



UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE CUENCA

CARRERA DE COMPUTACIÓN

DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE UNA HERRAMIENTA WEB DE ANÁLISIS ESTADÍSTICO Y MONITOREO PARA EVALUAR LA ACCESIBILIDAD EN PÁGINAS WEB SEGÚN LOS CRITERIOS DE CONFORMIDAD DE LA WCAG 2.1

Trabajo de titulación previo a la obtención del título de Ingeniero en Ciencias de la Computación

AUTORES: DOUGLAS BRYAN SARMIENTO BASURTO

CESAR EDUARDO ZHIZHPON TACURI

TUTOR: ING. CRISTIAN FERNANDO TIMBI SISALIMA, MSC.

Cuenca - Ecuador

2022

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, Douglas Bryan Sarmiento Basurto con documento de identificación N° 1400668180 y Cesar Eduardo Zhizhpon Tacuri con documento de identificación N° 0105587919; manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Cuenca, 25 de julio del 2022

Atentamente,

Douglas Bryan Sarmiento Basurto

1400668180

Cesar Eduardo Zhizhpon Tacuri

0105587919

CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA

Nosotros, Douglas Bryan Sarmiento Basurto con documento de identificación N° 1400668180 y Cesar Eduardo Zhizhpon Tacuri con documento de identificación N° 0105587919, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto Técnico: “Diseño, desarrollo e implementación de una herramienta web de análisis estadístico y monitoreo para evaluar la accesibilidad en páginas web según los criterios de conformidad de la WCAG 2.1”, el cual ha sido desarrollado para optar por el título de: Ingeniero en Ciencias de la Computación, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, 25 de julio del 2022

Atentamente,

Douglas Bryan Sarmiento Basurto

1400668180

Cesar Eduardo Zhizhpon Tacuri

0105587919

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Cristian Fernando Timbi Sisalima con documento de identificación N° 0103709911, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE UNA HERRAMIENTA WEB DE ANÁLISIS ESTADÍSTICO Y MONITOREO PARA EVALUAR LA ACCESIBILIDAD EN PÁGINAS WEB SEGÚN LOS CRITERIOS DE CONFORMIDAD DE LA WCAG 2.1, realizado por Douglas Bryan Sarmiento Basurto con documento de identificación N° 1400668180 y por Cesar Eduardo Zhizhpon Tacuri con documento de identificación N° 0105587919, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Cuenca, 25 de julio del 2022

Atentamente,

Ing. Cristian Fernando Timbi Sisalima, MsC.

0103709911

Dedicatoria

Dedico el presente trabajo a mi familia, quienes, a través de sus consejos, su afecto y comprensión me han ayudado a ser una mejor versión de mí mismo. Quienes, a pesar de la distancia me han brindado el cariño y el apoyo necesario para motivarme seguir adelante a pesar de las dificultades.

A mi mami Lorena, mi ejemplo de superación, la persona que más me ha apoyado durante el transcurso de mi vida y quien me brindó las herramientas que me permitieron perseguir mis sueños sin contratiempos.

A mi mamita Josefina, por ser la primera en enseñarme a levantarme con más determinación después de cada fallo. Una mujer valiente que me ha hecho ver que la vida, aunque complicada, es hermosa.

A mi hermana Denisse, el motivo más reciente para esforzarme al máximo en todas las metas que me he propuesto. Ella, a través de sus tiernas palabras me recordaban a diario la razón por la que me encontraba lejos persiguiendo mis sueños.

Y a mi papá Nestor, por estar pendiente de mi bienestar y de mi futuro, e incentivarme a llevar un equilibrio entre estudio y descanso. Sus consejos me han ayudado a enfocarme en lo que soy capaz de hacer, mantenerme firme y con los pies en la tierra.

Douglas Bryan Sarmiento Basurto.

La presente tesis está dedicada a mis padres Yolanda, Saúl y mi abuelo Leonardo quienes con gran perseverancia y esfuerzo me han brindado los medios para cumplir con una meta más, y por brindarme el ejemplo de esfuerzo, apoyo emocional y la valentía necesaria para sobrellevar los problemas que acarrearón durante el desarrollo del presente trabajo.

A mis hermanas Tatiana y Yuliana por estar conmigo con su apoyo y cariño incondicional durante todo este proceso. A quienes aprecio mucho y quiero demostrar que el esfuerzo aplicado a algo honesto tendrá una recompensa.

A toda mi familia que, mediante sus consejos de vida, palabras de aliento y oraciones me han convertido en una mejor persona para tener el enfoque necesario, y llevar a cabo el desarrollo de este proyecto con las mejores intenciones y esfuerzo que este haya requerido.

Cesar Eduardo Zhizhpon Tacuri.

Agradecimiento

Agradezco principalmente a mi familia: Jaime, César, Edwin, Betty, Angelica y Elizabeth. Por comprender mi decisión de viajar lejos en busca de mis sueños y apoyarme de todas las formas posibles para no rendirme en el proceso. Sin su constante apoyo no estaría donde estoy hoy.

A mis profesores, en especial a: Gilberto, Roberto, Luis, Paola, Gabriel, Mauricio, Carlos, German, Andrea, Diego, Rodolfo, Vladimir, Jennifer, Gustavo, Walter y Pablo. Porque con paciencia, y a través de su experiencia y anécdotas nos enseñaron no solo para el ámbito profesional, sino para la vida en general. Gracias por hacer que el aprendizaje fuera más dinámico e interesante. Todos ellos, representan claramente los valores de la UPS.

A nuestros tutores, Cristian Timbi y Remigio Hurtado, por su acompañamiento durante el desarrollo de este proyecto. Gracias por brindarnos sus recomendaciones y su tiempo para poder resolver nuestras dudas e inquietudes, y así culminar con éxito esta etapa.

A mis amigos: Eduardo, Roberto, Bernardo, Robbyn, Pablo, John, Carlos, Florencio, Martin, Katherine, Juanca, Willan, Juanfer, Paul, Cachito, Pedro, Kevin, Diana, Jessica y Francisco. Gracias por compartir tantos momentos inolvidables en estos años, y hacerme sentir cómodo y seguro aún lejos de mi hogar. A todos ellos, de todo corazón, mis más sinceros agradecimientos.

Douglas Bryan Sarmiento Basurto.

Agradezco a mis padres Yolanda y Saúl que me han dado la oportunidad de formar parte de la Universidad Politécnica Salesiana y por ser los pilares fundamentales en mi desarrollo académico. A Leonardo, Tatiana, Yuliana, Yulian, César, Sofía, Lucía, Jorge y a mi familia que a pesar de la lejanía de una u otra forma me han acompañado durante el proceso de formación de la Carrera.

A nuestros tutores, Cristian Timbi y Remigio Hurtado, por habernos guiado en el desarrollo de este tema de tesis, que brindaron su tiempo y paciencia para resolvernos dudas y brindarnos su conocimiento para aplicar las mejores prácticas en este proyecto.

A los docentes de la Universidad Politécnica Salesiana, por compartirnos sus conocimientos con el espíritu salesiano, que nos brindaron sus palabras de aliento, y nos han regalado recompensas en pro al gran esfuerzo que conlleva el desarrollo de las materias de la Carrera.

A mis amigos: Bryan, Roberto, Bernardo, Pablo, Robbyn y a mis compañeros que nos brindaron su apoyo y que compartimos grandes momentos, que hicieron amena la formación académica, para convertirnos en grandes profesionales y llevar el espíritu salesiano para brindar nuestros servicios en apoyo a la sociedad.

Cesar Eduardo Zhizhpon Tacuri.

Resumen

El presente trabajo trata de la construcción de una herramienta web para el análisis de la accesibilidad en páginas web a través de dos módulos, que se soportan en el API del analizador web desarrollado dentro del Proyecto del Observatorio de Accesibilidad Web de la Universidad Politécnica Salesiana (UPS). Por un lado, el primer módulo se encarga del monitoreo continuo y automatizado de páginas web registradas para evaluar el nivel de cumplimiento de accesibilidad según los criterios de conformidad de la WCAG. El segundo módulo se encarga de recopilar los datos obtenidos a través del monitoreo para abstraer información relevante en base a filtros como: país de origen de la página web, la categoría de clasificación de este, así como rangos de fechas en donde se ha realizado el monitoreo.

La principal motivación de la realización de este trabajo es la de brindar un servicio, a quién lo requiera, que sirva como herramienta de apoyo para simplificar los procesos de evaluación y mejora de la accesibilidad en sitios web, y así garantizar un mayor alcance de la información en el entorno digital al disminuir las limitantes que pudieran existir en los mismos.

El desarrollo de la herramienta fue implementada a través de una arquitectura distribuida en cuatro capas. La primera capa, de presentación, fue desarrollada usando el *framework* Angular junto con PrimeNG y Highcharts para la visualización de gráficas y datos. La segunda capa, de negocio, fue desarrollada usando Jakarta EE para implementar los módulos descritos anteriormente. La tercera y cuarta capa, persistencia y datos, fueron implementadas a través de JPA y PostgreSQL como base de datos.

Palabras Clave: Accesibilidad Web, WCAG, Aplicación Web, Cómputo Paralelo, Angular, JavaEE, PostgreSQL, Dashboard.

Abstract

This work deals with the construction of a web tool for the analysis of accessibility in web pages through two modules, which are supported by the API of the web analyzer developed within the Web Accessibility Observatory Project of the Salesian Polytechnic University (UPS). On the one hand, the first module is responsible for the continuous and automated monitoring of registered web pages to assess the level of accessibility compliance according to the WCAG compliance criteria. The second module oversees compiling the data obtained through the monitoring to abstract relevant information based on filters such as: country of origin of the website, the classification category of the website, as well as date ranges where the monitoring has been performed.

The main motivation of this work is to provide a service, to whoever requires it, that serves as a support tool to simplify the processes of evaluation and improvement of accessibility on websites, and thus ensure a greater scope of information in the digital environment by reducing the limitations that may exist in them.

The development of the tool was implemented through a four-layer distributed architecture. The first layer, the presentation layer, was developed using the Angular framework together with PrimeNG and Highcharts for the visualisation of graphs and data. The second layer, business, was developed using Jakarta EE to implement the modules described above. The third and fourth layers, persistence, and data, were implemented using JPA and PostgreSQL as the database.

Keywords: Web Accessibility, WCAG, Web Application, Parallel Computing, Angular, JavaEE, PostgreSQL, Dashboard.

Índice de Contenido

| | | |
|------|--|----|
| I. | Introducción..... | 1 |
| II. | Problema..... | 2 |
| 1. | Antecedentes | 2 |
| 2. | Importancia y Alcance..... | 2 |
| 2.1. | Justificación..... | 2 |
| 2.2. | Grupo Objetivo..... | 3 |
| III. | Objetivos Generales y Específicos | 4 |
| 1. | Objetivo General | 4 |
| 2. | Objetivos Específicos | 4 |
| IV. | Revisión de la Literatura | 5 |
| 1. | Accesibilidad Web | 5 |
| 1.1. | Principios de Accesibilidad Web..... | 5 |
| 1.2. | Niveles de Conformidad..... | 5 |
| 1.3. | Proceso de Evaluación..... | 6 |
| 1.4. | Herramientas de Accesibilidad Web | 6 |
| 2. | Ingeniería de Software..... | 7 |
| 2.1. | Metodología en Cascada..... | 7 |
| 2.2. | Arquitectura de Microservicios | 8 |
| 2.3. | API RESTful | 9 |
| 3. | Análisis Estadístico | 9 |
| 4. | Cómputo Paralelo..... | 10 |
| 5. | Herramientas de Desarrollo..... | 10 |
| 5.1. | Angular | 11 |
| 5.2. | Highcharts Library | 11 |
| 5.3. | Jakarta Enterprise Edition..... | 11 |
| 5.4. | WildFly..... | 12 |
| V. | Marco Metodológico | 13 |
| 1. | Metodología de Desarrollo | 13 |
| 1.1. | Actividades por Objetivos | 13 |
| 2. | Propuesta de solución..... | 15 |
| 2.1. | Primera Fase: Investigación | 15 |
| 2.2. | Segunda Fase: Construcción del <i>Backend</i> | 15 |
| 2.3. | Tercera Fase: Construcción del <i>Frontend</i> | 17 |

| | | |
|-------|--|----|
| 2.4. | Cuarta Fase: Integración de Herramientas | 17 |
| VI. | Resultados | 18 |
| 1. | Especificación de Requerimientos | 18 |
| 1.1. | Requerimientos Funcionales | 18 |
| 1.2. | Requerimientos No Funcionales..... | 23 |
| 2. | Diseño..... | 24 |
| 2.1. | Arquitectura General | 24 |
| 2.2. | Prototipos..... | 25 |
| 2.3. | Diagrama de Clases | 29 |
| 3. | Codificación | 30 |
| 3.1. | Desarrollo del módulo de monitoreo | 30 |
| 3.2. | Desarrollo del módulo de análisis estadístico..... | 33 |
| 3.3. | Proceso de integración de herramientas de accesibilidad..... | 34 |
| 4. | Prueba y Verificación | 38 |
| 4.1. | Pruebas Funcionales | 38 |
| VII. | Cronograma | 48 |
| VIII. | Presupuesto..... | 50 |
| IX. | Conclusiones | 51 |
| X. | Recomendaciones | 52 |
| | Referencias Bibliográficas | 53 |
| | Anexos..... | 55 |

Índice de Tablas

| | |
|--|----|
| Tabla 1. Fases del desarrollo en cascada..... | 7 |
| Tabla 2. Actividades para el objetivo específico 1..... | 13 |
| Tabla 3. Actividades para el objetivo específico 2..... | 13 |
| Tabla 4. Actividades para el objetivo específico 3..... | 14 |
| Tabla 5. Actividades para el objetivo específico 4..... | 14 |
| Tabla 6. Actividades para el objetivo específico 5..... | 14 |
| Tabla 7. Requerimiento Funcional: RF-1, Análisis de Accesibilidad Web..... | 18 |
| Tabla 8. Requerimiento Funcional: RF-2, Evaluación en Paralelo | 18 |
| Tabla 9. Requerimiento Funcional: RF-3, Análisis de Contraste..... | 19 |
| Tabla 10. Requerimiento Funcional: RF-4, Dashboard..... | 19 |
| Tabla 11. Requerimiento Funcional: RF-5, Actualización Dinámica de Gráficas | 20 |
| Tabla 12. Requerimiento Funcional: RF-6, Monitoreo de Accesibilidad | 20 |
| Tabla 13. Requerimiento Funcional: RF-7, Análisis Estadístico | 21 |
| Tabla 14. Requerimiento Funcional: RF-8, Solicitud de Registro (Usuario)..... | 21 |
| Tabla 15. Requerimiento Funcional: RF-9, Solicitud de Registro (Administrador) | 22 |
| Tabla 16. Requerimiento Funcional: RF-10, Gestión de Países..... | 22 |
| Tabla 17. Requerimiento Funcional: RF-11, Gestión de Categorías..... | 23 |
| Tabla 18. Requerimiento Funcional: RF-12, Gestión de Páginas Web..... | 23 |
| Tabla 19. Pseudocódigo del monitoreo de la Accesibilidad Web | 31 |
| Tabla 20. Tecnologías implementadas para la capa de presentación | 35 |
| Tabla 21. Tecnologías implementadas para la capa de negocio..... | 36 |
| Tabla 22. Tecnologías implementadas para la capa de persistencia..... | 37 |
| Tabla 23. Tecnologías implementadas para la capa de datos | 37 |
| Tabla 24. Prueba Funcional 1: Análisis de Accesibilidad Web | 38 |
| Tabla 25. Prueba Funcional 2: Dashboard | 39 |
| Tabla 26. Prueba Funcional 3: Actualización dinámica de gráficas (Países)..... | 40 |
| Tabla 27. Prueba Funcional 4: Solicitud de Registro (Usuario)..... | 41 |
| Tabla 28. Prueba Funcional 5: Solicitud de Registro (Administrador) | 43 |
| Tabla 29. Prueba Funcional 6: Gestión de Páginas Web (Creación)..... | 44 |
| Tabla 30. Prueba Funcional 7: Análisis de Contraste..... | 45 |
| Tabla 31. Cronograma de actividades del proyecto | 48 |
| Tabla 32. Resumen de horas por estudiante | 49 |
| Tabla 33. Presupuesto para la realización del proyecto | 50 |

Índice de Figuras

| | |
|---|----|
| Figura 1. Etapas de la WCAG-EM..... | 6 |
| Figura 2. Comparación entre arquitectura monolítica y de microservicios..... | 8 |
| Figura 3. Esquema del módulo de monitoreo..... | 16 |
| Figura 4. Esquema del módulo estadístico..... | 16 |
| Figura 5. Arquitectura de la Herramienta Web..... | 24 |
| Figura 6. Prototipo: Dashboard de la Herramienta Web..... | 26 |
| Figura 7. Prototipo: Solicitud de Registro (Usuario)..... | 27 |
| Figura 8. Prototipo: Solicitud de Registro – General (Administrador)..... | 28 |
| Figura 9. Prototipo: Solicitud de Registro – Visualización (Administrador)..... | 28 |
| Figura 10. Página de Inicio del OAW Ecuador..... | 29 |
| Figura 11. Diagrama de clases para los módulos de monitoreo y análisis estadístico..... | 30 |

Índice de Anexos

| | |
|--|----|
| Anexo 1. Prototipo: Gestión de países | 55 |
| Anexo 2. Prototipo: Gestión de páginas | 56 |
| Anexo 3. Prototipo: Previsualización de correo electrónico de respuesta (Administrador)..... | 57 |
| Anexo 4. Prototipo: Ejemplo de correo electrónico de respuesta (Usuario) | 58 |
| Anexo 5. Manual de despliegue de servicios – Creación de directorios | 59 |
| Anexo 6. Manual de despliegue de servicios – Base de datos (PostgreSQL: 10.13) | 60 |
| Anexo 7. Manual de despliegue de servicios – Servidor de aplicaciones (WildFly: 26.1.1) | 61 |
| Anexo 8. Manual de despliegue de servicios – Servidor web (Nginx: 1.16.1) | 68 |
| Anexo 9. Manual de despliegue de servicios – Backend (Java: 1.8)..... | 69 |
| Anexo 10. Manual de despliegue de servicios – Frontend (Angular: 11.0.0, PrimeNG: 11.2.0)..... | 70 |
| Anexo 11. Manual de despliegue de servicios – Módulo de Análisis de Conformidad..... | 71 |
| Anexo 12. Manual de despliegue de servicios – Módulo para el Análisis de Contraste..... | 72 |

I. Introducción

En la actualidad, el constante cambio tecnológico y el continuo crecimiento de la transformación digital hace evidente el reto de las organizaciones y empresas para ofrecer servicios e información en la web que realmente llegue a todo el mundo sin importar sus condiciones.

De esta manera, la implementación de accesibilidad en sitios web es de vital importancia para asegurar que se garantiza el derecho a la información. Ya sean instituciones educativas, organizaciones gubernamentales o empresas que ofrezcan servicios, la importancia que se le debería dar al desarrollo de sitios web accesibles es alta. Sin embargo, la falta de conocimiento o importancia en estos temas impide que la información sea accesible para cualquier usuario, y es que la accesibilidad web no debería ser vista como una pérdida de tiempo o un proceso complejo.

