD |
| Test Links | http://192.168.0.15:4200/home/ http://192.168.0.15:4200/home/ |
| OSVDB Entries | OSVDB-0 |

Figura 69. Reporte de Análisis de Nikto (Fuente: Elaboración Propia)

Estos resultados nos indican los tipos de métodos ejecutados y una descripción de las vulnerabilidades detectadas de acuerdo con la base de datos abierta e independiente de vulnerabilidades (OSVDB), tales como:

- Retrieved x-powered-by header: Express. Esta vulnerabilidad nos indica que la aplicación web progresiva Enseña2 está retornando un encabezado de respuesta el cual puede permitir conocer la manera en que fue desarrollado el sistema.

- Retrieved access-control-allow-origin header: *. Esta vulnerabilidad nos indica que el servidor de nuestra aplicación puede compartir los recursos de la respuesta con quien realiza la petición.
- The anti-clickjacking X-Frame-Options header is not present. Esta vulnerabilidad indica que falta un encabezado de X-Frame-Options, lo cual permite que un atacante pueda “secuestrar” los clics realizados en la aplicación web progresiva Enseña2 y dirigirlos hacia otro lugar.
- The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS. Esta vulnerabilidad nos indica que la aplicación web progresiva Enseña2 no cuenta con la cabecera X-XSS-Protection la cual puede evitar una secuencia de comandos en sitios cruzados o Cross-site scripting.
- Allowed HTTP Methods: GET, HEAD. Esta vulnerabilidad implica que puede existir un nivel de riesgo en los métodos GET y HEAD de HTTP.

Los demás resultados del análisis realizado por NIKTO se encuentra adjunto en la ruta indicada dentro del apartado 8.5 (Pruebas de Seguridad) que está dentro de los Anexos del presente documento.

7.4.5.2. *Análisis de Vulnerabilidades con Uniscan.*

Uniscan es una herramienta que permite identificar vulnerabilidades en cualquier aplicación web. Esta herramienta nos permite generar reportes a través de un archivo log en el cual se pueden observar las diversas pruebas de seguridad realizadas por el mismo, con el fin de identificar las posibles vulnerabilidades existentes dentro del prototipo de aplicación web progresiva Enseña2.

Dentro del log generado se detalla el proceso de análisis ejecutado por la herramienta Uniscan como: Ping y Traceroute a la IP del servidor de la aplicación, el uso de herramientas como Nslookup y Nmap, entre otras. A continuación, se da algunos detalles de la ejecución de las pruebas realizadas por Uniscan a la aplicación Enseña2:

- En el ping realizado hacia la IP de la aplicación web de Enseña2 se observó que el 100% de los paquetes enviados todos se perdieron y no llegaron al destino.

```
Ping:
PING 192.168.0.15 (192.168.0.15) 56(84) bytes of data.

--- 192.168.0.15 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3067ms
```

Figura 70. Ping realizado por Uniscan (Fuente: Elaboración Propia)

- En la ejecución del Traceroute, Uniscan permite identificar las posibles rutas de los paquetes y promediar la latencia de tráfico y el tiempo que tardó en ida y vuelta a través de la red.

```

Traceroute:
traceroute to 192.168.0.15 (192.168.0.15), 30 hops max, 60 byte packets
 1 * * *
 2 * * *
 3 * * *
 4 * * *
 5 * * *
 6 * * *
 7 * * *
 8 * * *
 9 * * *
10 * * *
11 * * *
12 * * *
13 * * *
14 * * *
15 * * *
16 * * *
17 * * *
18 * * *
19 * * *
20 * * *
21 * * *
22 * * *
23 * * *
24 * * *
25 * * *
26 * * *
27 * * *
28 * * *
29 * * *
30 * * *

```

Figura 71. Traceroute realizado por Uniscan (Fuente: Elaboración Propia)

- Con la ejecución de NSLookup, la herramienta Uniscan trató de localizar el nombre de dominio de la dirección IP de la aplicación web de Enseña2.

```

Nslookup:
Server: 192.168.0.1
Address: 192.168.0.1#53

** server can't find 15.0.168.192.in-addr.arpa: SERVFAIL
** server can't find 15.0.168.192.in-addr.arpa: NXDOMAIN

```

Figura 72. NSLookup realizado por Uniscan (Fuente: Elaboración Propia)

- Por medio de Nmap, Uniscan realizó una exploración y un análisis de seguridad buscando todos los puertos disponibles y abierto para la IP de la aplicación web progresiva Enseña2.

```
Nmap:
Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-22 20:50 EDT
NSE: Loaded 153 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 20:50
Completed NSE at 20:50, 0.00s elapsed
Initiating NSE at 20:50
Completed NSE at 20:50, 0.00s elapsed
Initiating NSE at 20:50
Completed NSE at 20:50, 0.00s elapsed
Initiating ARP Ping Scan at 20:50
Scanning 192.168.0.15 [1 port]
Completed ARP Ping Scan at 20:50, 0.05s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 20:50
Completed Parallel DNS resolution of 1 host. at 20:50, 0.01s elapsed
Initiating SYN Stealth Scan at 20:50
Scanning 192.168.0.15 [1000 ports]
Discovered open port 135/tcp on 192.168.0.15
Discovered open port 3306/tcp on 192.168.0.15
Discovered open port 8090/tcp on 192.168.0.15
Discovered open port 2179/tcp on 192.168.0.15
Discovered open port 5357/tcp on 192.168.0.15
Completed SYN Stealth Scan at 20:50, 5.01s elapsed (1000 total ports)
Initiating Service scan at 20:50
Scanning 5 services on 192.168.0.15
Completed Service scan at 20:51, 21.25s elapsed (5 services on 1 host)
Initiating OS detection (try #1) against 192.168.0.15
Retrying OS detection (try #2) against 192.168.0.15
NSE: Script scanning 192.168.0.15.
Initiating NSE at 20:51
Completed NSE at 20:51, 14.05s elapsed
Initiating NSE at 20:51
Completed NSE at 20:51, 0.19s elapsed
Initiating NSE at 20:51
Completed NSE at 20:51, 0.00s elapsed
Nmap scan report for 192.168.0.15
Host is up (0.00046s latency).
Not shown: 995 filtered ports
```

Figura 73. Nmap realizado por Uniscan (Fuente: Elaboración Propia)

Se identifica que dentro del análisis realizado por Nmap, se observó que la IP del servidor cuenta con 1000 puertos disponibles, de los cuales, 5 se encuentran abiertos representando un riesgo de seguridad de la aplicación web Enseña2.

```
Scanning 192.168.0.15 [1000 ports]
Discovered open port 135/tcp on 192.168.0.15
Discovered open port 3306/tcp on 192.168.0.15
Discovered open port 8090/tcp on 192.168.0.15
Discovered open port 2179/tcp on 192.168.0.15
Discovered open port 5357/tcp on 192.168.0.15
```

Figura 74. Análisis de Puertos de Nmap (Fuente: Elaboración Propia)

De igual manera, Nmap permitió identificar el detalle de algunas herramientas y software que usan dichos puertos, cómo, por ejemplo, MySQL 8.0.23, dejando en evidencia que este hace uso del puerto 3306, implicando con ello un riesgo dado que con realizar una investigación adecuada sobre las vulnerabilidades de MySQL en esa versión los atacantes podrían aprovechar los bugs y/o errores conocidos que pueda existir y atacar el sistema. Adicional a lo anterior, Nmap permite identificar el tipo de certificado SSL que cuenta MySQL, en la cual se observa que es una llave pública del tipo RSA y una llave pública de 2048 bits y se encuentra cifrado con el algoritmo SHA256 de RSA Encryption.

```
3306/tcp open mysql MySQL 8.0.23
| mysql-info:
| Protocol: 10
| Version: 8.0.23
| Thread ID: 29
| Capabilities flags: 65535
| Some Capabilities: Support41Auth, DontAllowDatabaseTableColumn, SupportsCompression, LongPassword, InteractiveClient, FoundRows,
SupportsLoadDataLocal, SupportsTransactions, IgnoreSigpipes, ConnectWithDatabase, SwitchToSSLAfterHandshake, IgnoreSpaceBeforeParenthesis,
Speaks41ProtocolNew, Speaks41ProtocolOld, ODBCClient, LongColumnFlag, SupportsAuthPlugins, SupportsMultipleResults
| Status: Autocommit
| Salt: \x10\x15\x02A\x05}\x16uxM\x08m\x13btew
| _Auth Plugin Name: caching_sha2_password
| ssl-cert: Subject: commonName=MySQL_Server_8.0.23_Auto_Generated_Server_Certificate
| Issuer: commonName=MySQL_Server_8.0.23_Auto_Generated_CA_Certificate
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2021-03-02T06:08:52
| Not valid after: 2031-02-28T06:08:52
| MD5: 200c 0b7b 3029 005b dffc 25b9 7b8b 472f
| _SHA-1: e793 16b9 8a23 55ac b0b0 40da 346d d571 74b2 b012
| _ssl-date: TLS randomness does not represent time
```

Figura 75. Análisis de Puertos abiertos de Nmap (Fuente: Elaboración Propia)

Los demás resultados del análisis realizado por Uniscan se encuentra adjunto en la ruta indicada dentro del apartado 8.5 (Pruebas de Seguridad) que está dentro de los Anexos del presente documento.