Por lo tanto, este trabajo plantea una idea que pretende cambiar la visión que se le da a la accesibilidad web. La propuesta de este trabajo es la del desarrollo de una herramienta que le permita al usuario monitorear la accesibilidad de un sitio web, por medio del registro de sus páginas web para que sean incluidas en una evaluación periódica que analice qué tanto cumple un conjunto de estándares de accesibilidad. De esta manera se establece un punto de partida para que se piense en un plan de mejora que le permita alcanzar resultados óptimos en cuanto a la accesibilidad se refiere. En otras palabras, se trata de una herramienta de análisis automático que complementa la evaluación de la accesibilidad de un experto.

Es por eso por lo que en el presente trabajo se detalla acerca de esta problemática en el contexto actual, así como la importancia que tiene buscar soluciones para reducir el impacto que esta genera. Así mismo, se detallará el proceso que se ha seguido, y las tecnologías que se han utilizado para obtener una herramienta que permita disminuir la brecha de acceso a la información disponible en la web.

II. Problema

En esta sección se detalla el problema central sobre el que se basa este proyecto, así como la importancia del desarrollo y la implementación de este tipo de soluciones en el contexto actual.

1. Antecedentes

La transformación digital es un fenómeno que crece cada año, cambiando así la manera en la que las personas realizan diferentes actividades de su día a día, lo que a su vez obliga a las empresas e instituciones privadas y públicas a buscar maneras para adaptarse al continuo cambio digital (Jiménez, 2020). Este cambio se volvió más evidente los últimos años debido al confinamiento causado por la emergencia sanitaria (COVID-19), en donde se presentaron grandes retos para diferentes áreas como: la educación, salud, gestión gubernamental, entre otros. Como resultado, se han experimentado grandes avances en las TICs, así como un incremento en la relevancia del área de desarrollo web.

Con este cambio se hace notorio un problema, y es la atención e importancia que se le da a la accesibilidad web, un aspecto importante para asegurar que el contenido esté disponible y al alcance de todos sin importar sus discapacidades. Por esta necesidad, en varios países, se establecen leyes en las cuales se plantean algunos requerimientos mínimos de accesibilidad que deben alcanzar determinados sitios web desarrollados o disponibles en el país. En Ecuador, por ejemplo, según un reglamento técnico que entró en vigor en 2016, se estableció que para el mes de agosto del 2020 los sitios web de servicios públicos deberían cumplir con los requerimientos de accesibilidad nivel AA, por lo menos, según los criterios de la WCAG 2.0 (Luján Mora, 2022).

Sin embargo, según un estudio realizado en Ecuador, se tomaron a 44 universidades a las cuales se les evaluaron sus páginas web mediante herramientas *on-line* (Acosta & Luján-Mora, 2017), y se logró determinar que no cumplían con los criterios mínimos de accesibilidad según las legislaciones ecuatorianas. Por otro lado, en el estudio realizado a varias páginas web gubernamentales del Ecuador (Macías et al., 2020), se evidencia un escaso cumplimiento de las normas técnicas vigentes referentes a la accesibilidad web. Este incumplimiento de legislaciones limita el alcance que tienen los servicios y la información disponible en la web generando así una desigualdad digital.

2. Importancia y Alcance

2.1. Justificación

Una verdadera inclusión digital solo es posible si la accesibilidad web, y por lo tanto las legislaciones que las promueven, son cumplidas. Además, es necesario aclarar que esta inclusión no solo presenta un beneficio a personas que posean alguna discapacidad, que, según estadísticas en Ecuador, se alcanzan los 471,205 de individuos (CONADIS, 2022); si no, a todas

las personas independientemente de su edad, nacionalidad, género o condición social y económica.

Para alcanzar la inclusión digital es necesario realizar evaluaciones del grado de accesibilidad de las páginas web. Varios estudios (Timbi-Sisalima, Amor, et al., 2016; Yachimba & Aracelly, 2022) han recopilado algunas de las herramientas web disponibles que permiten automatizar la evaluación de páginas web según diferentes guías de accesibilidad, indicar en qué lugares se incumplen los criterios e incluso brindar recomendaciones de cómo alcanzar un mejor nivel de conformidad. Sin embargo, estas herramientas no ofrecen funcionalidades que permitan al evaluador realizar un control continuo acerca de las evaluaciones que realiza.

Por esta razón, en este trabajo se propone la construcción de una herramienta web que integre la evaluación y monitorización de páginas web, como un recurso que apoye a desarrolladores y evaluadores a reducir tiempo y esfuerzo en los procesos de evaluación al brindar un control programado y automatizado de las evaluaciones que realicen. De esta manera estos profesionales podrían elaborar planes de mejoramiento continuo centrado en aspectos de accesibilidad y así cumplir con las legislaciones vigentes que aseguren la inclusión digital.

2.2. Grupo Objetivo

La herramienta web podrá ser utilizada por cualquier individuo que tenga interés en temas relacionados con la accesibilidad web, mayormente profesionales con conocimientos técnicos, que necesiten de una herramienta de apoyo para realizar un seguimiento y control de accesibilidad en determinadas páginas web. Además, puede servir como un recurso para investigadores que tengan relación con el Grupo de Investigación de Inteligencia Artificial y Tecnologías de Asistencia (GI-IATA), los cuales tengan líneas de investigación relacionadas con temas de Accesibilidad Web.

III. Objetivos Generales y Específicos

En esta sección se detallan los objetivos que se pretenden alcanzar con el desarrollo de este proyecto.

1. Objetivo General

Construir una herramienta web de análisis estadístico y monitoreo para evaluar la accesibilidad en páginas web según los criterios de conformidad de la WCAG 2.1.

2. Objetivos Específicos

- **OE1:** Estudiar los fundamentos del proceso de evaluación de accesibilidad y sus herramientas de apoyo.
- **OE2:** Diseñar y desarrollar un módulo de monitoreo de la accesibilidad de sitios web.
- **OE3:** Diseñar un módulo de análisis estadístico de la accesibilidad de sitios web.
- **OE4:** Integrar la herramienta de evaluación de accesibilidad y la herramienta de análisis de contraste de la UPS y los módulos desarrollados en una nueva aplicación web con una interfaz usable y accesible.
- **OE5:** Diseñar y ejecutar un plan de pruebas para validar el funcionamiento de la aplicación desarrollada.

IV. Revisión de la Literatura

En esta sección se detallan los conceptos más relevantes acerca de las tecnologías y metodologías que se utilizaron para el desarrollo de este proyecto.

1. Accesibilidad Web

De acuerdo con el Artículo 4, Numeral 8, de la Ley Orgánica de Discapacidades, acuerda que, las tecnologías y sistemas orientados al manejo de información y las comunicaciones deben garantizar la accesibilidad a todas las personas con discapacidad. Esto quiere decir que los desarrolladores deben implementar medidas necesarias para que las funcionalidades de los sitios web puedan ser accedidas por usuarios con discapacidad (Asamblea Nacional República del Ecuador, 2012).

1.1. Principios de Accesibilidad Web

Hay cuatro principios de accesibilidad con las bases necesarias para que el contenido en las páginas web puedan cumplir con el acceso a su contenido (W3C, 2018). Por lo tanto, los componentes y la información que componen a la interfaz del usuario deben cumplir con las siguientes características:

- **Perceptible:** La información que se presenta debe ser percibida por los usuarios para todos sus sentidos.
- **Operable:** La interfaz y navegación debe permitir ser operada por los usuarios, donde todas las interacciones deben ser capaces de realizarse.
- **Comprensible:** El funcionamiento de la interfaz junto con la información que contiene debe presentar la capacidad de ser comprensible para los usuarios.
- **Robusto:** El contenido presentado debe de ser sólido a tal punto que las tecnologías de asistencia puedan interpretarlo de manera correcta.

1.2. Niveles de Conformidad

La satisfacción de los criterios estipulados en WCAG 2.1 se da al cumplir con los criterios de conformidad establecidos. Para ello, existen cinco requisitos que, al ser cumplidos, el contenido puede ser considerado como “conforme” (W3C, 2018). Para ello, el primer el primer requisito consta de niveles de conformidad, donde cada nivel se enfoca a necesidades de distintos grupos de circunstancias y usuarios.

Los grados de conformidad son tres: A, AA y AAA. En el primer nivel de conformidad se evalúan 25 criterios; 13 criterios en el segundo, y 23 criterios para el tercero (W3C, 2018). Cabe resaltar, que para alcanzar un nivel se deben satisfacer todos los criterios del nivel anterior.

1.3. Proceso de Evaluación

Para el proceso de evaluación se tienen en cuenta algunas actividades que no necesariamente deben ser realizadas secuencialmente. Además, estas pueden variar dependiendo del tipo de página web a evaluarse, el objetivo para evaluar y el enfoque que tenga el evaluador. En ese contexto, se puede mencionar a WCAG-EM, una metodología que consta de cinco etapas (Figura 1), la cual guía y orienta al evaluador a seguir un conjunto de buenas prácticas que faciliten el proceso de evaluación, análisis y documentación de resultados (W3C, 2022).

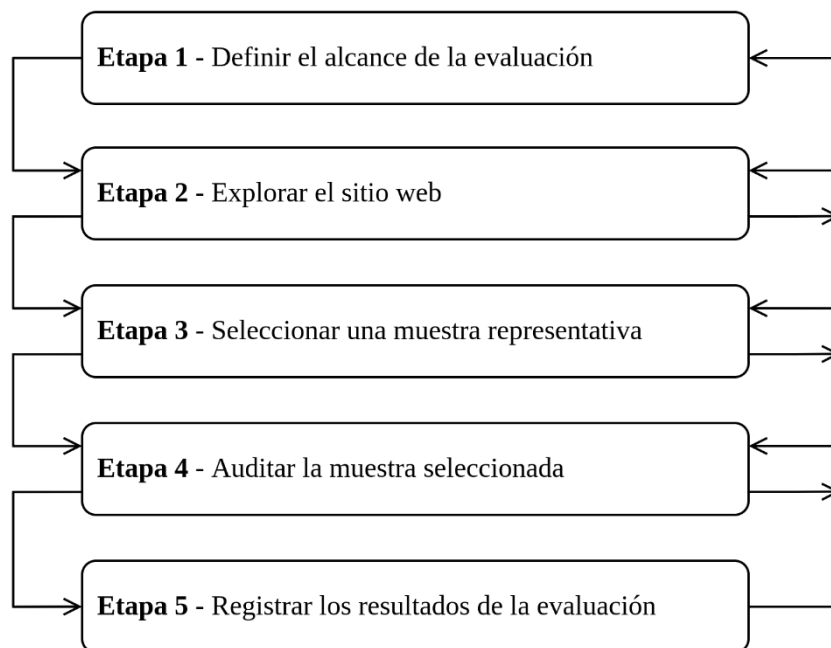


Figura 1. Etapas de la WCAG-EM
Fuente: Adaptado de la W3C (2022)

Por último, durante las etapas antes mencionadas se pueden usar herramientas complementarias para culminar, con mayor facilidad, el proceso de evaluación. De estas herramientas, algunas las proporciona la W3C, y otras son brindadas por terceros con diferentes funcionalidades y limitaciones.

1.4. Herramientas de Accesibilidad Web

En el estudio de Timbi-Sisalima, Amor, et al. (2016), se comparan varias herramientas, con licencias de uso no privativas, para el análisis de la accesibilidad web en términos de las guías que seguían para el análisis, las posibilidades de configurar el análisis, la existencia recomendaciones de mejora por parte de la herramienta, entre otros. De este estudio se pueden resaltar 4 herramientas que sobresalieron por sobre las demás: Tenon, AccessMonitor, TAW y OAW. Esta última herramienta mencionada es la herramienta seleccionada como base para el desarrollo de este proyecto.

La herramienta OWA (Observatorio Web de Accesibilidad) fue desarrollada por la Universidad Politécnica Salesiana en colaboración con el Consejo Nacional para la Igualdad de Discapacidades (CONADIS), la cual cuenta con un conjunto de servicios web que permite el análisis de manera remota mediante una API REST que verifica el cumplimiento de los criterios asociados al estándar WCAG (Timbi-Sisalima, Hilera, et al., 2016).

Los resultados de la evaluación de esta herramienta pueden ser obtenidos en formato JSON o XML, los cuales contienen una serie de elementos con los resultados del análisis realizado, junto con un resumen estadístico de los principios de accesibilidad del estándar WCAG.

2. Ingeniería de Software

La Ingeniería de Software abarca todo lo referente acerca de la construcción de software, desde los procesos, las metodologías, estándares, patrones de diseño, arquitecturas de software, entre otros. Todo esto para asegurar que la calidad del producto final sea acorde a las necesidades del cliente, y disminuir en lo posible errores y fallas durante el desarrollo y el posterior despliegue y mantenimiento.

2.1. Metodología en Cascada

Esta metodología de desarrollo sigue un enfoque secuencial de la construcción de software, donde se identifican las fases (Tabla 1) que se deben cumplir de manera ordenada durante el desarrollo de un proyecto de software. Como concepto, González González & Calero Castañeda (2019), afirman que es “un enfoque sistemático y secuencial, disciplinado y basado en análisis, diseño, pruebas y mantenimiento” (p. 3).

Tabla 1. Fases del desarrollo en cascada

| Fases | Descripción |
|-----------------------------------|---|
| Ingeniería y análisis del sistema | Se establecen las funcionalidades y requisitos actuales del sistema |
| Análisis de los requisitos | Se potencia el rendimiento, las funciones e interfaces |
| Diseño | Se selecciona la arquitectura y se estructuran los datos |
| Codificación | Se desarrollan los diseños que fueron establecidos |
| Prueba y verificación | Se evalúan las funcionalidades para verificar si alcanzan con los requerimientos establecidos |
| Mantenimiento | Se solucionan errores, se optimizan y ajustan funcionalidades en base a nuevos requerimientos |

2.2. Arquitectura de Microservicios

Para el desarrollo de sistemas, tradicionalmente se ha utilizado una arquitectura monolítica. En este tipo de arquitectura, todas las funcionalidades se encuentran establecidas dentro de un solo componente. Por otro lado, la arquitectura de microservicios rompe este esquema separando a las funcionalidades en diferentes componentes independientes entre sí, pero que trabajan de manera conjunta (Figura 2).

Dentro del trabajo de Wan et al. (2020) se resaltan tres ventajas de elegir la arquitectura de microservicios: control de complejidad, independencia y alta escalabilidad. Estas ventajas ofrecen a las empresas y organizaciones la posibilidad de aumentar el alcance de sus servicios, según se requiera, sin invertir mucho tiempo y sin afectar la calidad de estos. De esta manera aseguran la satisfacción del cliente.

A continuación, se enumerarán las principales características de la arquitectura de microservicios:

- **Independencia:** Los componentes se orientan hacia una funcionalidad en específico.
- **Reusabilidad:** Cada componente puede ser usado para varios propósitos en diferentes partes de la aplicación según se requiera.
- **Escalabilidad:** Según crezcan las necesidades, se podrá aumentar fácilmente cada componente por separado.
- **Robustez:** Los errores de un componente no afectan a toda la aplicación, ni la obliga a detenerse.
- **Flexibilidad:** Cada componente puede ser desarrollado en diferentes lenguajes y/o implementado con diferentes herramientas.

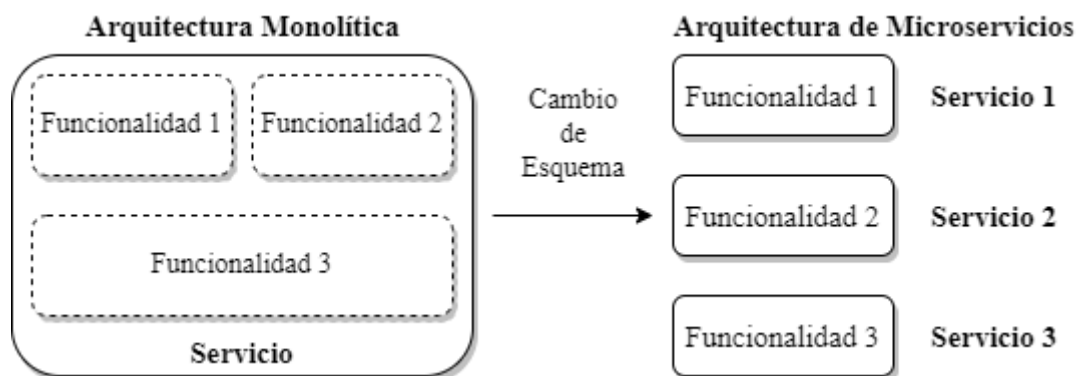


Figura 2. Comparación entre arquitectura monolítica y de microservicios

2.3. API RESTful

Interfaz de Programación de Aplicaciones Web (API Web) provee a los desarrolladores *frameworks* de terceros para aplicar a los servicios que se basan en el protocolo HTTP. Por otra parte, la Transferencia de Estado Representacional (REST) es el núcleo para realizar las comunicaciones con las aplicaciones externas. Por lo tanto, API RESTful es un patrón de diseño de software para brindar interoperabilidad entre distintos sistemas o aplicaciones, para lograr habilitar los servicios que funcionan en la web (Cruz et al., 2020).

A fin de realizar mejores diseños de los sistemas que utilicen este patrón hay que tener en cuenta los principios sobre los que se rige. Pautasso (2014), describe cinco principios, de la siguiente manera:

- **Direccionabilidad:** hace referencia a que todos los recursos deben de tener una identificación, la cual debe ser única y estable.
- **Interfaz uniforme:** hace referencia a que todos los recursos compartan e interactúen a través de una interfaz común. Esta interfaz deberá comprender los métodos del protocolo HTTP, por ejemplo: PUT, GET, POST, entre otros.
- **Servicios de interacciones sin estado:** hace referencia a que entre los servicios no se deberán de establecer comunicaciones entre ellas con estados. Esto quiere decir, que se mantendrán independientes unas de otras.
- **Servicios de mensajes autodescritivos:** hace referencia a que es necesario que los mensajes de solicitudes y respuestas incluyan metadatos acerca de las representaciones de los recursos con el fin que para otro servicio no sea ambiguo comprender y procesar la representación.
- **Hipermedia:** hace referencia a la posibilidad de que los recursos estén relacionados entre sí, en otras palabras, quiere decir que los recursos podrían tener incrustados dentro de sí mismos referencias hacia otros recursos (pp. 32 - 33).

3. Análisis Estadístico

El estudio de datos recolectados mediante un proceso determinado permite la obtención de medidas estadísticas. Esto tiene el objetivo de obtener información que sea relevante, y que permita el análisis, interpretación y uso de los resultados en la solución de determinados problemas.

Un término frecuente dentro de la estadística es el promedio, ya que permite encontrar la mayor concentración de medidas en un grupo. Por lo tanto, permite describir a un conjunto de datos para resaltar las características más importantes que permitan alguna intervención o mejora en una determinada área de interés (León Pirela & Pérez, 2019).

A su vez, estos conjuntos de datos pueden ser aprovechados por otras herramientas o tecnologías, como por ejemplo, para diseñar dashboards. Córdova Viera et al. (2021) definen a los dashboards como: “herramientas que permiten compartir, agrupar, centralizar y proporcionar una visualización gráfica de la información relevante de una organización, facilitando la toma de decisiones”.

Por tanto, es mediante la combinación del análisis estadístico y la utilización de dashboards, que se puede abstraer información realmente útil y aplicable dentro del contexto del problema que se esté estudiando. Esto debido a que la información se presenta de una manera más sencilla de interpretar y analizar, así como segmentar la información en base a variables de interés.

4. Cómputo Paralelo

La computación paralela pretende maximizar el uso de los recursos computacionales para enfrentar problemas complejos en plazos de tiempo relativamente cortos. Las soluciones pueden clasificarse mediante la Taxonomía de Flynn. Esta consta de dos enfoques principales: flujo de instrucciones, donde un sistema con una cantidad determinada de CPUs tiene la misma cantidad de flujos de instrucciones; y flujo de datos, donde un programa se ejecuta en diferentes listas con un flujo de datos específico (Zaccone, 2019).

Las instrucciones y listas de datos son consideradas independientes entre sí. Existen cuatro categorías que reflejan la ejecución de instrucciones y su respectivo manejo de los datos:

- **SISD:** Una instrucción que opera sobre un flujo de datos determinado.
- **MISD:** Ejecuta varias operaciones sobre los mismos datos.
- **SIMD:** Los procesadores trabajan de manera simultánea ejecutando la misma instrucción, pero, con diferentes flujos de datos.
- **MIMD:** Los procesadores trabajan de manera asíncrona, lo que permite una ejecución de diferentes instrucciones con diferentes datos.

5. Herramientas de Desarrollo

Actualmente existen varios lenguajes de programación y *frameworks* orientados al desarrollo de aplicaciones, cada uno con sus ventajas y desventajas dependiendo del objetivo, alcance, extensión y complejidad del proyecto de software a desarrollarse.

En el caso del desarrollo web, las tecnologías han evolucionado hasta el punto en el que algunas tareas son simplificadas y permiten al equipo de desarrollo centrarse mayormente en implementar las funcionalidades principales del sistema. A continuación, se detallarán las principales herramientas que se han seleccionado para el desarrollo de este proyecto.

5.1. Angular

Es un *framework* para desarrollo web *frontend* que es mantenido desde 2008 por Google. Aunque originalmente el lenguaje de programación que usaba era JavaScript, en sus últimas versiones se cambió por TypeScript, un lenguaje orientado a objetos basado en clases y de tipado fuerte (Wohlgethan, 2018), lo que permite una mayor flexibilidad y reducción considerable de la aparición de errores en la codificación.

Otras de las ventajas que se pueden resaltar de Angular son la facilidad de aprendizaje que tiene debido a su comunidad y la gran documentación detallada que provee, la prioridad que se le da a la estabilidad en cada lanzamiento y el soporte para desarrollar aplicaciones más accesibles y adaptables a diferentes regiones (Rangle.io, 2022).

Adicionalmente, Angular permite el uso de plantillas predefinidas como base para el desarrollo de aplicaciones. En ese contexto, tenemos PrimeNG, una colección de componentes y temas que permiten la personalización y configuración de la interfaz de usuario al incorporar diferentes estilos y animaciones (PrimeNG, 2022).

5.2. Highcharts Library

Highcharts es una librería de gráficas desarrollada en JavaScript y basada en SVG con soporte para navegadores antiguos. Con esta librería se pueden desarrollar gráficas para presentar de manera visual datos estadísticos para su posterior análisis.

Además, cuenta con fácil instalación y una documentación bien detallada en su página oficial, lo que facilita su implementación en cualquier aplicación web. Adicionalmente, las gráficas soportan accesibilidad interactiva para las personas con discapacidad, permitiendo a los desarrolladores personalizar esta interacción para garantizar el mayor entendimiento posible (Highcharts, 2022).

5.3. Jakarta Enterprise Edition

Jakarta Enterprise Edition se compone de varias interfaces para proveer una API de manera pública para los desarrolladores. También, estas interfaces están bien definidas y estandarizadas para garantizar la facilidad de implementación para cualquier propósito. Por lo que, esta edición pasa por un estricto procedimiento de pruebas para ser aceptadas y liberadas al ámbito empresarial (Saeed, 2020).

Jakarta Enterprise Edition 8 (JEE 8) mejora varias implementaciones comunes dentro del desarrollo de aplicaciones orientadas a la web respecto a su anterior versión llamada JEE 7. La versión 7 se concentró en la compatibilidad con HTML5, desarrollando el apartado de WebSockets y JSON. En base a ello, la versión 8 heredó las implementaciones mencionadas, enfocándose en modernizar las aplicaciones web para la nube. Además, de presentar mayor facilidad durante el proceso de desarrollo (Padmanabhan, 2018).

5.4. WildFly

WildFly es un servidor de aplicaciones de código abierto desarrollado en Java para aplicaciones del mismo lenguaje de programación. Además, cuenta con varios frameworks que reducen la cantidad de tiempo a invertir en el despliegue de una aplicación, así como facilitar la configuración y personalización de este (WildFly, 2022).

Entre sus frameworks, es relevante mencionar a Elytron, el cual está orientado a implementar seguridad en la aplicación principalmente a través de la autenticación. Según Docs WildFly (2021), este framework: “aporta a WildFly un único marco de seguridad unificado para todo el servidor de aplicaciones. Como marco único, se podrá utilizar tanto para configurar el acceso de gestión al servidor como para las aplicaciones desplegadas en el servidor”.

V. Marco Metodológico

En esta sección se detallan los aspectos relevantes acerca de la metodología de desarrollo seleccionada, así como la propuesta de solución de este proyecto.

1. Metodología de Desarrollo

Para desarrollar este proyecto se utilizará la metodología de desarrollo secuencial en cascada, la cual fue explicada en la sección anterior. Para ello, se respetará el proceso de “ciclo de vida” del software. Por lo tanto, se partirá desde el análisis de requerimientos y funcionalidades de las herramientas propuestas. Posteriormente se realizarán los diseños, diagramas y prototipos necesarios para orientar mejor el proceso de codificación. Finalmente se realizarán una serie de pruebas para validar el funcionamiento del sistema en base a los requerimientos planteados.

1.1. Actividades por Objetivos

Para apoyar el flujo de desarrollo del proyecto, así como el cumplimiento de los objetivos propuestos, se han definido algunas actividades para cada uno de los objetivos específicos, las cuales se enumerarán a continuación.

- OE1. Estudiar los fundamentos del proceso de evaluación de accesibilidad y sus herramientas de apoyo.

Tabla 2. Actividades para el objetivo específico 1.

| Código | Actividad |
|--------|---|
| OE1A1 | Estudiar los fundamentos del proceso de evaluación de la accesibilidad web. |
| OE1A2 | Estudiar la documentación existente de las herramientas de evaluación de accesibilidad web y análisis de contraste de la UPS. |

- OE2. Diseñar y desarrollar un módulo de monitoreo de la accesibilidad de sitios web.

Tabla 3. Actividades para el objetivo específico 2.

| Código | Actividad |
|--------|--|
| OE2A1 | Diseñar el diagrama de clases. |
| OE2A2 | Desarrollar un motor para procesamiento concurrente de evaluación de accesibilidad de páginas web. |
| OE2A3 | Desarrollar las funcionalidades del módulo de monitoreo. |

- OE3. Diseñar un módulo de análisis estadístico de la accesibilidad de sitios web.

Tabla 4. Actividades para el objetivo específico 3.

| Código | Actividad |
|---------------|---|
| OE3A1 | Desarrollar el módulo de análisis estadístico basado en los datos de accesibilidad de sitios web. |

- OE4. Integrar la herramienta de evaluación de accesibilidad y la herramienta de análisis de contraste de la UPS y los módulos desarrollados en una nueva aplicación web con una interfaz usable y accesible.

Tabla 5. Actividades para el objetivo específico 4.

| Código | Actividad |
|---------------|---|
| OE4A1 | Diseñar y desarrollar la interfaz de usuario de la aplicación web. |
| OE4A2 | Diseñar y desarrollar un componente de administración CRUD de los parámetros necesarios para la evaluación y monitoreo de sitios web. |
| OE4A3 | Desarrollar los servicios RESTful para comunicar la aplicación web con los módulos de monitoreo y análisis estadístico. |
| OE4A4 | Integrar las herramientas de evaluación de accesibilidad y análisis de contraste a la aplicación web. |
| OE4A5 | Documentar acerca del proceso de diseño, desarrollo e integración de los diferentes elementos que componen la aplicación web. |

- OE5. Diseñar y ejecutar un plan de pruebas para validar el funcionamiento de la aplicación desarrollada.

Tabla 6. Actividades para el objetivo específico 5.

| Código | Actividad |
|---------------|---|
| OE5A1 | Diseñar un plan de pruebas funcionales para la evaluación de la integración de las herramientas de evaluación de accesibilidad y analizador de contraste. |
| OE5A2 | Ejecutar los planes de pruebas. |

2. Propuesta de solución

La implementación del proyecto que se propone como solución al problema planteado puede abstraerse en cuatro fases: de investigación, de construcción del *backend*, de construcción del *frontend* y de integración de herramientas.

2.1. Primera Fase: Investigación

La primera fase aborda la investigación y obtención de conocimientos sobre los fundamentos e importancia de la accesibilidad web. Además, se indaga sobre el desarrollo de aplicaciones web con arquitecturas distribuidas y se analiza la documentación de recursos complementarios al desarrollo. Por último, se contempla el estudio de la documentación asociada a las herramientas de evaluación de accesibilidad y contraste de páginas web para conocer más acerca de sus funcionalidades básicas y tenerlas presentes para la posterior integración.

2.2. Segunda Fase: Construcción del *Backend*

La segunda fase aborda la construcción del *backend* de la herramienta, la cual contempla principalmente dos módulos funcionales:

- **Módulo de Monitoreo:** el cual dará seguimiento a los sitios web registrados en la base de datos (Figura 3). Para ello, se plantea realizar las solicitudes mediante concurrencia siguiendo la categoría SIMD de la Taxonomía de Flynn. Es decir, un conjunto de páginas registradas se repartirá de manera equitativa a un número determinado de hilos, los cuales realizarán la consulta a la API de evaluación de accesibilidad.

Posteriormente, una vez que la API haya retornado los resultados de todas las páginas que se mandaron a analizar, se integrarán los resultados para ser mandados a persistir en la base de datos.

- **Módulo Estadístico:** el cual consiste en un servicio que permitirá obtener datos estadísticos de las solicitudes realizadas en el módulo de monitoreo (Figura 4). Es decir, se realizan consultas a la base de datos de los resultados almacenados del monitoreo de los sitios web para crear un conjunto de datos estadísticos para su posterior análisis y representación visual.

Para ello, se utilizarán consultas que vayan de acuerdo con los atributos disponibles de cada análisis realizado. Por ejemplo, cada página pertenece a una categoría y país, por lo que se pueden obtener análisis estadísticos de uno o varios países, junto con una o varias categorías.

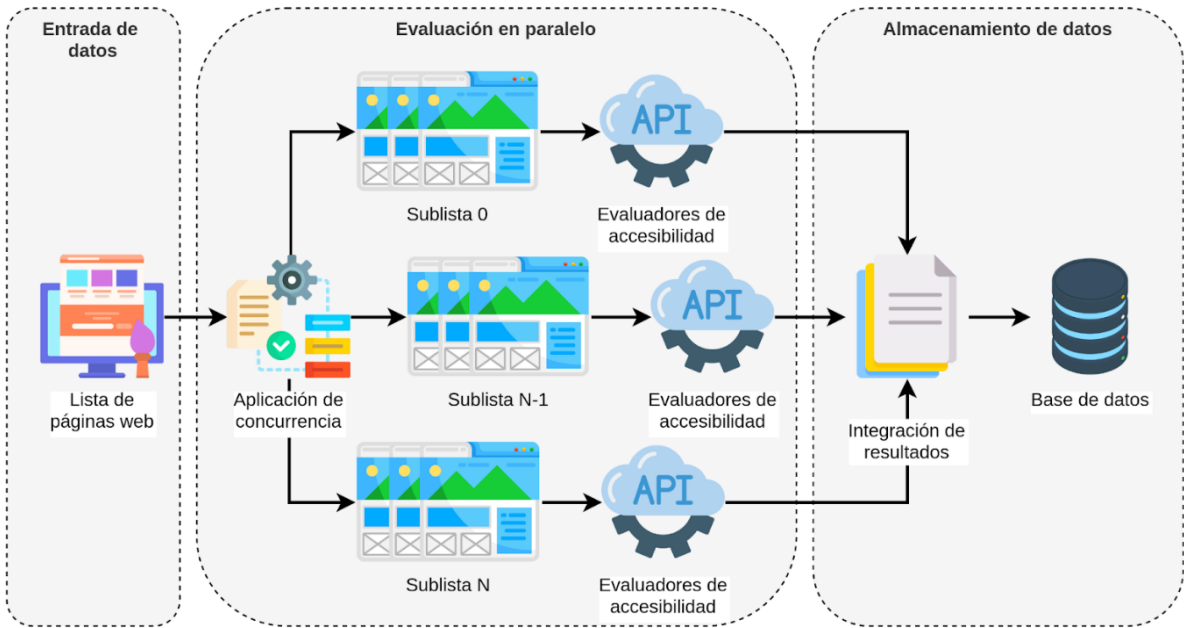


Figura 3. Esquema del módulo de monitoreo



Figura 4. Esquema del módulo estadístico

2.3. Tercera Fase: Construcción del *Frontend*

La tercera fase aborda la construcción del *frontend*. Para esto, se implementa una comunicación con el *backend* utilizando los servicios desarrollados en la segunda fase. Además, esta fase estará orientada principalmente para dos componentes.

El primer componente consta de la realización de un *dashboard*, donde se presentará el análisis estadístico mediante gráficas que muestren de manera visual los datos recolectados a través del monitoreo de las páginas web registradas para el análisis.

El segundo componente está orientado a la administración de los parámetros relacionados a los sitios web que formarán parte del monitoreo de accesibilidad web, teniendo como característica que los usuarios podrán realizar solicitudes de registro de las páginas web, los cuales deberán ser aprobados para poder comenzar con el proceso de análisis y monitoreo.

Ambos componentes, así como todos los otros componentes que conformarán la parte visual de la herramienta web, serán desarrollados con la ayuda de la plantilla Mirage de PrimeNG. Esta plantilla provee componentes básicos como: el *header*, el *footer*, tablas dinámicas, menús desplegables, entre otros. Esto reduce el tiempo y la dificultad a la hora de implementar los prototipos que serán desarrollados para esta fase.

Cabe resaltar, que aunque la plantilla de Mirage ya ofrezca componentes de gráficas para la visualización de datos, se usará la librería de Highcharts para este fin. La razón de esto es que esta librería es más flexible, y ofrece más posibilidades de personalización en comparación con las que ya vienen por defecto en PrimeNG.

2.4. Cuarta Fase: Integración de Herramientas

La cuarta fase aborda la integración de los módulos de monitoreo y análisis desarrollados en fases anteriores, con las herramientas de evaluación web de la UPS. Estas herramientas corresponden al análisis de criterios de conformidad y el análisis de contraste de páginas web. Para ello, se realizará una unificación homogénea junto con las nuevas tecnologías aplicadas en el desarrollo de los nuevos módulos en el *frontend*. De esta manera, el *backend* de los módulos anteriores pueden ser adaptados a favor de cubrir las necesidades y requisitos de la integración de las herramientas.

VI. Resultados

En esta sección se detallan los resultados obtenidos durante todo el proceso de desarrollo del proyecto propuesto.

1. Especificación de Requerimientos

1.1. Requerimientos Funcionales

Este tipo de requerimientos definen las funcionalidades que deberían estar implementadas en el software; en otras palabras, definen qué es lo que debe hacer el software. A continuación, se detallarán los más relevantes dentro del desarrollo:

Tabla 7. Requerimiento Funcional: RF-1, Análisis de Accesibilidad Web

| Nombre | Análisis de Accesibilidad Web | Código | RF-1 |
|--|-------------------------------|-----------|------|
| | | Prioridad | Alta |
| Descripción: La herramienta web debe permitir realizar el análisis de accesibilidad web según criterios de conformidad de la WCAG. | | | |
| Entrada: <ul style="list-style-type: none">• URL de la página web.• Resolución de pantalla.• Nivel de Conformidad (A, AA, AAA). | | | |
| Proceso: Se realiza la petición de análisis de accesibilidad, según los parámetros de entrada ingresados, a la API de la UPS. | | | |
| Salida: <ul style="list-style-type: none">• JSON con los resultados del análisis. | | | |

Tabla 8. Requerimiento Funcional: RF-2, Evaluación en Paralelo

| Nombre | Evaluación en Paralelo | Código | RF-2 |
|---|------------------------|-----------|------|
| | | Prioridad | Alta |
| Descripción: La evaluación de páginas web debe realizarse en paralelo a través de un motor. | | | |
| Entrada: <ul style="list-style-type: none">• Una lista de páginas web. | | | |
| Proceso: El motor separará de manera equitativa, según la cantidad de hilos establecidos, la lista. Posteriormente se realizarán las peticiones de análisis de manera concurrente. | | | |
| Salida: <ul style="list-style-type: none">• Los resultados de evaluación de la lista de páginas. | | | |

Tabla 9. Requerimiento Funcional: RF-3, Análisis de Contraste

| Nombre | Análisis de Contraste | Código | RF-3 |
|--|-----------------------|-----------|------|
| | | Prioridad | Alta |
| Descripción: La herramienta web debe permitir realizar el análisis de contraste de la página web. | | | |
| Entrada: <ul style="list-style-type: none"> • URL de la página web. • Correo electrónico. • Nivel de Conformidad (A, AA, AAA). • Resolución de pantalla (Ancho y Alto). • Factor de escala. • Tiempo de espera (s). | | | |
| Proceso: Se realiza la petición de análisis de contraste, según los parámetros de entrada ingresados, a la API de la UPS. Una vez esté finalizado el análisis se enviará un correo electrónico al usuario confirmando los resultados de este. | | | |
| Salida: <ul style="list-style-type: none"> • Correo electrónico con el enlace a los resultados de contraste. | | | |

Tabla 10. Requerimiento Funcional: RF-4, Dashboard

| Nombre | Dashboard | Código | RF-4 |
|--|-----------|-----------|------|
| | | Prioridad | Alta |
| Descripción: La herramienta web debe tener un dashboard que contenga una representación visual de los análisis realizados. | | | |
| Entrada: <ul style="list-style-type: none"> • Opción en pantalla (Acceder al Dashboard). • Resultados de los análisis. | | | |
| Proceso: Al ingresar a la página del Dashboard, se llamará a un servicio, el cual solicitará a través de consultas, los resultados de los análisis realizados. Una vez obtenidos, estos se utilizarán para ser mostrados de manera visual. | | | |
| Salida: <ul style="list-style-type: none"> • Gráficas relevantes de los resultados. • Tablas con opciones de filtrado. | | | |

Tabla 11. Requerimiento Funcional: RF-5, Actualización Dinámica de Gráficas

| Nombre | Actualización Dinámica de Gráficas | Código | RF-5 |
|---|------------------------------------|-----------|-------|
| | | Prioridad | Media |
| Descripción: Las gráficas realizadas deben actualizarse dinámicamente en base a parámetros como rangos de fecha, categorías y países de origen de las páginas web. | | | |
| Entrada: <ul style="list-style-type: none"> • Gráficas iniciales. • Parámetros de actualización. | | | |
| Proceso: Al modificar un parámetro, como la fecha de inicio y final, los servicios enviarán nuevas peticiones para hacer nuevas consultas para que se adapten a los nuevos parámetros de entrada. | | | |
| Salida: <ul style="list-style-type: none"> • Gráficas actualizadas. | | | |

Tabla 12. Requerimiento Funcional: RF-6, Monitoreo de Accesibilidad

| Nombre | Monitoreo de Accesibilidad | Código | RF-6 |
|--|----------------------------|-----------|------|
| | | Prioridad | Alta |
| Descripción: Las páginas registradas serán analizadas de manera automatizada para poder realizar un monitoreo constante. | | | |
| Entrada: <ul style="list-style-type: none"> • Páginas registradas. • Opciones predeterminadas de monitoreo automatizado. | | | |
| Proceso: Según las opciones de monitoreo, configuradas por los administradores de la Herramienta Web, cada determinado tiempo las páginas registradas se mandarán a analizar nuevamente para evaluar si hay o no mejoras en estas. | | | |
| Salida: <ul style="list-style-type: none"> • Resultados de los análisis actualizados. | | | |