7.4.5.3. Análisis de Vulnerabilidades con OWASP ZAP.

OWASP ZAP es un escáner web que permite identificar vulnerabilidades en cualquier aplicativo web a través de diversas pruebas de penetración. Esta herramienta logra identificar algunas alertas que se clasifican de acuerdo con su nivel de criticidad y prioridad: Alerta de Riesgo Alto, Alerta de Riesgo Medio, Alerta de Riesgo Bajo y Alertas Informativas.

De acuerdo con lo anterior, OWASP ZAP a través de su análisis logra identificar 9 alertas relacionadas a vulnerabilidades conocidas en la aplicación web progresiva Enseña2. Dichas alertas están categorizadas de la siguiente manera: tres corresponden a alertas con un nivel de riesgo medio, tres a alertas con un nivel de riesgo bajo y 3 alertas informativas, como se puede observar en la siguiente imagen donde se pueden identificar los resultados obtenidos:

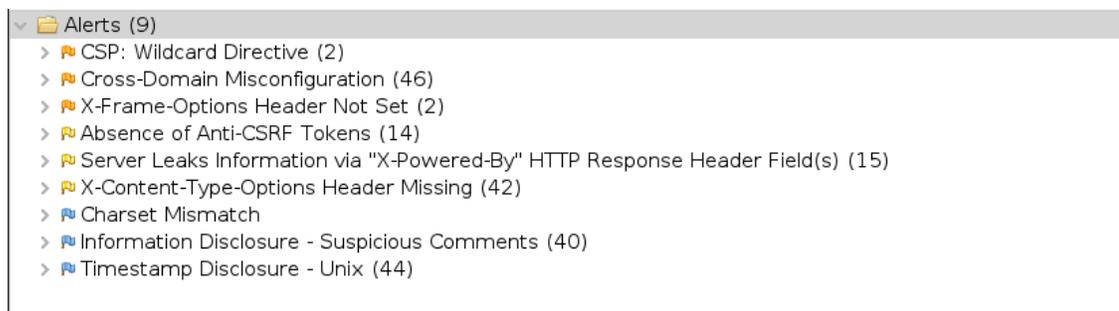


Figura 76. Alertas de Análisis de OWASP ZAP (Fuente: Elaboración Propia)

A continuación, se da a conocer algunas de las alertas generadas por la herramienta OWASP ZAP con respecto al análisis realizado sobre el prototipo de aplicación web progresiva Enseña2, por nivel de riesgo:

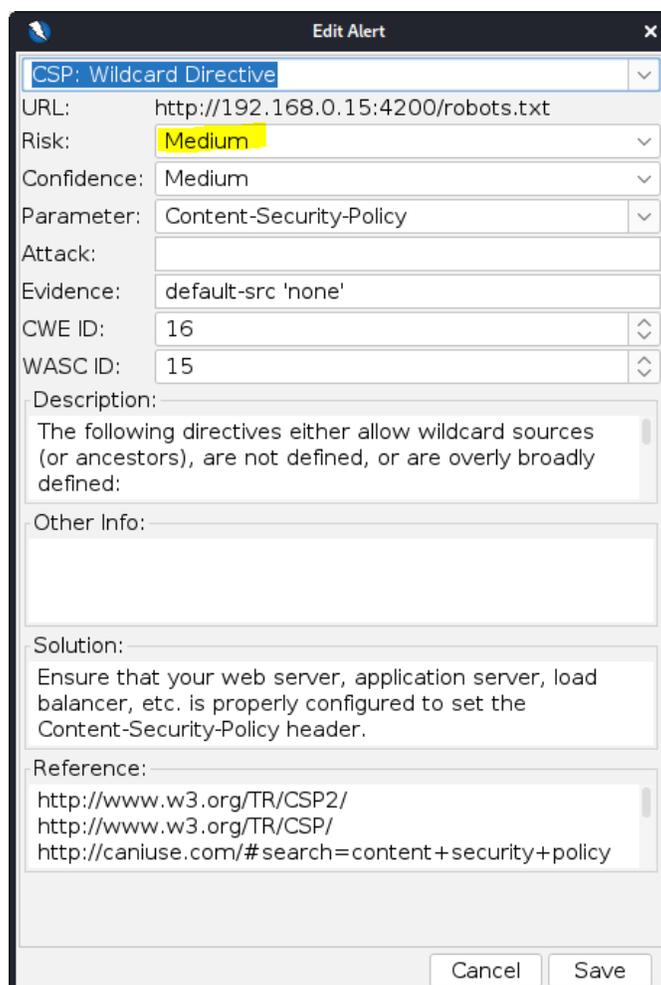


Figura 77. Muestra de Alerta de Prioridad Media (Fuente: Elaboración Propia)

CSP, Wildcard Directive, nivel de riesgo medio, indica que el sistema está permitiendo wildcard sources que no están definidas o se definieron de manera amplia. La solución para ello es asegurar que la aplicación web Enseña2 tenga correctamente configurado el encabezado del Content-Security-Policy.

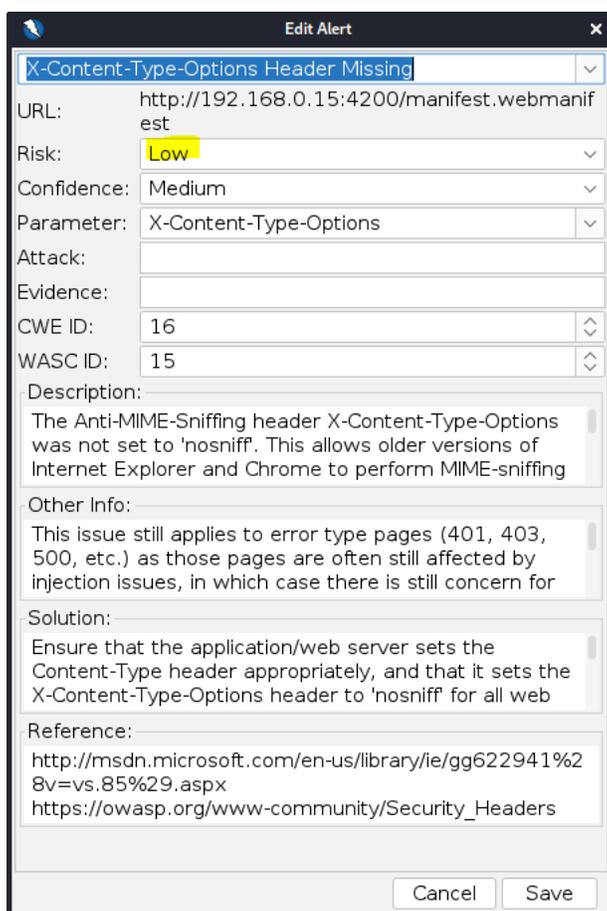


Figura 78. Muestra de Alerta de Prioridad baja (Fuente: Elaboración Propia)

X-Content-Type-Options Header Missing, nivel de riesgo bajo, indica que el sistema está permitiendo que las versiones anteriores de Internet Explorer y Chrome realicen MME-Sniffing. La solución para ello es asegurar que la aplicación web Enseña2 tenga configurado correctamente el encabezado del Content-Type y que se configure el encabezado de X-Content-Type-Option en 'NoSniff' para toda la web.

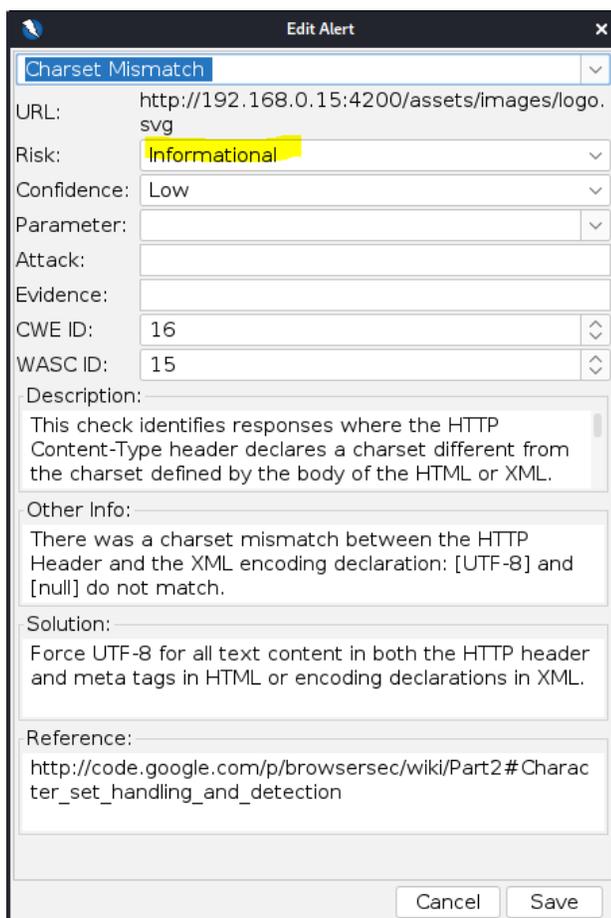


Figura 79. Muestra de Alerta Informativa (Fuente: Elaboración Propia)

Charset Mismatch, a manera informativa indica que se identificaron algunos responses desde el HTTP y que declararon en el encabezado Content-Type un charset diferente a los definidos dentro del cuerpo del HTML o XML. La solución sugerida es forzar UTF-8 para todo el contenido de texto tanto en el encabezado HTTP como en las metaetiquetas en HTML.