Tabla 13. Requerimiento Funcional: RF-7, Análisis Estadístico

| Nombre | Análisis Estadístico | Código | RF-7 |
|--|----------------------|-----------|------|
| | | Prioridad | Alta |
| Descripción: La herramienta web, permitirá tomar los datos de los resultados de análisis de accesibilidad, para poder realizar un análisis estadístico, el cual aporte con datos relevantes para el usuario. | | | |
| Entrada: <ul style="list-style-type: none"> Resultados de análisis de accesibilidad. | | | |
| Proceso: Los resultados de los análisis serán analizados en función de los países, categorías, y en base a los 4 principios de accesibilidad (Perceptible, Operable, Robusto, Comprensible), para posteriormente generar datos de resumen que se presentarán al usuario. | | | |
| Salida: <ul style="list-style-type: none"> Datos estadísticos sobre la accesibilidad web según los criterios de la WCAG. | | | |

Tabla 14. Requerimiento Funcional: RF-8, Solicitud de Registro (Usuario)

| Nombre | Solicitud de Registro – Usuario | Código | RF-8 |
|--|------------------------------------|-----------|-------|
| | | Prioridad | Media |
| Descripción: La herramienta web, permitirá a los usuarios realizar un registro de las páginas web para poder hacer el monitoreo automático. | | | |
| Entrada: <ul style="list-style-type: none"> País y Categoría. Título. Página Principal. Descripción (Opcional). Páginas adicionales (Opcional). Correos adicionales para notificación (Opcional). | | | |
| Proceso: El usuario rellenará los campos de la solicitud para registrar las páginas que desea que se monitoreen. En la misma, se verificará que estas páginas sean válidas, y de serlas, se generará la consulta para la revisión del administrador. | | | |
| Salida: <ul style="list-style-type: none"> Solicitud registrada en espera de revisión. | | | |

Tabla 15. Requerimiento Funcional: RF-9, Solicitud de Registro (Administrador)

| Nombre | Solicitud de Registro – Administrador | Código | RF-9 |
|--|--|-----------|-------|
| | | Prioridad | Media |
| Descripción: La herramienta web, permitirá a los administradores realizar la revisión de registros de páginas web para aprobar o denegar. | | | |
| Entrada: <ul style="list-style-type: none"> Solicitud de usuario. | | | |
| Proceso: El administrador revisará la solicitud registrada, este en base de lo que crea conveniente podrá aceptarla o denegarla. En cualquier caso, se enviará un correo electrónico al usuario indicándole el cambio de estado de la solicitud. | | | |
| Salida: <ul style="list-style-type: none"> Solicitud aprobada/denegada. Correo electrónico con la respuesta de la solicitud. | | | |

Tabla 16. Requerimiento Funcional: RF-10, Gestión de Países

| Nombre | Gestión de Países | Código | RF-10 |
|---|-------------------|-----------|-------|
| | | Prioridad | Baja |
| Descripción: La herramienta web, en su parte visual, debe implementar la gestión de países para el administrador. | | | |
| Entrada: <ul style="list-style-type: none"> Lista de los países registrados. | | | |
| Proceso: El administrador podrá editar, modificar, eliminar o crear más países. | | | |
| Salida: <ul style="list-style-type: none"> Lista de países actualizadas. | | | |

Tabla 17. Requerimiento Funcional: RF-11, Gestión de Categorías

| Nombre | Gestión de Categorías | Código | RF-11 |
|---|-----------------------|-----------|-------|
| | | Prioridad | Baja |
| Descripción: La herramienta web, en su parte visual, debe implementar la gestión de categorías para el administrador. | | | |
| Entrada: <ul style="list-style-type: none"> • Lista de las categorías registradas. | | | |
| Proceso: El administrador podrá editar, modificar, eliminar o crear más categorías. | | | |
| Salida: <ul style="list-style-type: none"> • Lista de categorías actualizadas. | | | |

Tabla 18. Requerimiento Funcional: RF-12, Gestión de Páginas Web

| Nombre | Gestión de Páginas Web | Código | RF-12 |
|--|------------------------|-----------|-------|
| | | Prioridad | Baja |
| Descripción: La herramienta web, en su parte visual, debe implementar la gestión de las páginas web registradas para el administrador. | | | |
| Entrada: <ul style="list-style-type: none"> • Lista de páginas web registradas. | | | |
| Proceso: El administrador podrá editar, modificar, eliminar o crear más páginas web para el registro. | | | |
| Salida: <ul style="list-style-type: none"> • Lista de páginas web actualizadas. | | | |

1.2. Requerimientos No Funcionales

Este tipo de requerimientos definen las funcionalidades que se relacionan con el diseño, rendimiento, disponibilidad, seguridad, entre otros. A continuación, se enumerarán los más relevantes dentro del desarrollo:

- [R-NF1] La herramienta web, en su parte visual, deberá tener un diseño *responsive*. Esto quiere decir que el contenido se podrá adaptar a varios tamaños de pantallas.
- [R-NF2] La herramienta web, en su parte visual, deberá tener un diseño amigable con el usuario. Esto quiere decir que deberá evitar el uso excesivo de colores, menús, o elementos complejos; en su lugar, deberá ser lo más minimalista posible.

- [R-NF3] La herramienta web, en su parte visual, deberá contar con un apartado en donde se detalle la información relacionada con la herramienta, así como la información de contacto.
- [R-NF4] La herramienta web, en su parte visual, deberá contar con un apartado en donde se detallen los términos y condiciones de uso de la esta.
- [R-NF5] Las funcionalidades con las que interactúe el usuario final deberán tener una retroalimentación fácilmente entendible. Esto quiere decir que no deberá ser técnica o muy extensa.
- [R-NF6] El usuario final, que realice una solicitud de registro, podrá elegir si los resultados del análisis y monitoreo se harán públicos o no.
- [R-NF7] El ingreso de parámetros como categorías y países, podrá hacerse de manera individual, o grupal.

2. Diseño

2.1. Arquitectura General

La arquitectura mediante la cual los componentes interactuarán entre ellos, de acuerdo con la propuesta de solución, se va a estructurar en cuatro capas: capa de presentación, capa de negocio (con microservicios), capa de persistencia y capa de datos (Figura 5).

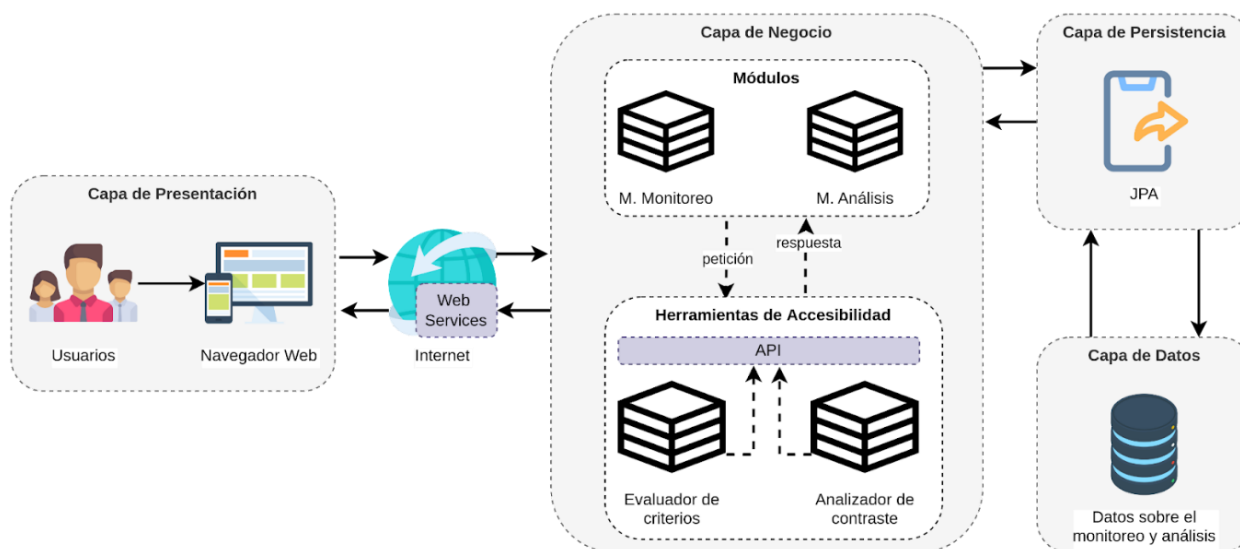


Figura 5. Arquitectura de la Herramienta Web

A continuación, se detallará brevemente cómo se compone cada una de las capas establecidas:

- **Capa de presentación:** esta capa se refiere a la parte visual de la herramienta, donde los usuarios pueden acceder e interactuar con las funcionalidades de esta, mediante el consumo de los servicios que proporciona la capa de negocio. Por tanto, esta capa se conforma de una aplicación web con un diseño que se adapte a la mayoría de los tamaños de pantallas (Responsive Web Design), permitiendo así el acceso desde un computador o dispositivo móvil. Además, contendrá toda la información relacionada con la herramienta y la información de contacto.
- **Capa de negocio:** en esta capa se conforman los módulos desarrollados y los servicios API ya existentes. Por un lado, el módulo de monitoreo se encargará de realizar el análisis de accesibilidad de manera periódica con un cronograma configurado a través de solicitudes a la API de análisis. Por otro lado, el segundo módulo se encargará de generar un análisis estadístico que puede ser accedido mediante servicios web para ser consumidos por la capa de presentación.
- **Capa de persistencia:** la capa de persistencia es un recurso para la estructura de datos relacionales. Lo que permite leer, crear, actualizar y eliminar instancias en la base de datos. De esta manera, permite acceder y manipular aquellas instancias o registros almacenados. Esto se logra mediante la API de persistencia de Java (JPA), ya que trabaja por encima de la conectividad a bases de datos de Java (JDBC). Además, esta API permite la administración de entidades (CRUD), incluye un lenguaje de consultas (JPQL) para la recuperación de datos, entre otras funcionalidades.
- **Capa de datos:** esta capa provee el servicio de almacenamiento de datos relacionales. Las instancias que se almacenan en este apartado son los sitios web, así como la relación que tienen con países y categorías. De igual forma, se almacenan los resultados del monitoreo de accesibilidad de las páginas web registradas. Finalmente, se proporciona el acceso a los conjuntos de datos necesarios para la generación del análisis estadístico.

2.2. Prototipos

Los prototipos, son una vista previa de lo que podría ser el software por desarrollarse. Dan una perspectiva al usuario de cómo podría verse el producto terminado, por lo tanto, puede no ser exactamente como el producto final. De igual forma, cabe resaltar que los prototipos pueden ser de baja, media y alta fidelidad dependiendo de la calidad de representación visual y funcional que transmiten al cliente.

Para este proyecto, se realizaron prototipos de baja fidelidad a través del software Balsamiq Cloud. A continuación, se presentarán los más relevantes, empezando por el prototipo del Dashboard (Figura 6).

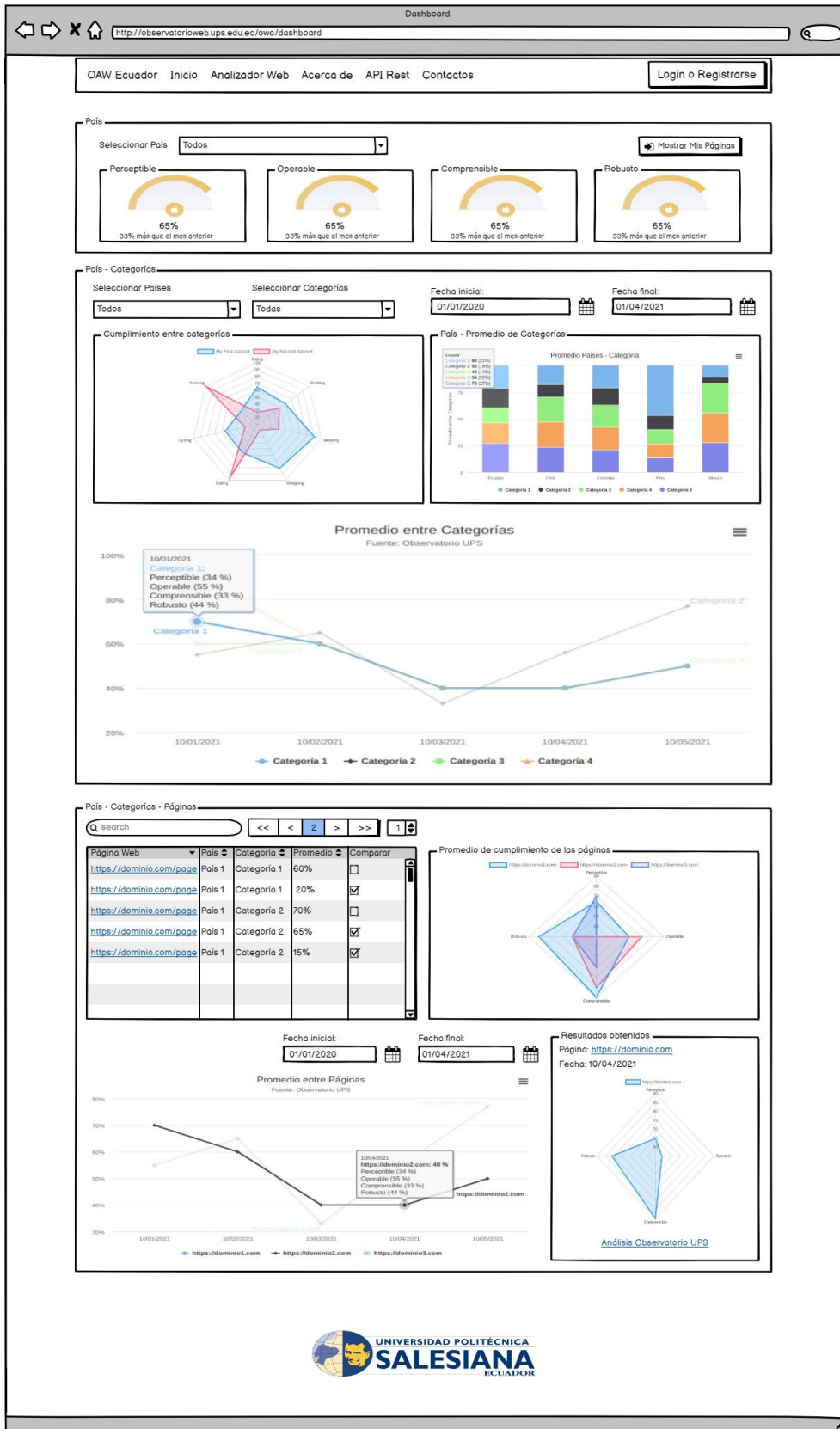


Figura 6. Prototipo: Dashboard de la Herramienta Web

Además, se mostrarán los prototipos realizados para la solicitud de registro, tanto desde la perspectiva del usuario (Figura 7), como desde de la perspectiva del administrador (Figura 8, Figura 9). Finalmente, otros prototipos pueden visualizarse en la sección Anexos (Anexo 1 - Anexo 4).

The screenshot shows a web browser window with the address bar containing `http://observatorioweb.ups.edu.ec/owa/user/sites/request`. The page title is "Registrar Páginas". The navigation menu includes "OAW Ecuador", "Inicio", "Analizador Web", "Acerca de", "API Rest", "Contactos", and a "Cerrar Sesión" button. The main heading is "Registrar Mis Páginas".

On the left, there is a "Menú" sidebar with items: "Item One", "Item Two", "Registrar mis Páginas" (highlighted), and "Item Four".

The registration form contains the following fields and options:

- País:** Dropdown menu with "Ecuador" selected.
- Categoría:** Dropdown menu with "Gubernamentales" selected.
- Detectado automáticamente:** "Ecuador"
- Título:** Text input field with "Servicio de Rentas Internas".
- Página Principal:** Text input field with `https://www.sri.gob.ec/web/intersri/home`.
- Páginas Adicionales (Opcional):** Text area containing:
 - `https://www.sri.gob.ec/web/intersri/biblioteca`
 - `https://www.sri.gob.ec/web/intersri/noticias7`
 - `https://www.sri.gob.ec/web/intersri/transparencia3`
- Descripción (Opcional):** Text area with "Páginas públicas pertenecientes al Servicio de Rentas Internas del Ecuador." and a character count of "70/140".
- Correos Adicionales para Notificar (Opcional):** Text area with `czhiz@est.ups.edu.ec; juanperez@gmail.com; marialopez@gmail.com`.
- He leído y acepto los [términos y condiciones](#).
- Deseo mostrar mis páginas con el observatorio UPS.

Buttons for "Cancelar" and "Enviar" are located at the bottom right of the form.

At the bottom of the page is the logo for "UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR".

Figura 7. Prototipo: Solicitud de Registro (Usuario)



Figura 8. Prototipo: Solicitud de Registro – General (Administrador)

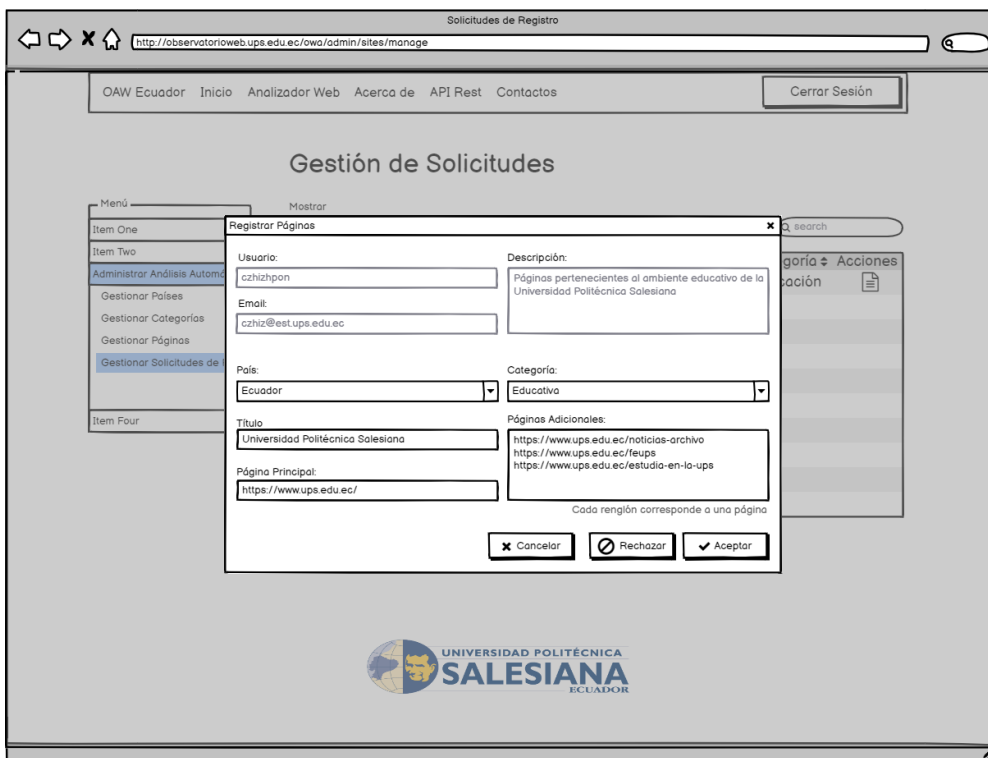


Figura 9. Prototipo: Solicitud de Registro – Visualización (Administrador)

Finalmente, cabe resaltar que no solo se utilizaron los prototipos realizados como referencia para lo que sería el producto final. También se utilizó el diseño utilizado para el sitio web del “Observatorio de Accesibilidad Web”, desarrollado por la Universidad Politécnica Salesiana (Figura 10).