Los demás resultados del análisis realizado por OWASP ZAP se encuentra adjunto en la ruta indicada dentro del apartado 8.5 (Pruebas de Seguridad) que está dentro de los Anexos del presente documento.

7.5. Instalación y Configuración

Para la instalación del prototipo de aplicación web progresiva Enseña2 se tuvieron en cuenta las especificaciones técnicas de hardware y software detallados en el punto 5.4.2.2 (Hardware y Software) del presente documento, en el cual se dan las características que debe tener el ambiente productivo en donde se realizará el correcto despliegue y la adecuada configuración del sistema desarrollado.

Partiendo de lo anterior, a continuación, se indican cuáles fueron los pasos seguidos para poder realizar la instalación correcta del prototipo de aplicación web progresiva Enseña2.

7.5.5. Descarga Archivos Fuentes

Para poder realizar la instalación de la aplicación web progresiva Enseña2, es necesario ir a la ruta en Google Drive en la cual están almacenados los archivos fuentes que fueron usados en el proceso de instalación del sistema desarrollado. En de dicha ruta se encuentra un archivo comprimido en WinRAR que contiene una carpeta la cual incluye tres subcarpetas que almacenan las fuentes para la instalación de la Base de Datos, Backend y FrontEnd del aplicativo progresivo Enseña2. Cabe mencionar que para poder acceder a dicha carpeta se debe contar con los permisos de acceso otorgados previamente.



The screenshot shows the Google Drive interface. At the top, there is a search bar with the text 'Buscar en Drive'. Below the search bar, the breadcrumb path is 'Mi unidad > Enseña2 APP'. A table lists the contents of the folder:

| Nombre ↓ | Propietario | Última modificación | Tamaño de archivo |
|-----------------|-------------|---------------------|-------------------|
| Enseña2 APP.rar | yo | 0:39 yo | 41 MB |

Figura 80. Ruta de Fuentes del aplicativo Enseña2 (Fuente: Elaboración Propia)

A continuación, se da a conocer la ruta en Google Drive:

https://drive.google.com/drive/u/1/folders/1kipKaPEELmzUR6kG18Kq_D6NIqVznZJ

Una vez ubicados en la ruta anterior se deberá realizar la descarga del archivo comprimido “Ensen2 APP.rar” (número de imagen) y posterior a ello extraer el contenido del archivo anterior el cual tiene una carpeta llamada “Ensen2 APP” (número de imagen).

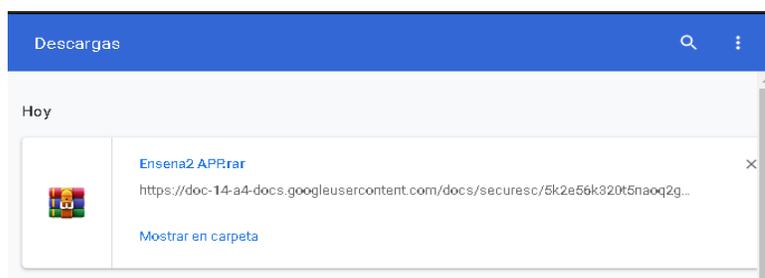


Figura 81. Archivo comprimido descargado (Fuente: Elaboración Propia)

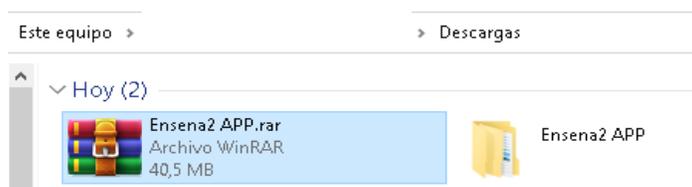


Figura 82. Archivo Descomprimido (Fuente: Elaboración Propia)

Dentro de la carpeta “Ensen2 APP” se encuentran tres subcarpetas como se deberá observar en la siguiente imagen:

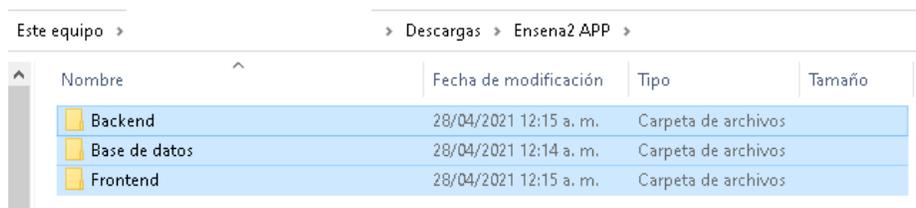
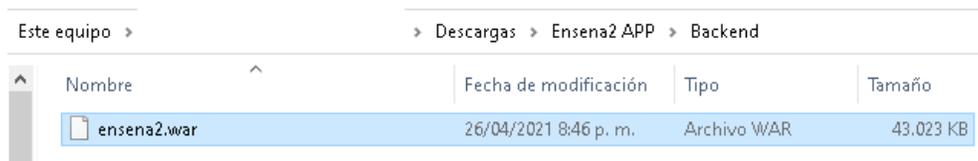


Figura 83. Carpetas de Fuentes (Fuente: Elaboración Propia)

A continuación, se detalla el contenido de cada una de estas carpetas:

- **Backend:** Esta carpeta contiene un archivo WAR (ensena2.war) que es un archivo de aplicación web (Web Application Archive). Este archivo es un comprimido que contiene las aplicaciones basadas en Java que se ejecutarán en el servidor de tipo Tomcat.

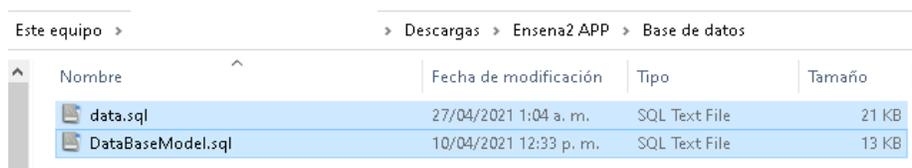


Este equipo > Descargas > Ensenas2 APP > Backend

| Nombre | Fecha de modificación | Tipo | Tamaño |
|-------------|-----------------------|-------------|-----------|
| ensena2.war | 26/04/2021 8:46 p. m. | Archivo WAR | 43.023 KB |

Figura 84. Archivo war (Fuente: Elaboración Propia)

- **Base de datos:** Esta carpeta contiene dos archivos, un archivo script (DataBaseModel.sql) para la creación del esquema de la Base de datos del sistema y un archivo script (data.sql) para la inserción de aquellos datos que son necesarios para la adecuada ejecución del sistema a instalar.



Este equipo > Descargas > Ensenas2 APP > Base de datos

| Nombre | Fecha de modificación | Tipo | Tamaño |
|-------------------|------------------------|---------------|--------|
| data.sql | 27/04/2021 1:04 a. m. | SQL Text File | 21 KB |
| DataBaseModel.sql | 10/04/2021 12:33 p. m. | SQL Text File | 13 KB |

Figura 85. Sentencias SQL (Fuente: Elaboración Propia)

- **FrontEnd:** Por último, esta carpeta contiene el proyecto web (Ensenas2) que será desplegado en el servidor apache y por medio el cual se podrá acceder a la interfaz gráfica del sistema desarrollado.

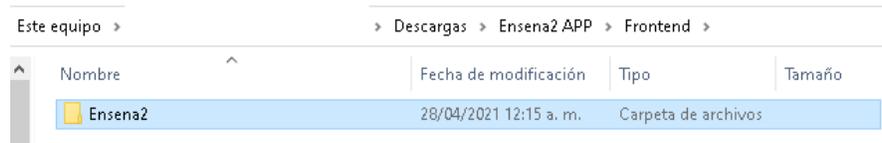


Figura 86. Carpeta Proyecto Web (Fuente: Elaboración Propia)

7.5.6. Instalación

7.5.6.1. Instalación de la Base de Datos.

Para la instalación de la base de datos solo fue necesario realizar la ejecución de los Scripts adjuntos en la carpeta “Base de datos” indicada en el punto anterior. A continuación, se presenta los pasos seguidos del proceso de instalación de la Base de Datos:

- Inicialmente, se ejecutó el archivo “DataBaseModel.sql” que contiene el script con las sentencias DDL para la creación del esquema de la base de datos ensena2db.

```

1 -- MySQL Script generated by MySQL Workbench
2 -- Sat Apr 10 12:33:57 2021
3 -- Model: New Model   Version: 1.0
4 -- MySQL Workbench Forward Engineering
5
6 • SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
7 • SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
8 • SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SL
9
10 -----
11 -- Schema mydb
12 -----
13 -- Schema ensena2db
14 -----
15 -----
16 -----
17 -- Schema ensena2db
18 -----
19 -----
20 • CREATE SCHEMA IF NOT EXISTS `ensena2db` DEFAULT CHARACTER SET utf8 COLLATE utf8_unicode_ci ;
21 • USE `ensena2db` ;
22 -----
23 -----
24 -- Table `ensena2db`.`Administracion`

```

Figura 87. Script de esquema Ensenadb (Fuente: Elaboración Propia)

Una vez ejecutado el script con el esquema de la base de datos se podrá observar que esta quedó correctamente creada en el servidor de Base de Datos.



Figura 88. Schema Ensenadb (Fuente: Elaboración Propia)

- Posterior a esto, se debe ejecutar el archivo “data.sql” que contiene el script con las sentencias DML para la inserción de los datos en la base de datos “ensena2db” con el fin de adecuar el ambiente para la correcta ejecución del sistema.

The screenshot shows a MySQL Data Modeler window with a script editor and an output pane. The script editor contains a MySQL dump script for the 'ensena2db' database. The output pane shows the execution of several SET statements, all of which were successful and affected 0 rows.

```

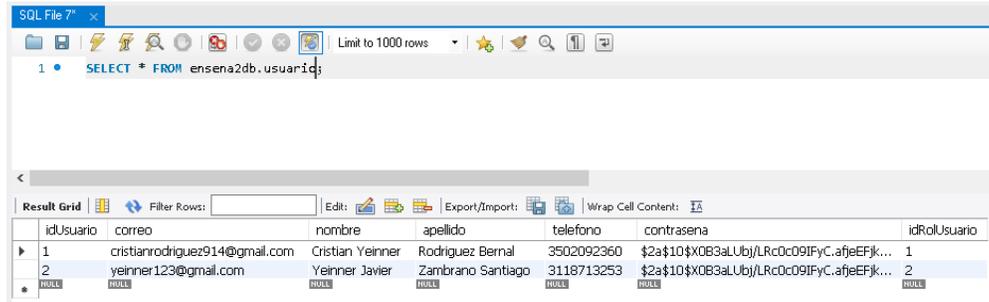
1  -- MySQL dump 10.13  Distrib 8.0.23, for Linux (x86_64)
2  --
3  -- Host: localhost    Database: ensena2db
4  --
5  -- Server version 8.0.23
6
7  /*140101 SET @OLD_CHARACTER_SET_CLIENT=@CHARACTER_SET_CLIENT */;
8  /*140101 SET @OLD_CHARACTER_SET_RESULTS=@CHARACTER_SET_RESULTS */;
9  /*140101 SET @OLD_COLLATION_CONNECTION=@COLLATION_CONNECTION */;
10 /*150503 SET NAMES utf8mb4 */;
11 /*140103 SET @OLD_TIME_ZONE=@TIME_ZONE */;
12 /*140103 SET TIME_ZONE='+00:00' */;
13 /*140014 SET @OLD_UNIQUE_CHECKS=@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
14 /*140014 SET @OLD_FOREIGN_KEY_CHECKS=@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
15 /*140101 SET @OLD_SQL_MODE=@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
16 /*140111 SET @OLD_SQL_NOTES=@SQL_NOTES, SQL_NOTES=0 */;

```

| # | Time | Action | Message | Duration / Fetch |
|-----|----------|--|-------------------|------------------|
| 177 | 23:51:03 | /*140014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */ | 0 row(s) affected | 0.000 sec |
| 178 | 23:51:03 | /*140014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */ | 0 row(s) affected | 0.000 sec |
| 179 | 23:51:03 | /*140101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */ | 0 row(s) affected | 0.000 sec |
| 180 | 23:51:03 | /*140101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS | 0 row(s) affected | 0.000 sec |
| 181 | 23:51:03 | /*140101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */ | 0 row(s) affected | 0.000 sec |
| 182 | 23:51:03 | /*140111 SET SQL_NOTES=@OLD_SQL_NOTES */ | 0 row(s) affected | 0.000 sec |

Figura 89. Script de Data Ensena2db (Fuente: Elaboración Propia)

Por último, se deberá realizar la verificación de que los datos insertados están en la base de datos, para lo cual se deberá ejecutar la consulta a las tablas “usuario” y “rolusuario” de la base de datos “ensena2db” como se observa en las siguientes imágenes:



SQL File 7* x

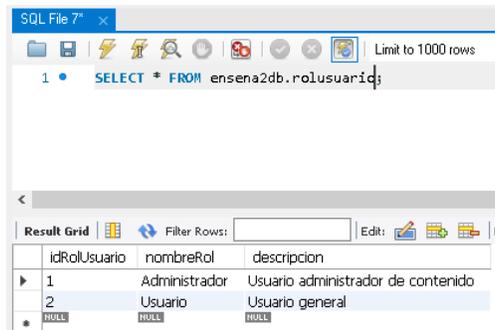
Limit to 1000 rows

1 • SELECT * FROM ensena2db.usuariid;

Result Grid

| | idUsuario | correo | nombre | apellido | telefono | contrasena | idRolUsuario |
|---|-----------|--------------------------------|------------------|-------------------|------------|---|--------------|
| ▶ | 1 | cristianrodriguez914@gmail.com | Cristian Yeinner | Rodriguez Bernal | 3502092360 | \$2a\$10\$X0B3aLubj/LRc0c09IFyC.afjeEFjk... | 1 |
| | 2 | yeinner123@gmail.com | Yeinner Javier | Zambrano Santiago | 3118713253 | \$2a\$10\$X0B3aLubj/LRc0c09IFyC.afjeEFjk... | 2 |
| * | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

Figura 90. Consulta tabla Usuario (Fuente: Elaboración Propia)



SQL File 7* x

Limit to 1000 rows

1 • SELECT * FROM ensena2db.rolusuariid;

Result Grid

| | idRolUsuario | nombreRol | descripcion |
|---|--------------|---------------|------------------------------------|
| ▶ | 1 | Administrador | Usuario administrador de contenido |
| | 2 | Usuario | Usuario general |
| * | NULL | NULL | NULL |

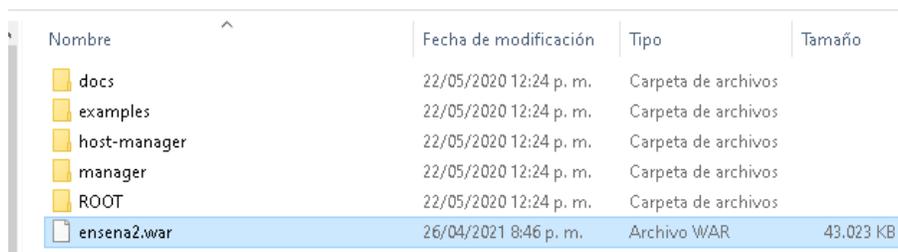
Figura 91. Consulta tabla RolUsuario (Fuente: Elaboración Propia)

Una vez confirmado que la Base de datos fue creada correctamente se procedió a realizar la instalación del Backend.

7.5.6.2. *Instalación Backend.*

Para la adecuada configuración del Backend se deberá contar con un servidor Tomcat, específicamente en su versión 9.0.43. Dentro de dicho servidor nos ubicamos en la carpeta webapps que se encuentra en la siguiente ruta: C:\tomcat\webapps.

Tomamos el archivo war “ensena2.war” que se encuentra en la carpeta “Backend” de las fuentes para la instalación del sistema y lo ubicamos en la ruta anterior cómo se observa en la siguiente imagen:



| Nombre | Fecha de modificación | Tipo | Tamaño |
|--------------|------------------------|---------------------|-----------|
| docs | 22/05/2020 12:24 p. m. | Carpeta de archivos | |
| examples | 22/05/2020 12:24 p. m. | Carpeta de archivos | |
| host-manager | 22/05/2020 12:24 p. m. | Carpeta de archivos | |
| manager | 22/05/2020 12:24 p. m. | Carpeta de archivos | |
| ROOT | 22/05/2020 12:24 p. m. | Carpeta de archivos | |
| ensena2.war | 26/04/2021 8:46 p. m. | Archivo WAR | 43.023 KB |

Figura 92. Ruta Tomcat (Fuente: Elaboración Propia)

Una vez el archivo war quede en la ruta, se deberá realizar el correcto despliegue o inicialización de la app en el servidor. Luego de esto, al ingresar a la plataforma de administración del servidor se confirma que el aplicativo Ensenas2 se encuentra ejecutándose como se observa en la siguiente imagen.

| Aplicaciones | | | |
|---------------|----------------------|---------------------------------|--------------|
| Ruta | Versión | Nombre a Mostrar | Ejecutándose |
| / | Ninguno especificado | Welcome to Tomcat | true |
| /docs | Ninguno especificado | Tomcat Documentation | true |
| /ensena2 | Ninguno especificado | | true |
| /examples | Ninguno especificado | Servlet and JSP Examples | true |
| /host-manager | Ninguno especificado | Tomcat Host Manager Application | true |
| /manager | Ninguno especificado | Tomcat Manager Application | true |

Figura 93. Ejecución Tomcat (Fuente: Elaboración Propia)

Para confirmar que la aplicación Backend está ejecutándose correctamente, se accedió por medio de la siguiente URL: <https://localhost:8080/ensena2/>. A continuación, se deja una imagen con la evidencia de conexión al Backend.