Figura 10. Página de Inicio del OAW Ecuador

2.3. Diagrama de Clases

El diagrama de clases define de manera más clara cuáles serán objetos abstraídos de la realidad (clases) que serán usados para implementar las funcionalidades descritas en los requerimientos. Además, presenta cómo será su interacción entre ellos, y cuáles serán sus comportamientos y parámetros individuales.

En este contexto, a continuación, se presentará el diagrama de clases realizado para describir la estructura y comportamiento de las clases necesarias para implementar las funcionalidades de monitoreo continuo y análisis estadístico (Figura 11).

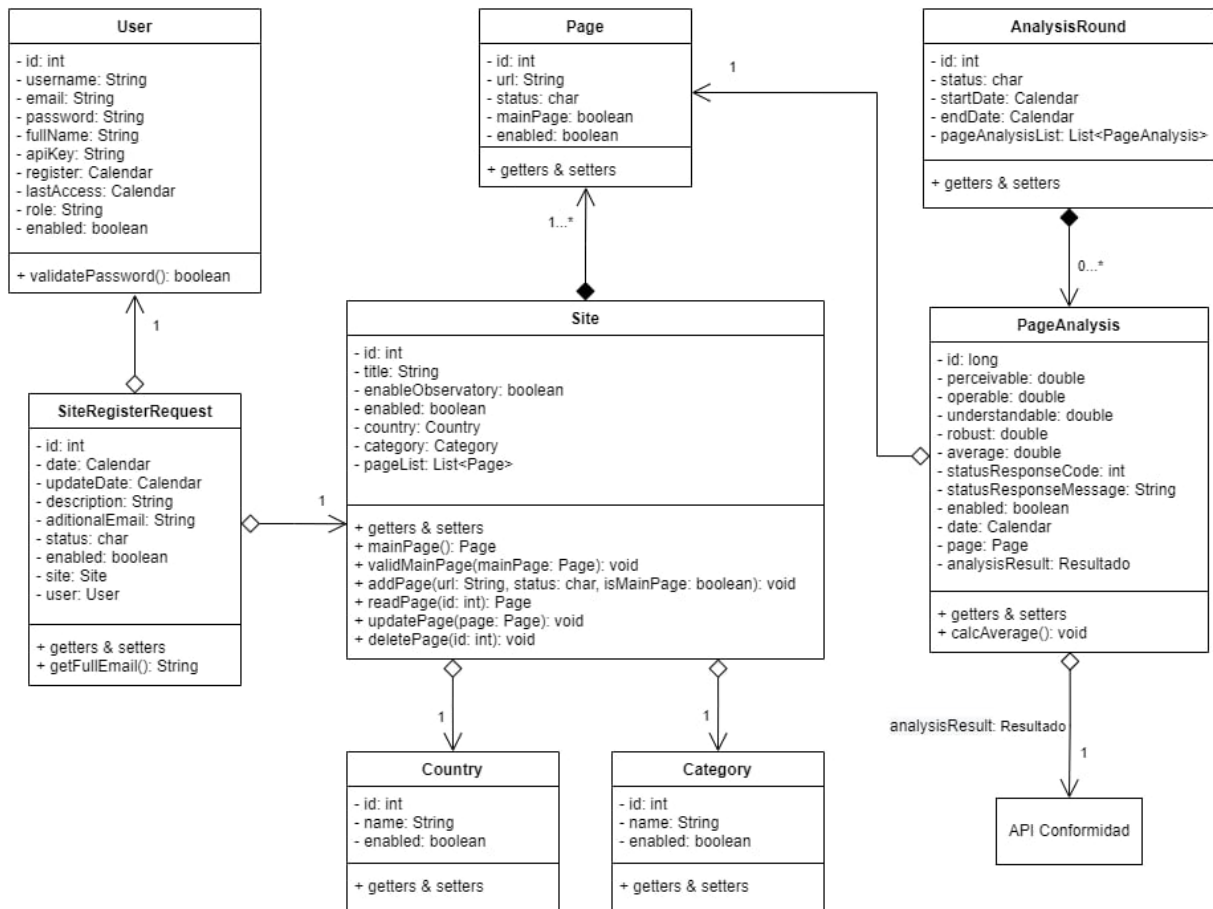


Figura 11. Diagrama de clases para los módulos de monitoreo y análisis estadístico

3. Codificación

Una vez acabada la fase de diseño, se realizó la implementación de la arquitectura general respetando la estructura propuesta. En esta fase se llevó a cabo la solución de los requerimientos funcionales y no funcionales mediante diversos lenguajes programación según el problema lo requería. Para ello, la codificación se puede abstraer en tres flujos de desarrollo que corresponden al módulo de monitoreo, módulo de análisis estadístico y la integración de las herramientas de accesibilidad, los cuales, serán detallados a continuación.

3.1. Desarrollo del módulo de monitoreo

El módulo de monitoreo fue desarrollado aplicando conceptos y técnicas de cómputo paralelo utilizando el lenguaje de programación Java. El cual, consiste en dividir la cantidad total de páginas registradas, para la cantidad de procesos o hilos. Donde, cada hilo realiza el análisis de conformidad a cada página que esté en cola (Tabla 19). De esta manera, se logró tener tiempos de análisis óptimos, a diferencia de realizar el proceso de análisis de manera secuencial.

Este módulo cuenta con un horario de ejecución mediante las utilidades de *Enterprise JavaBeans* (EJB). Además, este se encarga de realizar la persistencia de los resultados en la

base de datos. Para ello, el monitoreo se realiza en segundo plano y puede ser programada la hora y fecha de ejecución.

Asimismo, para tener una mejor administración en la configuración de las funcionalidades del módulo, se parametrizaron las siguientes opciones:

- Definición del horario del monitoreo, en términos de segundos, minutos y horas. Además, se determinan meses y días específicos.
- Cantidad de procesos o hilos para la ejecución concurrente.
- Cantidad de páginas que se asignarán a cada hilo, esta opción permite limitar el consumo de recursos.
- Cantidad de intentos en el análisis, esta opción permite aplicar una tolerancia por si ocurre un fallo inesperado durante el proceso de análisis.
- El nivel de conformidad y la resolución de pantalla con los que se realizará el análisis de conformidad.
- Atributos relacionados a la configuración del servicio de correo electrónico.

Además, para implementar mayor confidencialidad en los atributos que son más sensibles, se creó un archivo de configuración a parte que no será subido al repositorio remoto y que tiene que ser creado manualmente antes de desplegar en el servidor. Estos parámetros son:

- El correo electrónico que será el emisor.
- La contraseña del correo electrónico emisor, si es necesario realizar una autenticación.
- El o los correos electrónicos que serán los destinatarios para los mensajes de soporte.

Tabla 19. Pseudocódigo del monitoreo de la Accesibilidad Web

Algoritmo 1 Análisis de accesibilidad web concurrente

Entrada: Páginas registradas

Salida: Resultados del análisis de conformidad

- 1: **Inicio**
 - 2: **Esperando** (Horario)
 - 3: **Lectura** (Páginas Registradas)
 - 4: Generación (Hilos de Ejecución)
 - 5: Asignación (Páginas Registradas, Hilos de Ejecución)
-

-
- 6: Hilo de Ejecución = **Proceso Iterativo** (Hilos de Ejecución)
 - 7: **Cantidad** (Páginas por Hilo) = **Cantidad** (Páginas Registradas) / **Cantidad** (Hilos de Ejecución)
 - 8: Páginas Divididas = Subtraer (Páginas Registradas, Índice Inicial, Índice Final)
 - 9: Asignar (Páginas Divididas, Hilo de Ejecución)
 - 10: Comenzar (Hilos de Ejecución)
 - 11: Página = **Proceso Iterativo** (Páginas del Hilo)
 - 12: Resultado = Análisis de Conformidad (Página)
 - 13: **Si** (\sim Resultado Y Tolerancia > 0) **Entonces** (Ejecución 6.2.) **Caso Contrario** (Tolerancia - 1)
 - 14: Agregar Resultados (Resultado)
 - 15: Persistencia (Hilos de Ejecución)
 - 16: Resultado = Proceso Iterativo (Resultados)
 - 17: **Persistir** (Resultado)
 - 18: **Fin**
-

Este algoritmo se ejecuta en base a un horario programado que se configura al compilar el proyecto. Una vez llegado el momento, se realiza una consulta a la base de datos de las páginas registradas filtrando las que están habilitadas y que tengan el estado de aprobado. Después, se generan los hilos de ejecución y se asigna una lista de páginas divididas de manera equitativa. Además, los hilos pueden ser configurados con un límite máximo de páginas que analizarán para no consumir muchos recursos durante todo el proceso. Una vez terminado este procedimiento, si aún quedan páginas por ser analizadas, este se realizará nuevamente con las páginas restantes.

El proceso de cada hilo consiste en realizar el análisis de accesibilidad web de las páginas que se registraron mediante un proceso iterativo. Este, a su vez, se comunica con el módulo de análisis de conformidad y procesa los resultados para obtener los porcentajes de cumplimiento de los criterios de conformidad, los cuales serán de utilidad para el módulo de análisis estadístico.

Además, es importante resaltar que, este proceso tiene una tolerancia de fallo que repite el análisis en caso de que ocurra un error durante la comunicación. Finalmente, cuando termine el proceso de análisis de accesibilidad web de todos los hilos de ejecución, se realiza la persistencia de los datos para tener referencia de los resultados que se obtuvieron y ser consultados cuando se requiera.

3.2. Desarrollo del módulo de análisis estadístico

El módulo de análisis estadístico también fue desarrollado usando Java como lenguaje de programación. Para la generación de los datos estadísticos se realizaron tres servicios. Cada servicio tiene diferentes parámetros para filtrar los resultados de las consultas que se realizan a la base de datos y generar la respuesta en formato JSON. De esta manera, la capa de presentación puede utilizar estos datos para representar la información mediante gráficas.

El primer servicio genera los datos para la presentación del promedio de los principios de accesibilidad web de cada país. Para ello, se realiza una consulta a la base de datos para obtener el último y penúltimo monitoreo de todas las páginas analizadas. De esta manera, se obtiene una comparación general de cómo evolucionó la calificación de los principios de accesibilidad en comparación con el monitoreo previo.

El segundo servicio genera los datos para representar los promedios de los análisis de accesibilidad web de las categorías. Este genera la información necesaria para que sea representada en tres gráficas diferentes. Para ello, se debe proveer de las categorías, los países y un rango de tiempo que se utilizarán para filtrar los datos. Cada gráfica tiene un diferente enfoque, los cuales son:

- La primera gráfica de radar presenta el promedio de los principios de accesibilidad web de todas las categorías, respecto a los países y a un rango de tiempo seleccionado.
- La segunda gráfica de barras agrupa las categorías por cada país seleccionado, y presenta los promedios de accesibilidad web respecto que estén dentro de un rango de tiempo.
- La tercera gráfica muestra una línea de tiempo de la evolución de los principios de accesibilidad web en el rango de tiempo seleccionado.

Finalmente, el tercer servicio también genera datos estadísticos de los análisis de accesibilidad web, pero este se orienta a las páginas web. La información que se genera es representada en dos gráficas diferentes. Para ello, se debe proveer de una selección de páginas web y un rango de tiempo para filtrar los datos al momento de realizar las consultas a la base de datos. Estas gráficas se explican a continuación:

- La primera gráfica, una de radar, presenta el promedio de los principios de accesibilidad web de todas las páginas analizadas dentro del rango de tiempo seleccionado.
- La segunda gráfica presenta una línea de tiempo para observar la evolución de la accesibilidad web de las diferentes páginas web, dentro de un rango de tiempo determinado. Además, esta gráfica permite seleccionar un punto de análisis en la línea temporal para presentar a detalle los resultados que se obtuvieron.

3.3. Proceso de integración de herramientas de accesibilidad

Para el proceso de integración se tomaron en cuenta las tecnologías implementadas, o que fueron desarrolladas en las herramientas de accesibilidad existentes. Para esto, se realizó un análisis de las versiones que tengan soporte y compatibilidad con las tecnologías actuales. De esta manera, se pudieron detectar herramientas con componentes o módulos que están obsoletos o sin mantenimiento. Por lo tanto, en cada caso se tomaron diferentes enfoques para que se pueda realizar la integración de todas las herramientas.

Además, para aplicar la respectiva seguridad en los servicios RESTful desarrollados, se implementó el marco de seguridad Elytron que proporciona WildFly de manera nativa. De esta forma, se facilitó en gran medida la validación de roles que tienen permitido acceder a las funcionalidades de los servicios desarrollados. Para ello, estos servicios fueron segmentados en públicos y privados. Los servicios públicos no tienen un prefijo en la URL, a diferencia de las privadas, que tienen un prefijo respecto al rol el cual tiene acceso a determinada funcionalidad. Por lo tanto, para que un usuario pueda acceder a una funcionalidad, debe iniciar sesión y Elytron valida si este cumple con los requisitos para que el servidor de aplicación brinde sus servicios.

Los ajustes de Elytron consisten en definir varios parámetros que deben ser establecidos directamente en el archivo de configuración de WildFly. Estos ajustes pueden ser automatizados mediante un conjunto de instrucciones en un archivo para tener mayor facilidad de implementación para el despliegue del servicio (Anexo 7). Estas instrucciones son conformadas por comandos que la herramienta de administración de JBoss EAP puede procesar.

Elytron permite realizar varias configuraciones para validar las credenciales. Sin embargo, hay que considerar que este proyecto trabaja con bases de datos y cuenta con una tabla donde se almacenan los usuarios junto con sus credenciales cifradas y su respectivo rol. Además, el objetivo de configurar este marco de seguridad es que pueda ser accedido mediante una conexión HTTP y que pueda comunicarse con la capa de presentación mediante servicios web. Por lo que, la secuencia de configuraciones realizadas son las siguientes:

- 1. Definición de una conexión con la base de datos:** esta configuración consiste en establecer la consulta SQL que se realizará al momento de validar el inicio de sesión y obtener el rol asignado al usuario. Dentro de estos parámetros, también se debe establecer el algoritmo del cifrado que es aplicado a las contraseñas.
- 2. Crear el mapeo de los roles:** para que el marco de seguridad sepa qué roles se manejan en la capa de negocio, es necesario realizar un mapeo del rol que está asignado a un usuario. Cabe resaltar que, este marco soporta que un usuario tenga múltiples roles.
- 3. Crear el dominio de seguridad para el subsistema EJB3:** esta configuración enlaza la conexión a la base de datos, la validación de las credenciales del usuario y el mapeo

de roles con EJB3.

4. **Crear la autenticación mediante HTTP:** para que este marco de seguridad pueda ser aplicado a los servicios web, es necesario definir que el dominio de seguridad es accedido mediante una conexión HTTP.
5. **Establecer el método de autenticación:** para que la autenticación definida anteriormente pueda ser aplicada, es necesario enlazar este método con el servidor *Undertow*, que es el encargado de bloquear o permitir el paso de la conexión con los servicios web.
6. **Crear el dominio de seguridad de aplicación:** este se encarga de enlazar el dominio de seguridad y la autenticación mediante HTTP a nivel de aplicación. Es decir, que la seguridad dentro del servidor de aplicaciones de WildFly será manejada por Elytron.

A continuación, se presentarán las tecnologías implementadas, su versión y una descripción con lo que se realizó en cada capa de la arquitectura propuesta para integrar las herramientas que fueron desarrolladas.

Tabla 20. Tecnologías implementadas para la capa de presentación

| Tecnología | Versión | Descripción |
|------------|---------|---|
| Angular | 11.0.0 | Este <i>framework</i> fue utilizado para el desarrollo de las páginas web. Además, se optó por utilizar el patrón de arquitectura modelo-vista-controlador para facilitar su escalabilidad en el caso de ser necesario. |
| PrimeNG | 11.2 | Esta tecnología ofrece una serie de componentes orientados a la interfaz de usuario que está sobre el <i>framework</i> Angular. Básicamente, ofrece una mayor facilidad para aplicar estilos, animaciones, validaciones y ofrece componentes o widgets ya desarrollados para agilizar el desarrollo de las páginas web. |
| Highcharts | 9.1.2 | Esta librería de gráficas reemplazó el uso de aquellas que vienen por defecto con PrimeNG debido a que sus funcionalidades no fueron suficientes para cubrir con los requerimientos. |

Tabla 21. Tecnologías implementadas para la capa de negocio

| Tecnología | Versión | Descripción |
|----------------------------------|---------|--|
| Jakarta Enterprise Edition (JEE) | 1.8 | <p>Se utilizó esta plataforma para desarrollar los servicios RESTful, los módulos de monitoreo y análisis estadístico, así como la comunicación con la base de datos para la persistencia o lectura de los datos.</p> <p>También, en esta capa se configuró la validación de roles con el marco de seguridad Elytron de WildFly. Para ello, se colocaron las anotaciones con los roles en los respectivos servicios y se configuró el inicio de sesión mediante HTTP. Cabe resaltar que, la configuración se hace mediante el uso de un <i>script</i> para automatizar durante el despliegue. Dentro del script, se instancian los dominios de seguridad, así como la consulta a la base de datos para validar el inicio de sesión y el rol asignado a el usuario.</p> <p>Además, en esta plataforma se reutilizaron las funcionalidades ya desarrolladas de las herramientas de accesibilidad web de la UPS. Sin embargo, se aplicaron refactorizaciones de código y se modificaron algunas reglas de negocio que se aplicaban anteriormente para que se adapten a los requerimientos actuales.</p> |
| NodeJS | 14 | <p>En este entorno se desarrolló el analizador de contraste. Sin embargo, se realizaron varios cambios en las reglas de negocio y los métodos de las funcionalidades para que esta API se adapte a los requerimientos actuales.</p> <p>Además, se quitaron algunos procedimientos que quedaron obsoletos después de aplicar cambios necesarios.</p> <p>También, el anterior analizador de conformidad fue desarrollado con PhantomJS, que actualmente su mantenimiento se encuentra suspendido. Sin embargo, se pudo observar varias falencias</p> |

| | | |
|--|--|--|
| | | <p>durante el análisis.</p> <p>Por lo tanto, se sustituyó esta tecnología por una nueva llamada Puppeteer utilizando NodeJS como entorno. Esto resultó en una mejor renderización de las páginas y detección de elementos al aplicar el análisis de conformidad.</p> |
|--|--|--|

Tabla 22. Tecnologías implementadas para la capa de persistencia

| Tecnología | Versión | Descripción |
|--|----------------------|--|
| Java Persistence API (JPA) | 2.1 | Mediante esta API, se realizó la comunicación entre la capa de negocios y la capa de datos para su persistencia. |
| Java EE Database Connectivity (JDBC) | PostgreSQL JDBC 42.4 | Este controlador permite la interacción con la base de datos PostgreSQL, de esta manera, se realizan las ejecuciones de las consultas necesarias. |
| Java Persistence Query Language (JPQL) | 2.1 | <p>Es el lenguaje utilizado en Java para crear consultas orientadas a objetos. De esta manera, se puede facilitar la creación de consultas que pueden resultar complejas.</p> <p>Esto ayudó en las consultas generadas para obtener los datos para el análisis estadístico, facilitando en gran medida el procedimiento.</p> |

Tabla 23. Tecnologías implementadas para la capa de datos

| Tecnología | Versión | Descripción |
|-------------------|----------------|---|
| PostgreSQL | 10.13 | Es el sistema de bases de datos relacional utilizado para almacenar los resultados del análisis de conformidad y los de contraste. |
| DBeaver | 22.1.2 | Es una herramienta que sirve para gestionar bases de datos. Esta facilitó la migración de los datos necesarios a las nuevas tablas creadas o modificadas. |

4. Prueba y Verificación

4.1. Pruebas Funcionales

Como última parte del desarrollo realizado de este proyecto, la verificación del cumplimiento de los requerimientos se realiza a través de pruebas funcionales. Este tipo de pruebas se basan en analizar los requerimientos funcionales establecidos, explicar el proceso y los resultados que deberían obtenerse, y compararse con los resultados reales que se obtienen al interactuar con la herramienta web desarrollada. A continuación, se presentarán algunas de las pruebas realizadas para validar el cumplimiento de los requerimientos planteados.