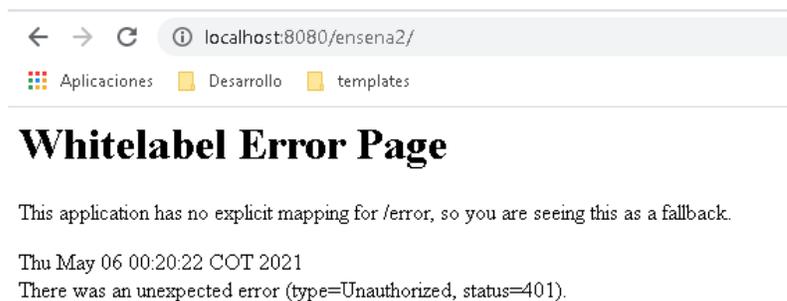
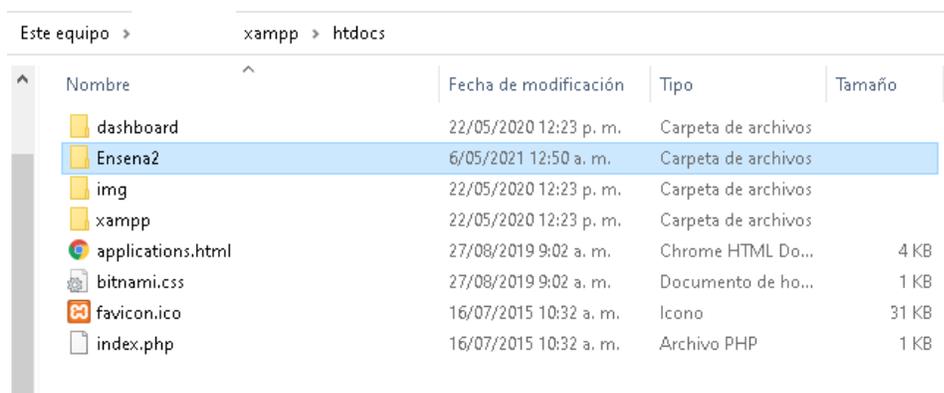


Figura 94. Backend disponible (Fuente: Elaboración Propia)

Por último, para garantizar que haya comunicación entre el Backend y la Base de datos realizamos el consumo del servicio de autenticación a través de un request enviado desde la aplicación Postman como se muestra en la imagen a continuación:

7.5.6.3. Instalación FrontEnd.

Para la correcta configuración del servidor FrontEnd se utilizó el servidor apache 2.4.46. Para realizar la instalación del proyecto web del aplicativo Enseña2 se tomó la carpeta llamada “Ensenas2” que contiene la fuente del proyecto web y que está dentro de la carpeta “Frontend”, y luego se dejó en la siguiente ruta del servidor:\htdocs del servidor apache como se observa en la siguiente imagen:



| Nombre | Fecha de modificación | Tipo | Tamaño |
|-------------------|------------------------|---------------------|--------|
| dashboard | 22/05/2020 12:23 p. m. | Carpeta de archivos | |
| Ensenas2 | 6/05/2021 12:50 a. m. | Carpeta de archivos | |
| img | 22/05/2020 12:23 p. m. | Carpeta de archivos | |
| xampp | 22/05/2020 12:23 p. m. | Carpeta de archivos | |
| applications.html | 27/08/2019 9:02 a. m. | Chrome HTML Do... | 4 KB |
| bitnami.css | 27/08/2019 9:02 a. m. | Documento de ho... | 1 KB |
| favicon.ico | 16/07/2015 10:32 a. m. | Icono | 31 KB |
| index.php | 16/07/2015 10:32 a. m. | Archivo PHP | 1 KB |

Figura 96. Ruta htdoc XAMPP (Fuente: Elaboración Propia)

Para ejecutar el FrontEnd fue necesario inicializar el servidor APACHE en el cual se encuentra alojada la carpeta del proyecto web. Una vez el servidor está corriendo se realiza la búsqueda en el navegador a través de la IP y el puerto configurado en el servidor apache, en el caso de este ejercicio práctico, nuestra IP y puerto fueron, <https://192.168.0.15:8081/home>. En la siguiente imagen se podrá observar el acceso exitoso.

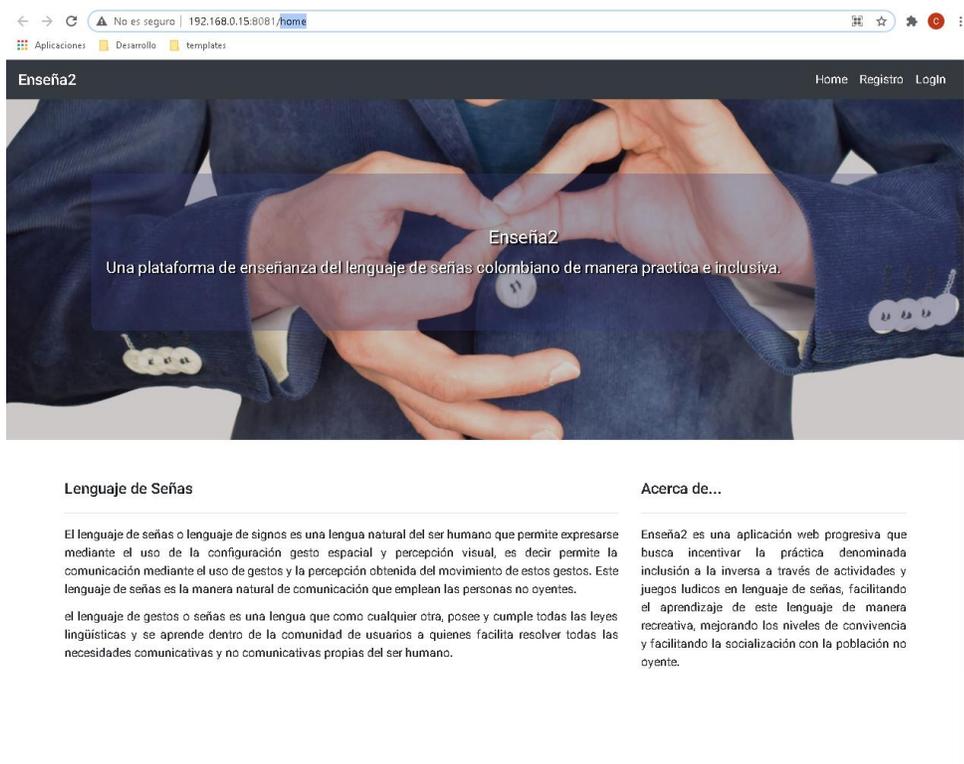


Figura 97. Aplicativo Enseña2 en línea (Fuente: Elaboración Propia)

Cabe destacar que el anterior enlace apunta hacia la IP del servidor que está configurado en 192.168.0.15 y el puerto configurado para este es el 8081. De igual forma, es posible acceder también por medio del localhost.

Adicionalmente, dado que esta es una aplicación web progresiva, se realiza la prueba accediendo desde un dispositivo móvil y al apuntar al enlace del aplicativo desde el navegador Google Chrome este da la opción de instalar el aplicativo en la pantalla principal del dispositivo móvil como se observa en la siguiente imagen:



Figura 98. Aplicativo Enseña2 Progresivo (Fuente: Elaboración Propia)

Por último, se valida la pantalla principal del dispositivo móvil y se observa que el acceso directo al aplicativo se encuentra correctamente creado como se muestra en la imagen a continuación:

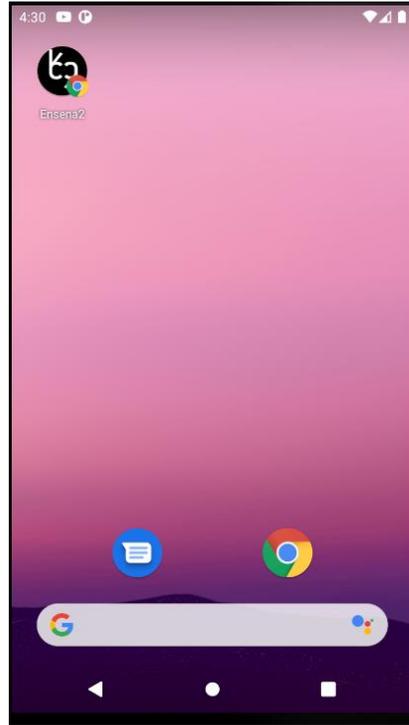


Figura 99. Acceso Directo Aplicativo Enseña2 (Fuente: Elaboración Propia)

Una vez el acceso directo esté ubicado en la pantalla principal, se podrá acceder a esta aplicación como si fuera una aplicación nativa del sistema, y gracias a que es una aplicación web progresiva, permitirá acceder y usar de este, aunque no se cuente con conectividad hacia la web, dado que la aplicación almacenará datos en memoria caché.

8. Anexos

En esta sección podremos encontrar todos los anexos que hicieron parte o apoyo de la elaboración de este documento, los cuales son de relevante importancia, pero no se encontraban en ninguno de los apartados adscritos en este documento:

8.1. Modelo Entidad Relación (MER)

Para el estudio e identificación de las entidades del sistema y sus respectivas relaciones, se planteó la elaboración de un MER (Modelo Entidad Relación) de alto nivel, de elaboración propia, con el fin de ilustrar estas entidades, y su relación entre sí, y el rol de estas entidades dentro del sistema enseñado.

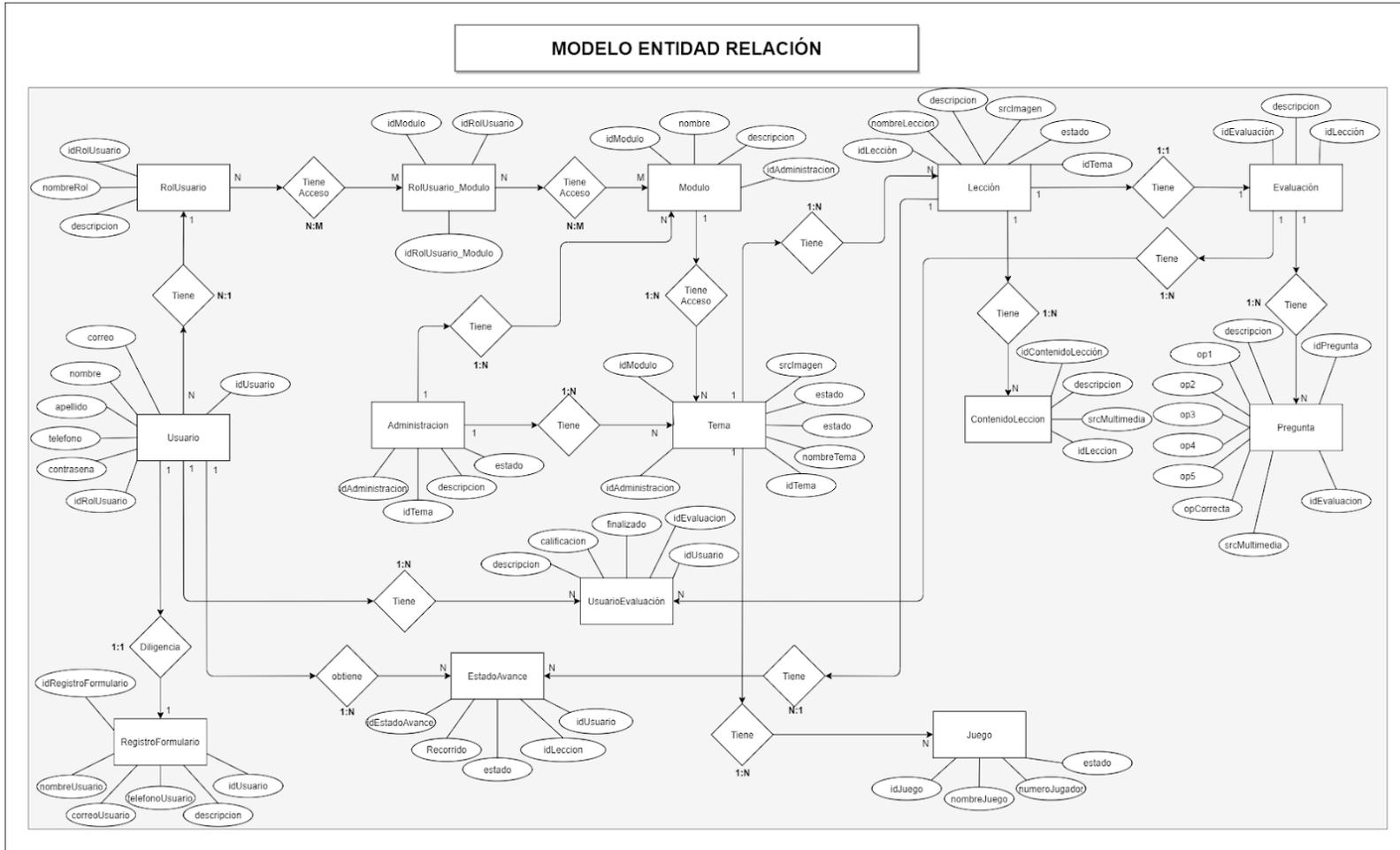


Figura 100. Diagrama MER (Fuente: Elaboración Propia)

8.2. Modelo EER

El siguiente modelo es el esquema gráfico de la Base de Datos del sistema desarrollado y extraído desde el gestor de BD MySQL Workbench 8.0. Dicho esquema ilustra las entidades creadas en la BD y su relación entre sí.

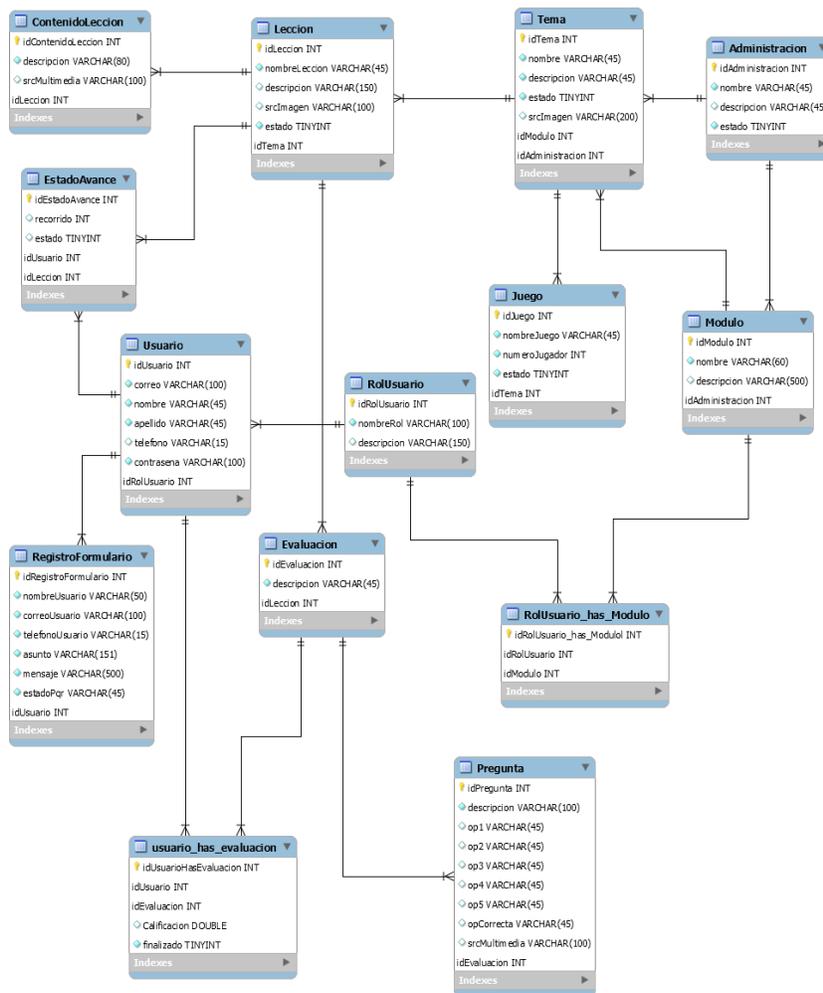


Figura 101. Diagrama ERR (Fuente: Elaboración Propia)

8.3. Lista de Chequeo de las Historias de Usuarios

A continuación, se adjunta el enlace de la ruta en drive donde se encuentra el archivo Excel “Lista de Chequeo de HU” que contiene las pruebas realizadas con el fin de garantizar el cumplimiento de los criterios de aceptación establecidos dentro de cada una de las historias de usuarios.

<https://drive.google.com/drive/u/1/folders/1fkA8kqEUEMEnvtjjiZ1PEuUwb5hd46nx>

Cabe mencionar que para acceder a dicha ruta se debe contar con los permisos necesarios de acceso.

8.4. SDL - Aplicación Enseña2

A continuación, se adjunta el enlace de la ruta en drive donde se encuentra el archivo Excel “SDL – Aplicación Enseña2” que contiene las pruebas realizadas con el fin de garantizar el cumplimiento de los criterios de aceptación establecidos dentro de cada una de las historias de usuarios.

<https://drive.google.com/drive/u/1/folders/1fkA8kqEUEMEnvtjjiZ1PEuUwb5hd46nx>

8.5. Pruebas de Seguridad

A continuación, se adjunta el enlace de la ruta en drive donde se encuentra los archivos con los resultados y reportes de los análisis realizados con las herramientas Nikto, Uniscan y OWASP ZAP. Para acceder a dicha ruta debe contar con los permisos necesarios.