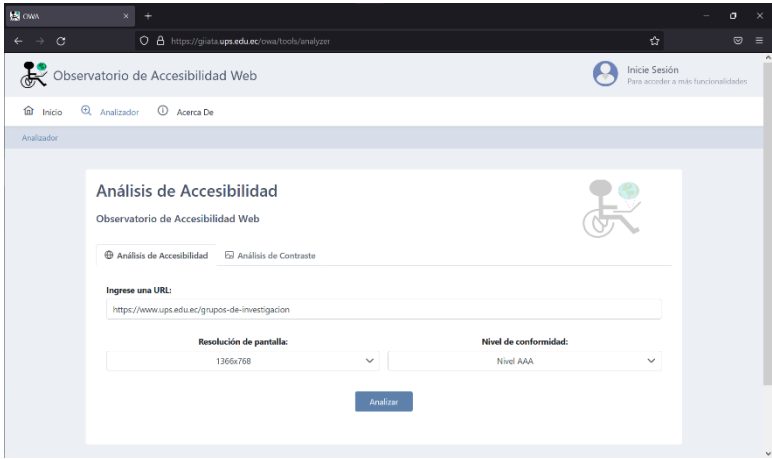
Tabla 24. Prueba Funcional 1: Análisis de Accesibilidad Web

| Prueba Funcional 1 | Código | PF-1 | Resultado |
|--------------------|-------------|------------------|-----------|
| | Responsable | Eduardo Zhizhpon | Aprobada |
| | Fecha | 12/07/2022 | |

Requerimiento:
RF-1: Análisis de Accesibilidad Web

Resultado Esperado:
El usuario ingresará la URL de la página que desee analizar, además de la resolución de pantalla y el nivel de conformidad para el análisis. A continuación, dará clic en “Analizar”, y posteriormente será redirigido a una página en donde se muestren dos secciones: un resumen del análisis, y otra en donde se presenten los resultados de manera más detallada, junto con una gráfica de radar.

Resultado Obtenido:



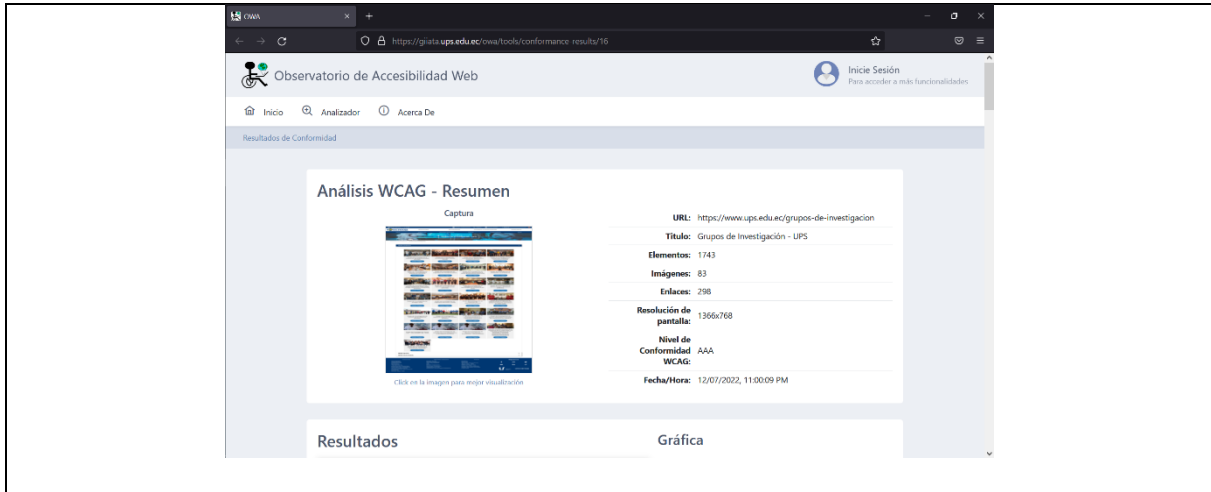


Tabla 25. Prueba Funcional 2: Dashboard

| Prueba Funcional 2 | Código | PF-2 | Resultado |
|--------------------|-------------|-----------------|-----------|
| | Responsable | Bryan Sarmiento | |
| | Fecha | 12/07/2022 | |
| Aprobada | | | |

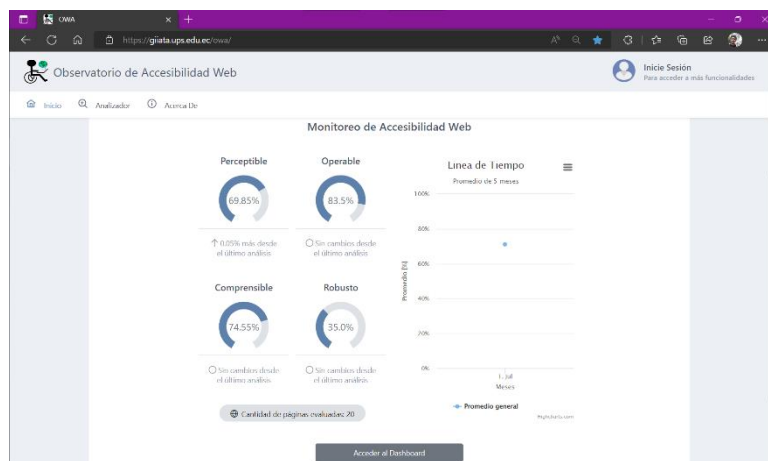
Requerimiento:

RF-4: Dashboard

Resultado Esperado:

El usuario, en la página de inicio, hará clic en el botón “Acceder al Dashboard”, posteriormente será redirigido a la página del Dashboard. Aquí, podrá visualizar tres secciones, en las cuales habrá más gráficas e información que podrá filtrar según países, categorías, páginas y fechas.

Resultado Obtenido:



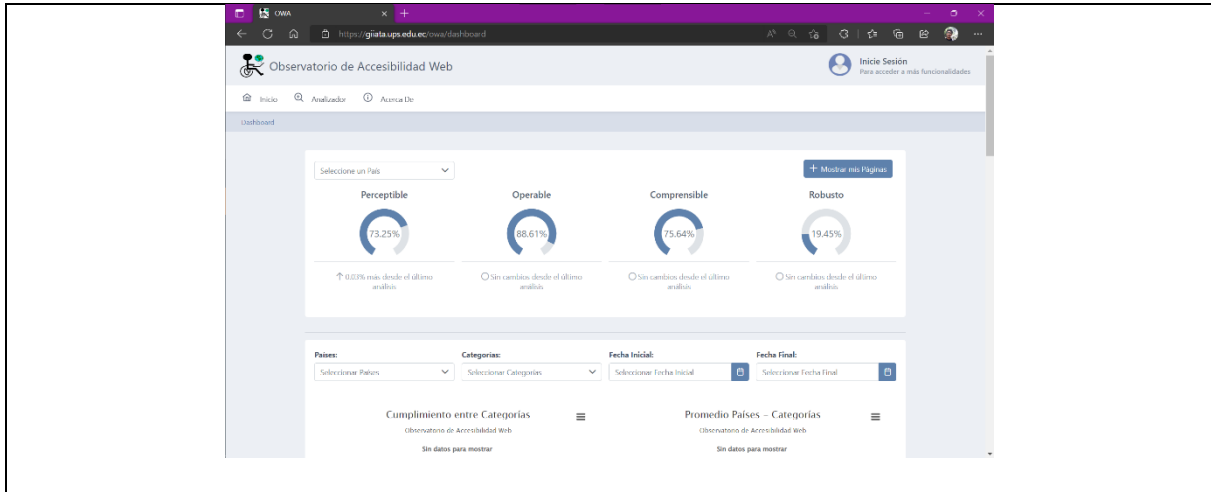


Tabla 26. Prueba Funcional 3: Actualización dinámica de gráficas (Países)

| Prueba Funcional 3 | Código | PF-3 | Resultado |
|--------------------|-------------|------------------|-----------|
| | Responsable | Eduardo Zhizhpon | |
| | Fecha | 13/07/2022 | |
| | | | Aprobada |

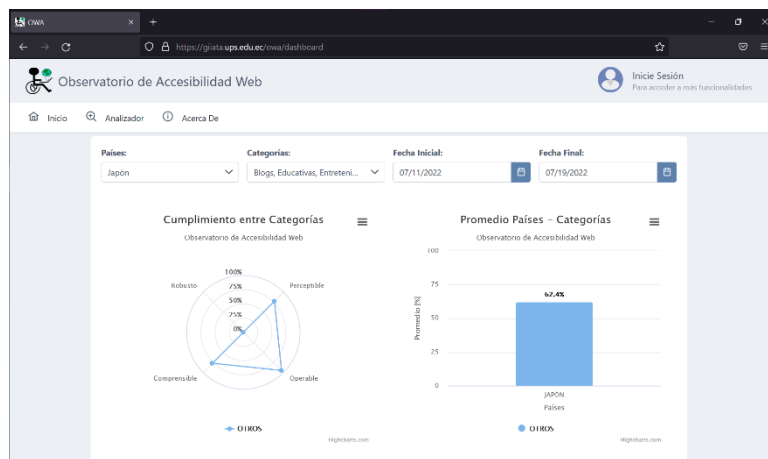
Requerimiento:

RF-5: Actualización Dinámica de Gráficas (Filtro por países)

Resultado Esperado:

El usuario se encontrará en la página “Dashboard”. Una vez allí, se observarán todas las gráficas disponibles. Al seleccionar, por ejemplo, el parámetro de “Países” de la segunda sección del dashboard, y activar la visualización de un país, se presentarán solo los resultados de ese país. Si se activa otro país, las gráficas se actualizarán para mostrar los resultados de ambos países, y se diferenciarán claramente uno del otro.

Resultado Obtenido:



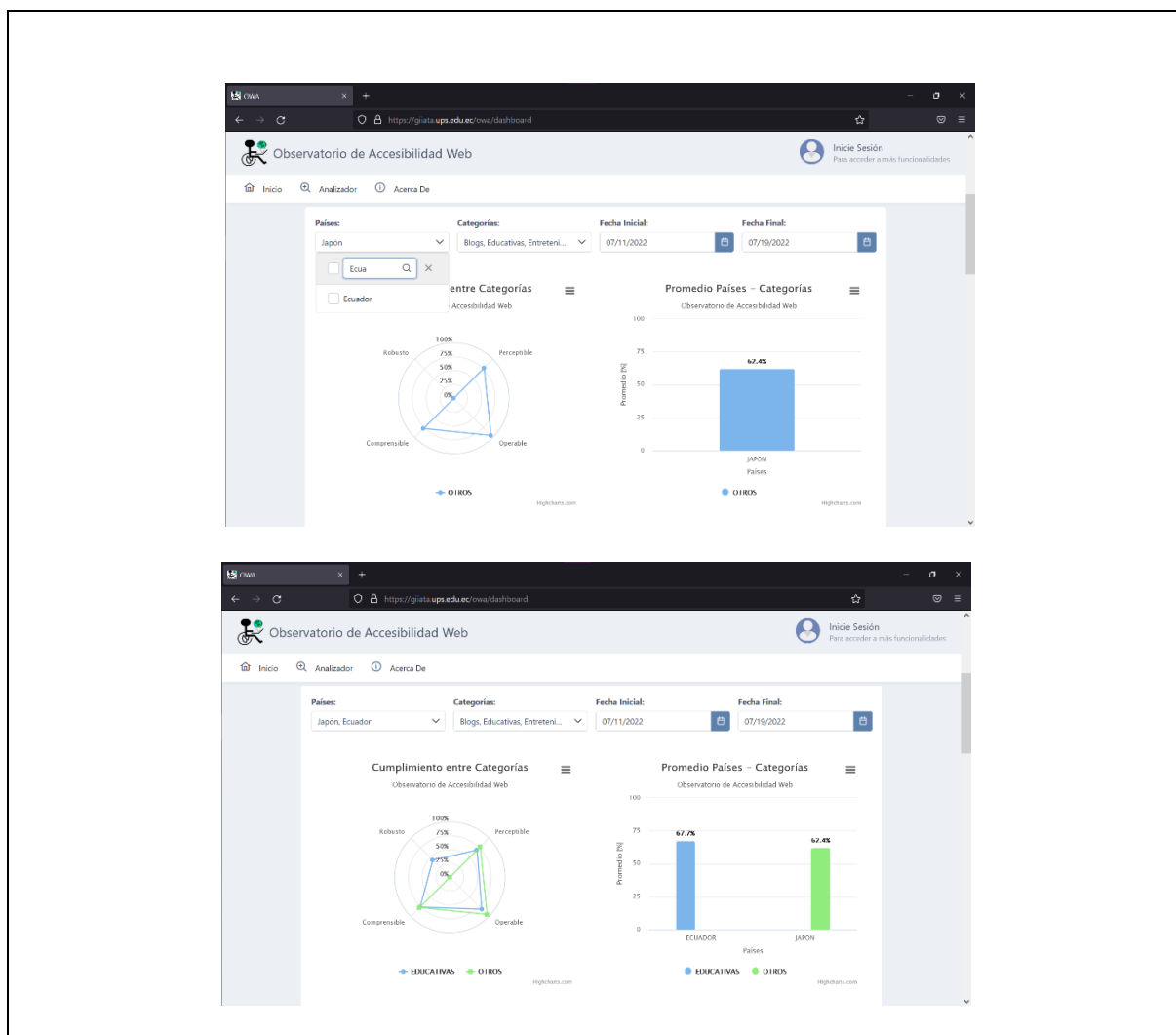


Tabla 27. Prueba Funcional 4: Solicitud de Registro (Usuario)

| Prueba Funcional 4 | Código | PF-4 | Resultado |
|--------------------|-------------|-----------------|-----------|
| | Responsable | Bryan Sarmiento | |
| | Fecha | 20/07/2022 | |

Requerimiento:

RF-8: Solicitud de Registro – Usuario

Resultado Esperado:

El usuario podrá generar una solicitud para registrar nuevas páginas. Para ello en la página “Mis páginas” hará clic en “Registrar Nuevo Sitio Web”. A continuación, se abrirá un formulario de registro en donde se tendrá que llenar información acerca de la página; si todas las páginas son válidas y se aceptan los términos y condiciones, el usuario podrá dar clic en “Registrar”. Finalmente, se presentará un mensaje indicando que la solicitud se creó correctamente.

Resultado Obtenido:

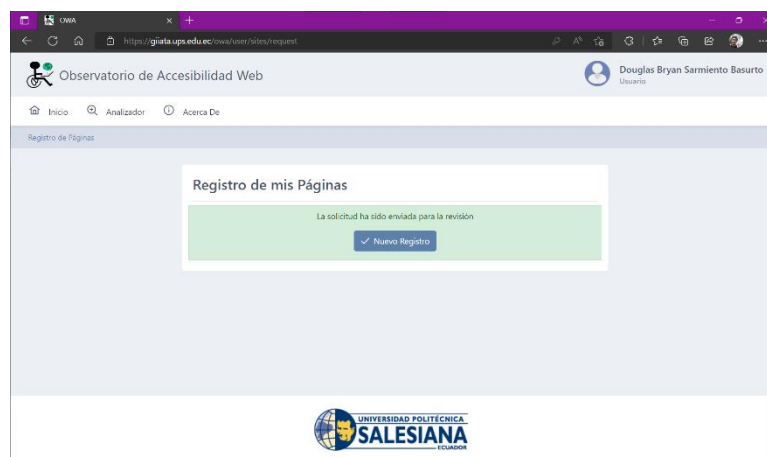
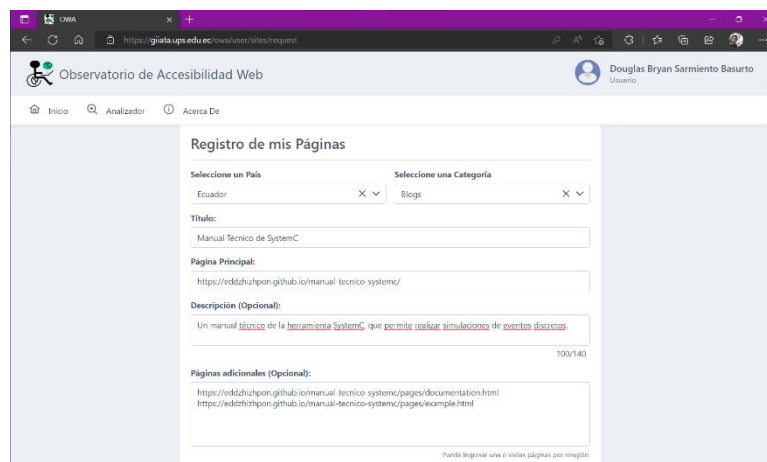
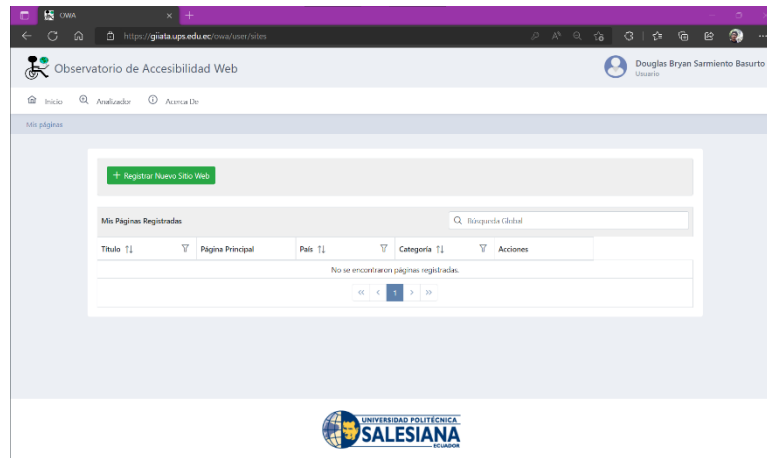


Tabla 28. Prueba Funcional 5: Solicitud de Registro (Administrador)

| | | | |
|---------------------------|--------------------|------------------|------------------|
| Prueba Funcional 5 | Código | PF-5 | Resultado |
| | Responsable | Eduardo Zhizhpon | |
| | Fecha | 20/07/2022 | |

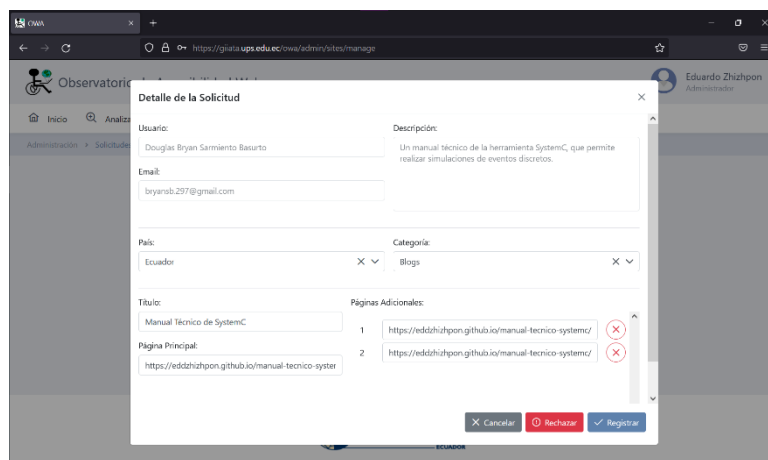
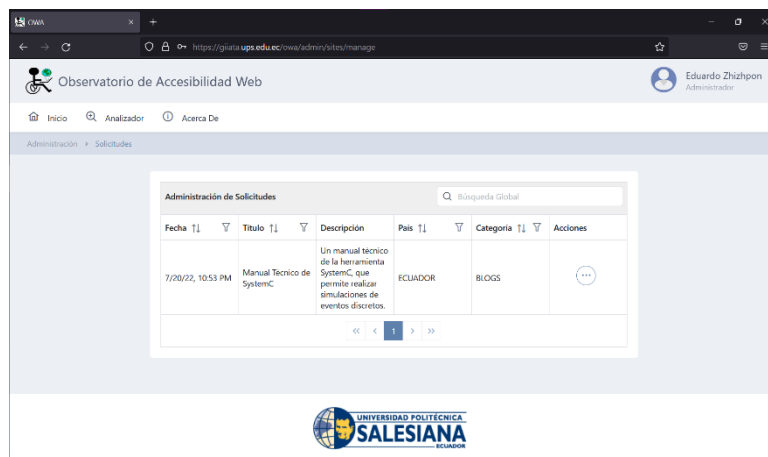
Requerimiento:

RF-9: Solicitud de Registro – Administrador

Resultado Esperado:

El administrador, en la página “Solicitudes”, podrá observar todas las solicitudes pendientes. Para abrir una solicitud, hará clic en el botón dentro de la columna “Acciones”. A continuación, se abrirá una ventana emergente en donde se presentará toda la información de la solicitud. En la misma podrá realizar modificaciones, de crear conveniente. Finalmente, luego de aprobar o rechazar la solicitud, aparecerá una nueva ventana emergente en la que se presente una previsualización del correo electrónico que se enviará al usuario indicando el cambio de estado de la solicitud.