<https://drive.google.com/drive/u/1/folders/1tPSAFEGyWcVkXV0QdE2vF8xGJmCU9SKq>

9. Conclusiones

Una vez finalizada la contextualización, el levantamiento de los requerimientos funcionales y no funcionales del sistema a través de las historias de usuario captadas por el Product Owner, la adecuada definición del marco de trabajo necesario para la debida gestión del proyecto, la correcta identificación del patrón arquitectónico y patrones de diseño ideales para la implementación del sistema y la acertada selección de las herramientas tecnológicas necesarias para la debida ejecución del desarrollo del proyecto, se logró cumplir el objetivo principal cuyo fin era diseñar y desarrollar un prototipo de una plataforma de enseñanza del lenguaje señas colombiano mediante una aplicación web progresiva, con un enfoque hacia la población oyente, promoviendo con ello la inclusión social a la inversa mediante el uso de actividades lúdicas. En base a lo anterior se puede concluir que, gracias al fácil acceso a dispositivos electrónicos como computadores y celulares, trae una gran ventaja a nivel de accesibilidad al conocimiento por parte de cualquier individuo y/o población, y para esto se encuentran en el mercado miles de aplicaciones, en este caso enfocadas a la inclusión social, pero todas estas se encuentran orientadas a la personas que cuentan con discapacidad, lo cual, aísla a estas aplicaciones a pertenecer a una población específica, por ende el desarrollo propuesto por nosotros en este proyecto, es un prototipo de aplicación web progresiva enfocada a la inclusión social a la inversa o a las personas que no presentan dicha discapacidad, buscando con ello promover el auto aprendizaje del lenguaje de señas Colombianas o LSC como primera instancia a través del uso de archivos multimedia o de gifs, logrando no solo promocionar el aplicativo Enseña2, sino que con los conocimientos adquiridos en nuestro módulo de aprendizaje, pueda esto permitir a la población oyente el deseo de querer interactuar con población no oyente rompiendo esa brecha del lenguaje. Como segunda instancia el uso de evaluaciones y

juegos lúdicos propuestos en la aplicación promueve el aprender este lenguaje de manera recreativa y didáctica, atrayendo aún más la atención de los usuarios oyentes a conocer este lenguaje, no sólo como aprendizaje si no dando pequeños pasos en el acercamiento con las poblaciones no oyentes y vulnerables, partiendo del principio de romper la brecha y dar un paso adelante en la inclusión social a la inversa.

Impulsar la elaboración de una aplicación web progresiva enfocada a la inclusión social a la inversa, requirió que se generará un equipo en el cual alguno de sus participantes tuviera como segunda lengua nativa el lenguaje de señas colombiano, por ende se conformó un equipo Scrum en el cual nuestro Product Owner (Jimmy Carrillo), contará con las habilidades necesarias tanto para solicitar los requerimientos, como para apoyar la creación de contenido, ya definido esto, se establecieron las ceremonias principales de Scrum que permitieron el impulso a la creación del proyecto, por ende se estableció como ejercicios de la ejecución de proyecto las ceremonias de Sprint Planing en donde se establece las actividades en el product backlog, y se escogen las actividades a realizar (sprint backlog) en cada uno de los 6 Sprints que se realizaron. Se realizó, como segunda ceremonia importante, el Daily Scrum, en el cual cada día a mediodía durante el tiempo de desarrollo de los 6 Sprints, nos reunimos para validar que se iba a realizar durante el día, si se tenían problemas y cómo los podríamos solucionar. La ceremonia de Sprint Review se ejecutaron durante los periodos de finalización de cada sprint, validando las entregas de valor conjunto a las historias de usuario y los criterios de aceptación propuestos, esto impulsando la velocidad de desarrollo y el incremento de valor del aplicativo. La ceremonia de Sprint

Retrospective en la cual validamos la viabilidad del producto y el cómo y en qué estado se encontraba el proyecto para finalmente realizar un refinamiento e iniciar un nuevo Sprint.

Desarrollar un proyecto bajo un marco de trabajo ágil como lo es SCRUM ofrece muchos beneficios dado que permite tener el foco en el cliente. Es por ello por lo que al estar atento a cuáles son las necesidades de estos y comprender qué es lo que ellos esperan que haga el sistema, ayuda a poder tener en claridad de qué manera se deberá ejecutar y cómo cumplir con las expectativas de estos. Gracias a las herramientas que ofrece SCRUM como las ceremonias diarias, el equipo puede estar en sintonía de lo que se está realizando, y si existe algún bloqueo en alguien del equipo, dado que SCRUM promueve el trabajo en equipo, se puede dar retroalimentación y llegar a una solución en poco tiempo. Uno de los beneficios más claro que ofrece SCRUM radica en la revisión continua que se realiza al trabajo y al equipo de desarrollo ya que esto da lugar a que el grupo esté en constante retroalimentación promoviendo la mejora continua, lo cual impacta significativamente la productividad, superando con ello las expectativas de los clientes.

De acuerdo con el análisis generado junto al equipo de desarrollo y al equipo de scrum estructurado para la elaboración del proyecto, la arquitectura propuesta, óptima y utilizada para la elaboración del prototipo de aplicación denominada como Enseña2, es el patrón MVC (Modelo Vista Controlador), ya que permite un desarrollo rápido, modular y sostenible, permitiendo así separar las responsabilidades del aplicativo a través de sus módulos, ejerciendo la vista a nivel del FrontEnd mediante el uso de angular, y el modelo y controlador a través del Backend impulsados bajo la tecnología de las APIS a través de SpringBoot.

Conjunto a la elaboración de los dos objetivos anteriores se logró realizar el análisis, diseño y desarrollo del proyecto denominado Enseña2 de manera fluida, funcional y estable técnicamente, utilizando las tecnologías de Angular y PWA para el FrontEnd del proyecto, Springboot para el Backend del proyecto y finalmente MySQL como Servidor de base de datos de manera correcta y netamente funcional.

La arquitectura MVC promueve de manera significativa un desarrollo escalable y permite que el trabajo pueda dividirse de forma sencilla, lo cual ayuda a tener una clara visión de la lógica del sistema, facilitando su mantenimiento y la escalabilidad de este. Este patrón arquitectural facilitó que las pruebas unitarias realizadas sobre cada uno de los componentes desarrollados dentro del proyecto hayan sido mucho más sencillos de hacer lo cual ayudó a identificar errores a tiempo para así poder corregirlos y poder mejorar el sistema. Dado el tipo de desarrollo que se lleva a cabo dentro de este estilo arquitectónico, la reutilización de código es mucho más sencilla, lo cual ayudó a agilizar el proceso de construcción del software.

El paradigma de la programación orientada a objetos promueve un desarrollo escalable, ya que permite obtener una estructura modular clara del sistema desarrollado, facilitando el mantenimiento del código y la modificación de los objetos existentes. Este paradigma, alineado junto al estilo arquitectural, promovió un desarrollo fluido de cada uno de sus componentes, facilitando en gran medida un ahorro en el tiempo de desarrollo y a una mayor productividad. Adicional a lo anterior, este tipo de paradigma nos ayudó a tener un código más ordenado y fácil de refactorizar.

Diseñar y mantener un sistema seguro es una labor que cada día cobra fuerzas en un mundo globalizado y en constante evolución, teniendo en cuenta que, dentro de esta era digital, un activo muy importante en las organizaciones es la información. Es por ello por lo que dentro del desarrollo de cualquier sistema de información se debe plantear estrategias y mecanismos que faciliten la protección de los datos manejados. Considerando lo dicho anteriormente, al implementar un mecanismo de autenticación por token o Jwt, se blindan los datos de los usuarios y se evita que estos sean manipulados y accedidos por terceros desconocidos y no deseados. Este mecanismo de autenticación ayuda a validar que un usuario esté debidamente logueado en la aplicación ya que el sistema valida que el usuario tenga una cadena cifrada, la cual es el token, y que sirve para que el usuario pueda informar al servidor que es un cliente autorizado para consumir los servicios ofrecidos por este. Cabe destacar que el propósito principal de este mecanismo es garantizar que la identidad de un usuario sea cierta, por otro lado, para ofrecer más seguridad al usuario dentro del aplicativo, el cifrado de la contraseña es indispensable, para hacer seguro los temas de autenticación, cabe destacar que en el ejercicio de aplicativo enseña2, se realizó el cifrado del token generado con Jwt y la contraseña, mediante el uso del algoritmo denominado BCrypt.

Debido a que el desarrollo del presente proyecto fue pensado para equipos y dispositivos móviles, durante la construcción del sistema se identificaron cuáles son los tamaños más comunes en los dispositivos móviles más populares. Partiendo de esto, en la codificación de la interfaz gráfica, en cada uno de sus componentes, se establecieron las configuraciones necesarias para que el sistema pudiera adaptar su interfaz al tamaño del dispositivo en el cual se acceda a este. El hecho de que nuestro sistema pueda adaptarse de forma efectiva y eficiente en diferentes ambientes ayuda significativamente a reducir el número de usuarios inconformes, ya que no todos cuentan con equipos que tengan un tamaño estándar, sino que existe una gran variedad.

10. Referencias

- Werbach, K and Hunter, D. (2012). For the Win: How Game Thinking Can Revolutionize Your Business. Philadelphia, PA: Wharton Digital Press.
- Agencia de Noticias UN. 2019. «Agencia de Noticias UN - UNIMEDIOS: Universidad Nacional de Colombia». Recuperado 17 de septiembre de 2020 ().
- Arenas, Miguel. 2020. «IncluSeñas - Aplicaciones en Google Play». Recuperado 17 de septiembre de 2020 (<https://play.google.com/store/apps/details?id=app.aresan.miguel.inclusenas&hl=es>).
- Soporte UN CIER. 2016. «DICcionario DE PALABRAS EN LENGUA DE SEÑAS (sordos) | Colombia Aprende». Recuperado 17 de septiembre de 2020 (<http://aprende.colombiaaprende.edu.co/es/node/89797>).
- Es3Colombia. 2020. «Hablemos LSC - Apps en Google Play». Recuperado 17 de septiembre de 2020 (https://play.google.com/store/apps/details?id=app.es3colombia.hablemos.myapplication&hl=es_CO).
- Garza Gutiérrez, Rocío. 2015. «Dilo en señas - Aplicaciones en Google Play». Recuperado 17 de septiembre de 2020 (<https://play.google.com/store/apps/details?id=com.jaguarlabs.lsm&hl=es>).
- Hindi Infoware. 2020. «Sign Language - Aplicaciones en Google Play». Recuperado 17 de septiembre de 2020 (<https://play.google.com/store/apps/details?id=com.bhbharesh.SignLanguage&hl=es>).
- INSOR. s. f. «INSOR Educativo - Aprende Lengua de Señas Colombiana LSC». INSOR Educativo. Recuperado 17 de septiembre de 2020 (<http://educativo.insor.gov.co/>).
- SAMPER PIZANO, ERNESTO, OCAMPO JOSÉ, y DUQUE DE OSPINA OLGA. 1996. «Ley 324 de 1996 - EVA - Función Pública». Recuperado 17 de septiembre de 2020 (<https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=349>).
- Mendez, J. A.. 2018. «SEP: Lengua de señas colombiana - Apps en Google Play». Recuperado 17 de septiembre de 2020 (https://play.google.com/store/apps/details?id=air.edu.ibero.SEP&hl=es_CO).
- NMD STUDIO. 2019. «aprender lenguaje de señas - Aplicaciones en Google Play». Recuperado 17 de septiembre de 2020 (<https://play.google.com/store/apps/details?id=mimo.language.sign&hl=es>).

- Ceibal LSU. 2017. «Hablemos otra lengua». Recuperado 17 de septiembre de 2020 (<https://www.ceibal.edu.uy/es/articulo/ceibal-lsu-en-las-tablets>).
- MINTIC, y FENASCOL. s. f. «Centro de relevo - Quiénes somos». Centro de relevo. Recuperado 17 de septiembre de 2020 (<http://centroderelvo.gov.co/632/w3-propertyvalue-15347.html>).
- Rodríguez, Manuel Ignacio, y Rocío del Pilar Velásquez G. 2000. «Historia y gramática de la lengua de señas». *Pedagogía y Saberes*. Recuperado 17 de septiembre de 2020 (<https://revistas.pedagogica.edu.co/index.php/PYS/article/view/6242>).
- Rozo Melo, Nancy. s. f. «Portal de Lenguas de Colombia: Diversidad y contacto». Recuperado 17 de septiembre de 2020 (<https://lenguasdecolombia.caroycuervo.gov.co/contenido/Lenguas-de-senas-colombiana/introduccion>).
- Vercher, Enrique. 2018. «La lengua de señas». *Agestrad*. Recuperado 17 de septiembre de 2020 (<https://www.agestrad.com/la-lengua-de-senas/>).
- Vázquez Pérez, José Armando, Manuel Rodríguez Gómez, y Roberto Marín García. 2012. «El software educativo en el proceso de enseñanza-aprendizaje de la disciplina Historia de Cuba». *EDUMECENTRO* 4(3):64-72.
- Díaz, Lía Trinidad Castrillón. 2017. «Los Juegos y su Rol en el Aprendizaje de una Lengua». *Revista La Tercera Orilla* 8.
- The Lego Foundation. 2018. «UNICEF-Lego-Foundation-Aprendizaje-a-traves-del-juego.pdf». Recuperado 17 de septiembre de 2020 (<https://www.unicef.org/sites/default/files/2019-01/UNICEF-Lego-Foundation-Aprendizaje-a-traves-del-juego.pdf>).
- Valencia, Bairon Jaramillo, Ayda Mabel Borja Perlaza, y Daniela Ríos Ortiz. 2019. «Influencia del proceso de inclusión a la inversa en el contexto educativo». *Pensamiento Americano* 12(24). Recuperado 17 de septiembre de 2020 (<https://publicaciones.americana.edu.co/index.php/pensamientoamericano/article/view/311/321>).
- Martínez, Mauricio, y Verónica Capurro. s. f. «Una Experiencia de Inclusión Inversa: Programa Jugando También se Aprende.» 1.
- Larenas, Claudio Heraldo Díaz, y Lilian Jansson Bruce. 2011. «El aprendizaje del inglés y el uso de tecnologías: percepciones de estudiantes y profesores de inglés del nivel secundario chileno». *Matices en Lenguas Extranjeras* 0(5):1-37.

- LSApp, y Posibillian Tech S.A. s. f. «LSApp - Apps en Google Play». Recuperado 18 de septiembre de 2020 (https://play.google.com/store/apps/details?id=com.lsapp&hl=es_CO).
- Pérez de Arado, Belén. 2011. «¿Lengua de señas? – Cultura Sorda». Recuperado 18 de septiembre de 2020 (<https://cultura-sorda.org/lengua-de-senas/>).
- Unesco. 2006. «Educación Inclusiva». Recuperado 18 de septiembre de 2020 (<http://www.inclusioneducativa.org/ise.php?id=1>).
- Parra Dussan, Carlos. 2010. «Dialnet-EducacionInclusivaUnModeloDeEducacionParaTodos-3777544.pdf».
- Unesco. s. f. «General_Presentation-48CIE-4__Spanish_.pdf». Recuperado 18 de septiembre de 2020 (http://www.ibe.unesco.org/fileadmin/user_upload/Policy_Dialogue/48th_ICE/General_Presentation-48CIE-4__Spanish_.pdf).
- Muente, Gabriela. 2019. «Software educativo: el uso de la tecnología en favor del aprendizaje». *Rock Content*. Recuperado 18 de septiembre de 2020 (<https://rockcontent.com/es/blog/software-educativo/>).
- Amaya, Tania Melina Euceda, y Nelly Maradiaga. 2007. «EL JUEGO DESDE EL PUNTO DE VISTA DIDÁCTICO A NIVEL DE EDUCACIÓN PREBÁSICA». 141.
- Cuervas, Javier. 2019. «¿Qué es Spring Framework? Características I». *¿Qué es Spring Framework?* Recuperado 23 de septiembre de 2020 (<https://www.atsistemas.com/es/blog/qu-es-spring-framework-caractersticas-i>).
- Spring.io. s. f. «Spring Data JPA». *Spring Data JPA*. Recuperado 23 de septiembre de 2020 (<https://spring.io/projects/spring-data-jpa>).
- Spring.io. s. f. «Spring Session». *Spring Session*. Recuperado 23 de septiembre de 2020 (<https://spring.io/projects/spring-session>).
- TORRES, RICARDO. s. f. «¿Qué es Angular y por qué elegirlo?» *Anexsoft*. Recuperado 23 de septiembre de 2020 (<https://anexsoft.com/que-es-angular-y-por-que-elegirlo>).
- Rojas, Carlos. 2019. «El patrón PRPL en Angular». Recuperado 23 de septiembre de 2020 (<https://blog.maestriajs.com/blog/angular/Angular-PRPL-Pattern/>).
- Perez, Antonio. 2020. «Todo sobre Caché con Service Workers (JavaScript)». *DIGITAL55*. Recuperado 23 de septiembre de 2020 (<https://www.digital55.com/desarrollo-tecnologia/cache-service-workers/>).

Kitsord, Educación. 2020. «Kitsord - Lengua de señas - Apps en Google Play». Recuperado 23 de septiembre de 2020 (https://play.google.com/store/apps/details?id=com.kitsord&hl=es_CO).

Ecured. s. f. «Lenguaje de señas - EcuRed». Recuperado 23 de septiembre de 2020 (https://www.ecured.cu/Lenguaje_de_se%C3%B1as).

MySQL. 2020. «MySQL». Recuperado 23 de septiembre de 2020 (<https://www.mysql.com/>).

Cleventy. 2020. «Tutorial y ejemplo de Spring Boot | Cleventy». Recuperado 23 de septiembre de 2020 (<https://cleventy.com/tutorial-spring-boot/>).

Angular.io. 2020. «Angular - FEATURES & BENEFITS». Recuperado 23 de septiembre de 2020 (<https://angular.io/features>).

Jmbeas. 2013. «Agile Inception». *José Manuel Beas*. Recuperado 23 de septiembre de 2020 (<https://jmbeas.es/guias/agile-inception/>).

Introducción a aplicaciones web progresivas—Aplicaciones Web Progresivas | MDN. (s. f.). Recuperado 22 de mayo de 2021, de https://developer.mozilla.org/es/docs/Web/Progressive_web_apps/Introduction