Resultado Obtenido:



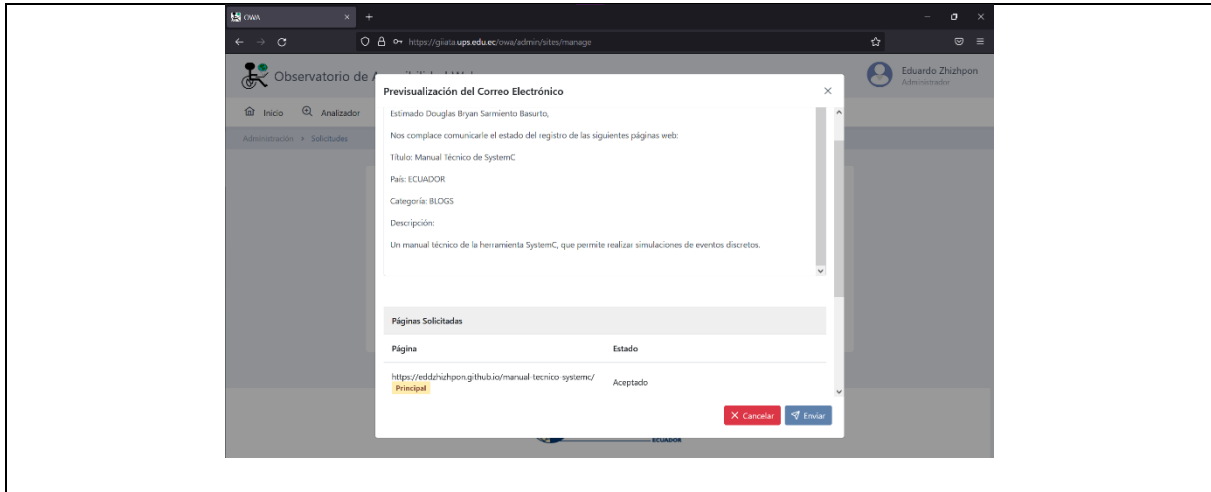


Tabla 29. Prueba Funcional 6: Gestión de Páginas Web (Creación)

| Prueba Funcional 6 | Código | PF-6 | Resultado |
|--------------------|-------------|------------------|-----------|
| | Responsable | Eduardo Zhizhpon | |
| | Fecha | 20/07/2022 | |
| Aprobada | | | |

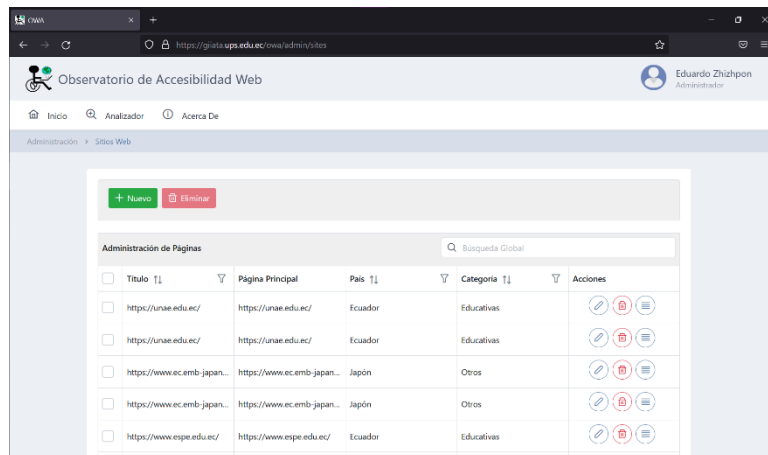
Requerimiento:

RF-12: Gestión de Páginas Web (Creación de páginas)

Resultado Esperado:

El administrador, en la página “Sitios Web”, podrá observar todas las páginas registradas en la base de datos. En esta, él podrá editar, eliminar (de manera lógica), crear nuevas páginas o restaurar páginas que se han eliminado. Para crear una página, hará clic en “Nuevo”, lo que hará que salga una ventana emergente con los datos básicos para crear nuevas páginas (país, categoría y URLs). Finalmente, hará clic en “Registrar”, lo que hará que desaparezca la ventana y aparezca un mensaje flotante.

Resultado Obtenido:



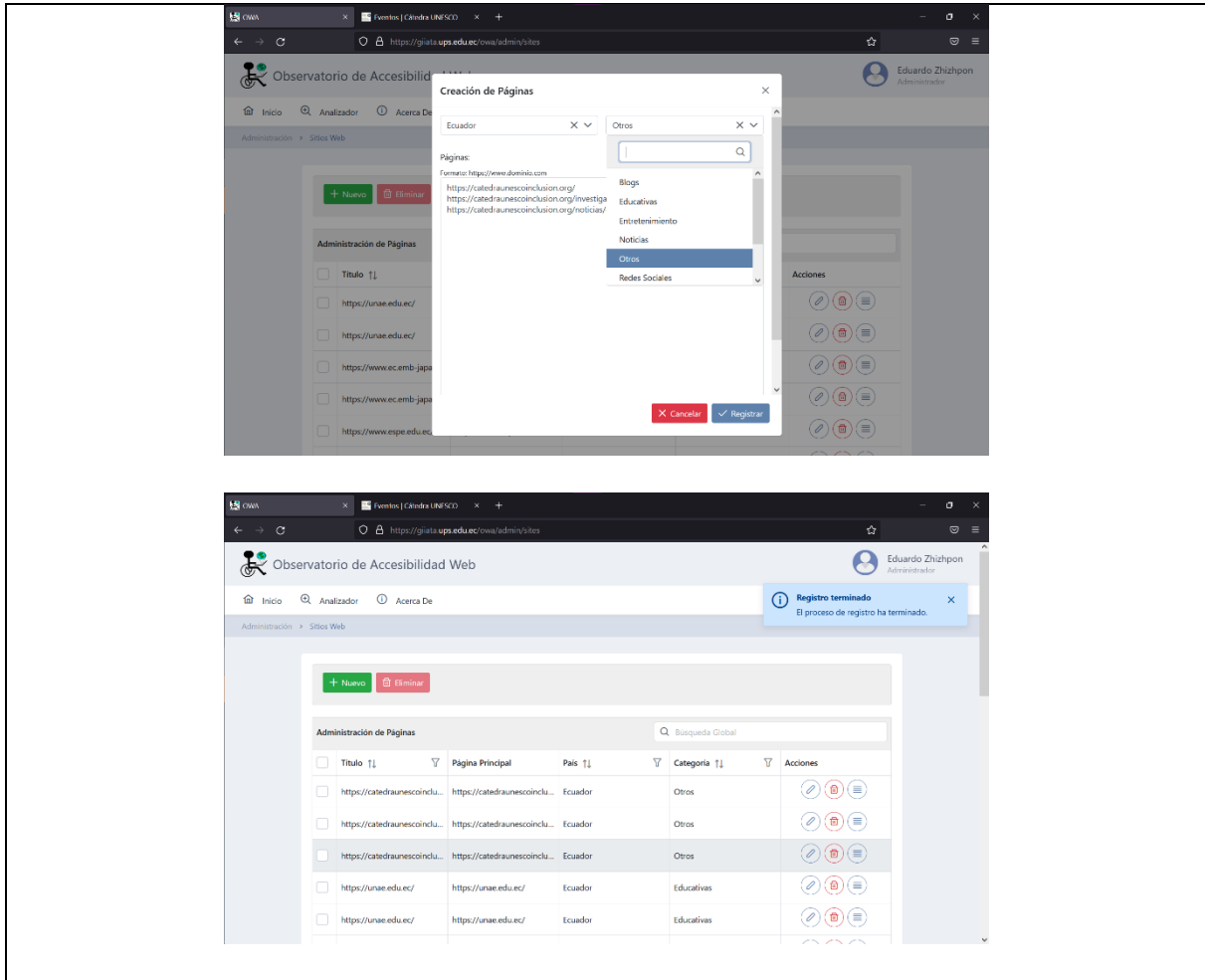


Tabla 30. Prueba Funcional 7: Análisis de Contraste

| Prueba Funcional 7 | Código | PF-7 | Resultado |
|--------------------|-------------|-----------------|-----------|
| | Responsable | Bryan Sarmiento | |
| | Fecha | 22/07/2022 | |
| Aprobada | | | |

Requerimiento:

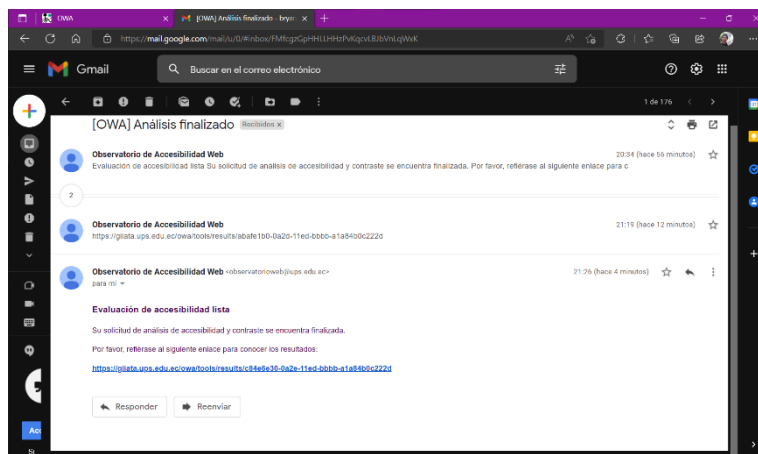
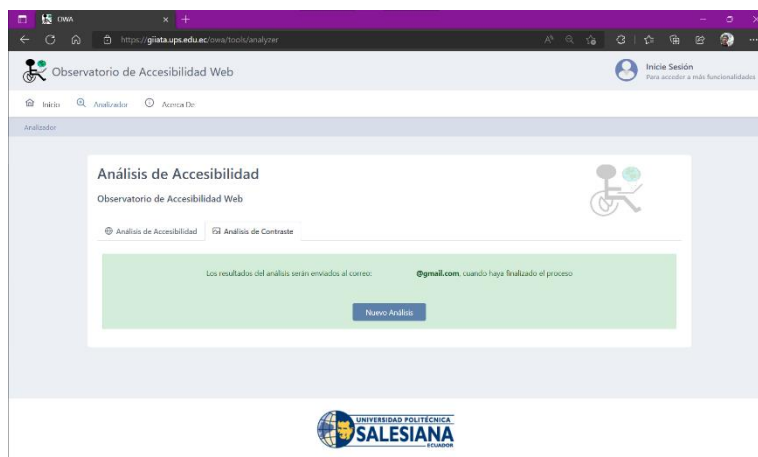
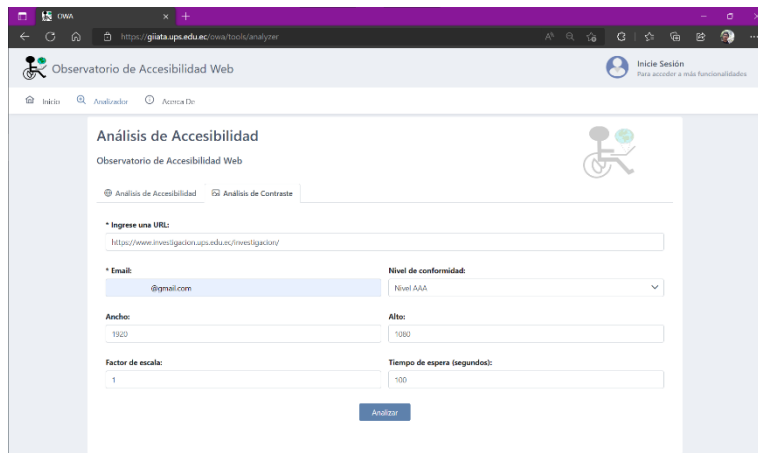
RF-3: Análisis de Contraste

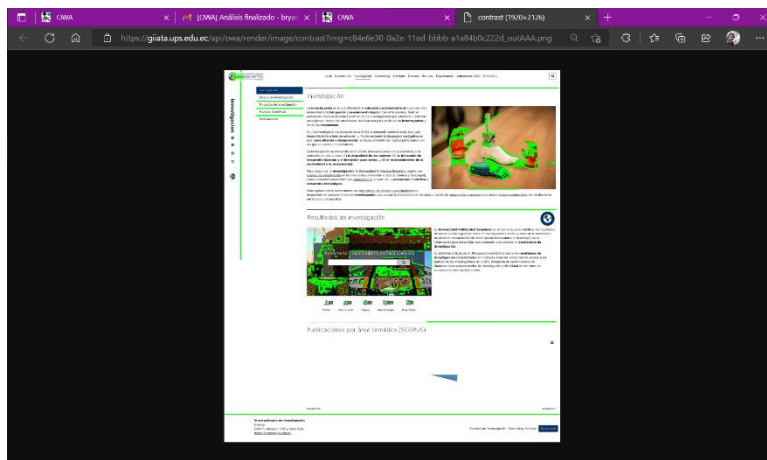
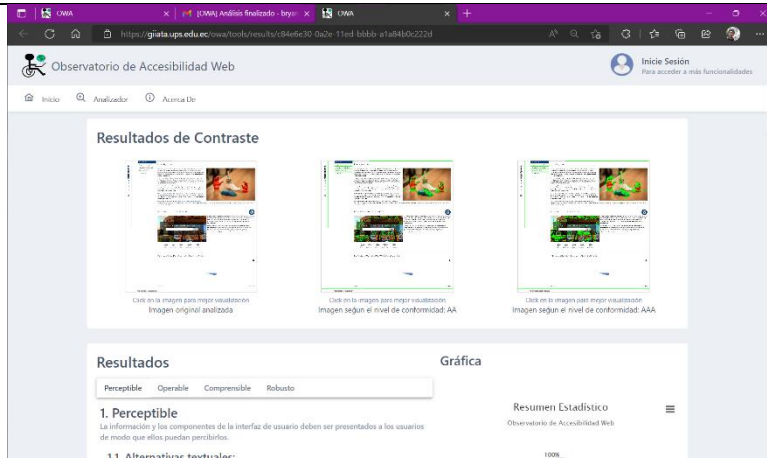
Resultado Esperado:

El usuario ingresará la URL de la página que desee analizar, así como su correo electrónico, la resolución de pantalla (ancho y alto), el nivel de conformidad (A, AA o AAA), el factor de escala y el tiempo de espera. Posteriormente dará clic en “Analizar”, para enviar la solicitud de análisis. Una vez hecho esto, aparecerá en pantalla un mensaje en donde se menciona que se informará cuando los resultados estén listos al correo que se proporcionó. Una vez se haya realizado el análisis al usuario le llegará un correo con el enlace en donde se encuentran los resultados de contraste. Al hacer clic en el mismo, se abrirá la página web que contendrá un resumen del análisis, las capturas de pantalla de la página web analizada,

así como los resultados del análisis de conformidad. Si el usuario hace clic en una de las imágenes, se abrirá otra pestaña con una mejor visualización de esta.

Resultado Obtenido:





VII. Cronograma

En esta sección se detalla el cronograma de actividades realizado durante el desarrollo de la Herramienta Web propuesta.

Tabla 31. Cronograma de actividades del proyecto

| Objetivo | Actividad | Fecha inicio | Fecha final | Responsables | Duración (h) |
|----------|--|--------------|-------------|--------------|--------------|
| OE1 | OE1A1: Estudio de fundamentos sobre el proceso de evaluación | 12/04/2022 | 12/04/2022 | B, E | 10 |
| | OE1A2: Estudio de documentación existente | 13/04/2022 | 13/04/2022 | B, E | |
| OE2 | OE2A1: Diseño del diagrama de clases | 14/04/2022 | 15/04/2022 | B, E | 100 |
| | OE2A2: Desarrollo del motor de procesamiento concurrente | 14/04/2022 | 27/04/2022 | B, E | |
| | OE2A3: Desarrollo de funcionalidades – Módulo de Monitoreo | 18/04/2022 | 27/04/2022 | B, E | |
| OE3 | OE3A1: Desarrollo de funcionalidades – Módulo Estadístico | 28/04/2022 | 10/05/2022 | B, E | 70 |
| OE4 | OE4A1: Diseño y desarrollo de la interfaz de usuario | 28/04/2022 | 10/05/2022 | B, E | 380 |
| | OE4A2: Diseño y desarrollo del componente de administración | 11/05/2022 | 07/06/2022 | B, E | |
| | OE4A3: Desarrollo de los servicios RESTful | 08/06/2022 | 15/06/2022 | B, E | |
| | OE4A4: Integración de las herramientas de la UPS | 16/05/2022 | 08/07/2022 | B, E | |
| | OE4A5: Documentación del proceso de desarrollo web | 08/07/2022 | 18/07/2022 | B, E | |
| OE5 | OE5A1: Diseño de un plan de pruebas funcionales | 11/07/2022 | 12/07/2022 | B, E | 40 |

| | | | | | |
|--|--------------------------------------|------------|------------|------|--|
| | OE5A2: Ejecución del plan de pruebas | 12/07/2022 | 22/07/2022 | B, E | |
|--|--------------------------------------|------------|------------|------|--|

Tabla 32. Resumen de horas por estudiante

| Total de horas: 600h | |
|--|--------------|
| Responsable | Horas |
| (B) - Sarmiento Basurto Douglas Bryan | 300h |
| (E) - Zhizhpon Tacuri Cesar Eduardo | 300h |
| Fecha de Inicio: 12/04/2022 — Fecha de Fin: 22/07/2022 | |

VIII. Presupuesto

A continuación, se muestra el presupuesto tentativo requerido para el desarrollo de la Herramienta Web desarrollada.

Tabla 33. Presupuesto para la realización del proyecto

| Denominación | Cantidad | Costo unitario | Costo total |
|----------------------------------|----------|----------------|---------------------|
| | Unidades | Dólares | Dólares |
| 1. Bienes | | | |
| Impresiones | 100 | 0,05 | 5 |
| 2. Tecnológico | | | |
| Computador portátil | 2 | 2.000 | 4.000 |
| Licencia de Highcharts | 1 | 690 | 690 |
| Licencia de plantilla de PrimeNG | 1 | 59 | 59 |
| 3. Servicios | | | |
| Servicio de transporte | 100 días | 0,7 | 70 |
| Servicio de internet | 5 meses | 30 | 150 |
| Alimentación | 100 días | 8 | 800 |
| 4. Personal | | | |
| Estudiante investigador | 2 | 5.400 | 10.800 |
| Asesor especializado | 1 | 1.000 | 1.000 |
| 5. Otros | | | |
| Fondo de emergencias | 1 | 530 | 530 |
| Total | | | \$ 18.104,00 |

IX. Conclusiones

Un tablero de mando (dashboard) soportado de una evaluación automatizada y periódica de páginas web, posibilita un seguimiento o monitoreo de la accesibilidad, convirtiéndose en una herramienta de apoyo para mejorar la calidad de los contenidos web.

La investigación preliminar, acerca de los aspectos más relevantes dentro del proceso de evaluación de la accesibilidad web, permitió encaminar los requerimientos y el diseño de los prototipos para complementar el conocimiento y evaluación de un experto. A su vez, resaltó el beneficio que genera el desarrollo de este tipo de herramientas.

La aplicación de técnicas de cómputo paralelo permitió agilizar el proceso de evaluación, considerando que la cantidad de páginas que se requieren analizar incrementa cada vez que se aprueba otra solicitud de registro. Esta ventaja se da al dividir el total de páginas registradas en partes iguales respecto a la cantidad de hilos de ejecución, lo que posibilita un mejor aprovechamiento de la capacidad de recursos disponibles en el servidor, y que este no colapse en el proceso de monitoreo.

El uso de filtros básicos, mediante países, categorías y fechas, fue de gran importancia para lograr segmentar los datos y así obtener información visual más específica acerca de un análisis. Es así como, se hace más evidente la evolución de la accesibilidad web en determinados sectores. De esta manera, se pueden identificar las áreas en donde más se requiera un plan de mejora, para así perfeccionar la calidad de los sitios web y aumentar el alcance de la información.

La integración de las herramientas pertenecientes al Proyecto del Observatorio de Accesibilidad Web (OAW) y la herramienta que se ha desarrollado, se realizó satisfactoriamente, luego de ejecutar un análisis de las tecnologías que fueron usadas. Los resultados del análisis determinaron la necesidad de realizar correcciones menores en el código antiguo, así como el uso de tecnologías más recientes en lugar de las que estaban implementadas y ya no contaban con soporte.

Finalmente, en base a los resultados del plan de pruebas diseñado, se determinó que la herramienta web cumple con todos los requerimientos funcionales establecidos para alcanzar los objetivos propuestos en base a la problemática expuesta.

En conclusión, el trabajo desarrollado cumple con su objetivo principal, el cual es proveer una herramienta web que sirva como apoyo para la evaluación constante de páginas web, y de esta manera cambiar la perspectiva que se tiene acerca de la importancia de desarrollar sitios web accesibles para garantizar la información a todas las personas.

X. Recomendaciones

Como recomendaciones, principalmente se debe resaltar la importancia de analizar la compatibilidad de versiones de las tecnologías que se desean utilizar, con las tecnologías disponibles. Esto evitará que surjan errores y fallos durante las fases tempranas del desarrollo.

Otra recomendación para tener en cuenta es la correcta realización de la fase de elicitación de requerimientos. Esto es importante debido a que los requerimientos son la base sobre la que se construye una aplicación. Si los requerimientos son ambiguos, o contradictorios entre sí, aumentará la complejidad del desarrollo. Además, otro aspecto importante es la priorización de requerimientos, de esta manera se podrá obtener una aplicación mayormente funcional en menor tiempo.

Por otro lado, es importante seguir convenciones y estándares en las fases de desarrollo para reducir la probabilidad de ocurrencia de errores, además de facilitar el proceso de mantenimiento una vez acabado con el desarrollo, y así garantizar un producto de calidad.

Como recomendación final, para trabajos futuros, se propone investigar qué otros servicios de análisis de accesibilidad se podrían integrar a la herramienta desarrollada. Esto, debido a que no todos los servicios de accesibilidad siguen las mismas pautas de evaluación, o logran identificar los mismos fallos. Por lo tanto, al integrar más servicios, es posible mejorar la fidelidad de los resultados obtenidos a través del monitoreo continuo.

Referencias Bibliográficas

- Acosta, T., & Luján-Mora, S. (2017). Análisis de la accesibilidad de los sitios web de las universidades ecuatorianas de excelencia. *Enfoque UTE*, 8(1), 46–61. <https://doi.org/10.29019/enfoqueute.v8n1.133>
- Asamblea Nacional República del Ecuador. (2012). Registro Oficial No 796/2012. Ley Orgánica de Discapacidades. In *Consejo Nacional para la Igualdad de Discapacidades*. https://www.consejodiscapacidades.gob.ec/wp-content/uploads/downloads/2014/02/ley_organica_discapacidades.pdf
- CONADIS. (2022). Estadísticas de Discapacidad. In *Consejo Nacional para la Igualdad de Discapacidades*. <https://www.consejodiscapacidades.gob.ec/estadisticas-de-discapacidad/>
- Córdova Viera, Y., Martínez Borrego, J., & Córdova Viera, E. (2021, October 4). Propuesta de metodología para el diseño de dashboard. *Revista Cubana De Transformación Digital*, 56–76. <https://doi.org/10.5281/zenodo.5545998>
- Cruz, F. A., Dabin, A. J., Dorsch, J. P., Koutsaniti, E., Lezcano, N. F., Martinasso, M., & Petrusic, D. (2020). FirecREST: a RESTful API to HPC systems. *2020 IEEE/ACM International Workshop on Interoperability of Supercomputing and Cloud Technologies (SuperCompCloud)*.
- Docs WildFly. (2021). *WildFly Elytron Security*. Docs WildFly. https://docs.wildfly.org/26/WildFly_Elytron_Security.html
- González González, F., & Calero Castañeda, S. L. (2019). Comparación de las metodologías cascada y ágil para el aumento de la productividad en el desarrollo de software. In *Repositorio Universidad Santiago de Cali*. <https://repository.usc.edu.co/handle/20.500.12421/1208>
- Highcharts. (2022). Highcharts Support. In *Highcharts*. <https://www.highcharts.com/blog/support/>
- Jiménez, A. E. (2020). Educación inclusiva e igualdad de las personas con discapacidad en la transformación digital. *Revista jurídica valenciana*, 35, 1–13. <https://dialnet.unirioja.es/servlet/articulo?codigo=7431277>
- León Pirela, A. R., & Pérez, C. E. (2019). ANÁLISIS ESTADÍSTICO EN INVESTIGACIONES POSITIVISTAS: MEDIDAS DE TENDENCIA CENTRAL. *Orb., Rev. Cient.* <https://dialnet.unirioja.es/servlet/articulo?codigo=7065797>
- Luján Mora, S. (2022). Accesibilidad Web: Ecuador. In *Accesibilidad Web Universidad de Alicante*. <http://accesibilidadweb.dlsi.ua.es/?menu=ecuador>
- Macías, R. M., Zambrano Mejía, L., Torres Rengifo, S. de J., Mora Marcillo, B., Almeida Cedeño, J. B., & Zambrano Mejía, D. A. (2020). Accesibilidad web: una revisión a la legislación y páginas web gubernamentales del Ecuador. *Suplemento CICA*

- Multidisciplinario ISSN:2631-2832, 4(10), 125–150.*
<https://suplementocica.uleam.edu.ec/index.php/SuplementoCICA/article/view/19>
- Padmanabhan, P. (2018). *Java EE 8 and Angular: A practical guide to building modern single-page applications with Angular and Java EE*. Packt Publishing.
- Pautasso, C. (2014). RESTful web services: Principles, patterns, emerging technologies. In *Web Services Foundations* (pp. 31–51). Springer New York. https://doi.org/10.1007/978-1-4614-7518-7_2
- PrimeNG. (2022). PrimeNG. In *Primefaces.org*. <https://www.primefaces.org/primeng/#/>
- Rangle.io. (2022). Why Angular? In *Rangle.io: Angular Training Guide*. https://angular-training-guide.rangle.io/why_angular_2
- Saeed, L. (2020). What Is Java EE? In *Introducing Jakarta EE CDI* (pp. 1–2). Apress. https://doi.org/10.1007/978-1-4842-5642-8_1
- Timbi-Sisalima, C., Amor, C. I. M., Otón, S., Hilera, J. R., & Aguado-Delgado, J. (2016). Comparative analysis of online web accessibility evaluation tools. *International Conference on Information Systems Development (ISD)*. <https://aisel.aisnet.org/isd2014/proceedings2016/CreativitySupport/3/>
- Timbi-Sisalima, C., Hilera, J. R., Otón, S., & Ingavelez, P. (2016). Developing a RESTful API for a web accessibility evaluation tool. *Proceedings of the 18th International Conference on Enterprise Information Systems*. <https://doi.org/10.5220/0005832404430450>
- W3C. (2018). Web Content Accessibility Guidelines (WCAG) 2.1. In W3C. <https://www.w3.org/TR/2018/REC-WCAG21-20180605/>
- W3C. (2022). Website accessibility conformance evaluation methodology (WCAG-EM) 1.0. In W3C. <https://www.w3.org/TR/WCAG-EM/>
- Wan, F., Wu, X., & Zhang, Q. (2020). Chain-oriented load balancing in microservice system. *2020 World Conference on Computing and Communication (WCCCT)*.
- WildFly. (2022). *About WildFly*. WildFly. <https://www.wildfly.org/about/>
- Wohlgethan, E. (2018). *Supporting Web Development Decisions by Comparing Three Major JavaScript Frameworks: Angular, React and Vue.js* [Universidad de Ciencias Aplicadas de Hamburgo]. https://reposit.haw-hamburg.de/bitstream/20.500.12738/8417/1/BA_Wohlgethan_2176410.pdf
- Yachimba, C., & Aracelly, J. (2022). *Evaluación de accesibilidad web en portales estatales de servicios ciudadanos*. <https://repositorio.pucesa.edu.ec/handle/123456789/3642>
- Zaccone, G. (2019). *Python Parallel Programming Cookbook: Over 70 recipes to solve challenges in multithreading and distributed system with Python 3* (2nd ed.). Packt Publishing.

Anexos

Anexo 1. Prototipo: Gestión de países



Anexo 2. Prototipo: Gestión de páginas

Gección de Páginas

http://observatorioweb.ups.edu.ec/owa/admin/sites

OAW Ecuador Inicio Analizador Web Acerca de API Rest Contactos Cerrar Sesión

Gección de Páginas

Menú


- Item One
- Item Two
- Administrar Análisis Automático**
- Gestionar Países
- Gestionar Categorías
- Gestionar Páginas**
- Gestionar Solicitudes de Págin
- Item Four

+ Nuevo Eliminar País: Todos Categorías: Todos

Mostrar: 20 << < 1 2 3 ... > >>

| Título | Página | País | Categoría | Acciones |
|---|----------------------------|---------|-----------|----------|
| <input type="checkbox"/> Universidad Politécni... | https://www.ups.edu.ec/ | Ecuador | Educativa | |
| <input type="checkbox"/> UDA | https://www.uazuay.edu.ec/ | Ecuador | Educativa | |
| <input type="checkbox"/> Universidad Católica | https://www.ucacue.edu.ec/ | Ecuador | Educativa | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Página 1 de 5 << < 1 2 3 ... > >>



UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR

Anexo 3. Prototipo: Previsualización de correo electrónico de respuesta (Administrador)

Solicitudes de Registro

http://observatorioweb.ups.edu.ec/owa/admin/sites/manage

OAW Ecuador Inicio Analizador Web Acerca de API Rest Contactos Cerrar Sesión

Previsualización del Correo Electrónico

Se puede editar los siguientes datos:

Estimado Juan Pérez,

Nos complace comunicarle el estado del registro de las siguientes páginas web:

Título: Universidad Politécnica Salesiana.

País: Ecuador

Categoría: Educación

Descripción:
Páginas pertenecientes al ambiente educativo de la Universidad Politécnica Salesiana.

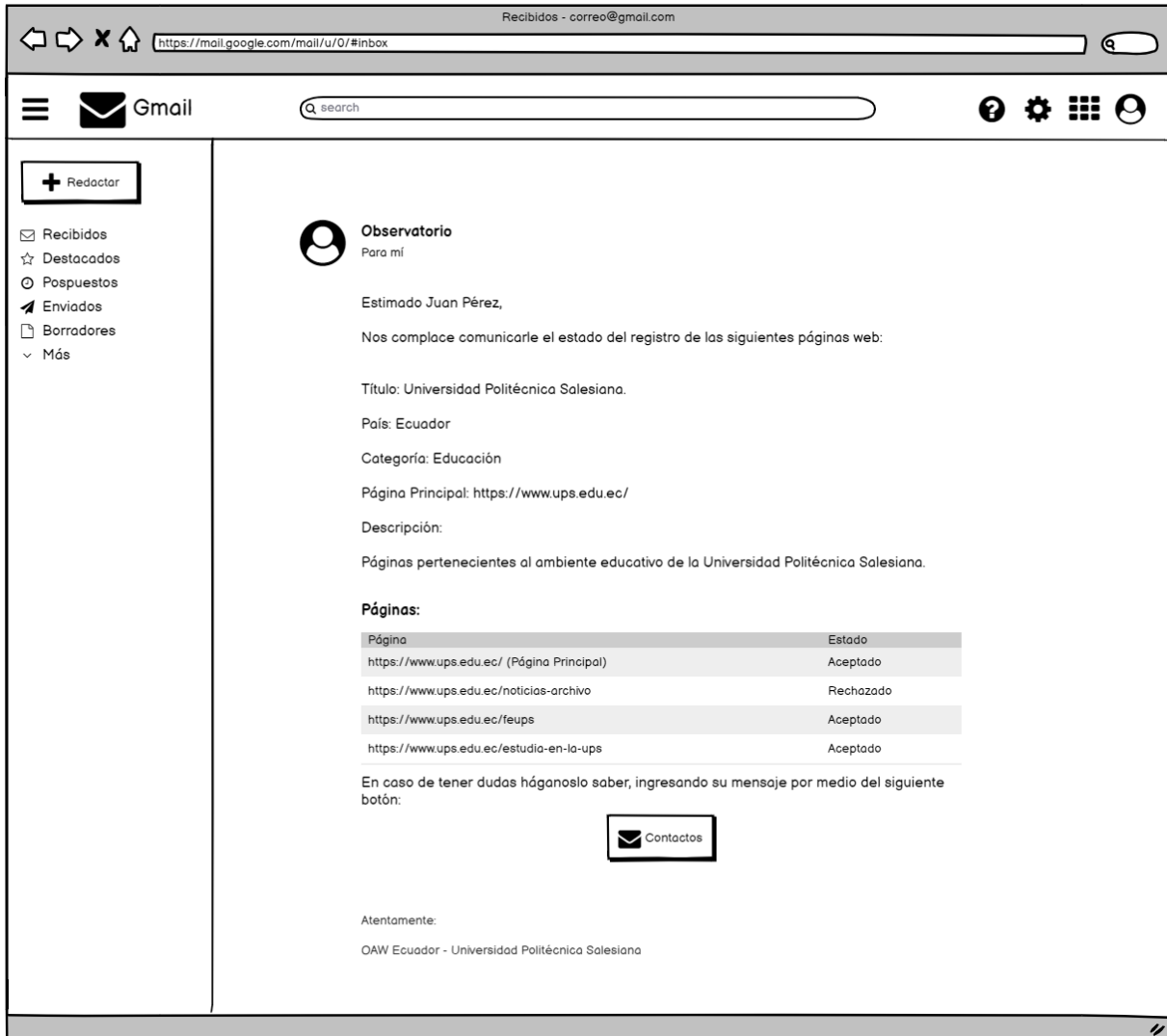
| Página | Estado |
|--|-----------|
| https://www.ups.edu.ec/ (Página Principal) | Aceptado |
| https://www.ups.edu.ec/noticias-archivo | Rechazado |
| https://www.ups.edu.ec/feups | Aceptado |

En caso de tener dudas háganoslo saber, ingresando su mensaje por medio del siguiente botón:

Atentamente:
OAW Ecuador - Universidad Politécnica Salesiana

UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR

Anexo 4. Prototipo: Ejemplo de correo electrónico de respuesta (Usuario)



1. Clonar el repositorio del proyecto en la máquina local y el servidor.

```
Manual de Despliegue - Directorios  
git clone https://github.com/user/repositorio.git /opt/source-path
```

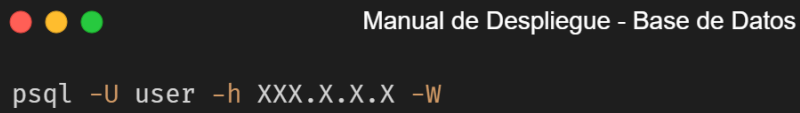
Nota: El repositorio en la máquina local servirá para compilar el *backend* con Maven y subirlo al servidor.

2. Crear los directorios de configuración en el servidor.

Se deberá crear un directorio en donde se guardarán los módulos con los ejecutables para el análisis de conformidad y de contraste, por lo que se crearán dos enlaces simbólicos dentro de este.

```
Manual de Despliegue - Directorios  
  
# Path para los módulos  
mkdir /opt/modules-path  
  
cd /opt/source-path  
  
# Creación de enlaces simbólicos  
ln -s $PWD/path-conformance /opt/modules-path/conformance  
ln -s $PWD/path-contrast /opt/modules-path/contrast
```

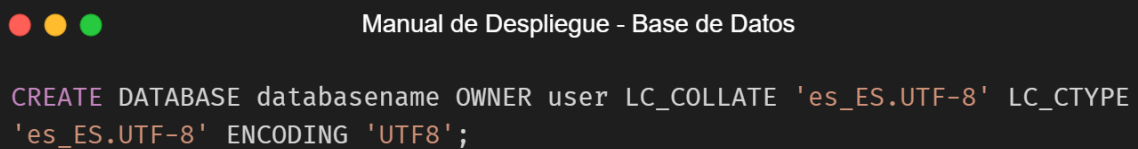

1. Acceder a PostgreSQL.



```
Manual de Despliegue - Base de Datos  
psql -U user -h XXX.X.X.X -W
```

2. Crear una base de datos.

Se creará la base de datos en donde se guardarán todos los datos relacionados con los análisis de las páginas. Es importante indicar explícitamente que se usará UTF-8 como formato de codificación.



```
Manual de Despliegue - Base de Datos  
CREATE DATABASE databasename OWNER user LC_COLLATE 'es_ES.UTF-8' LC_CTYPE  
'es_ES.UTF-8' ENCODING 'UTF8';
```

1. Descargar y descomprimir WildFly.

```
Manual de Despliegue - WildFly

# Descarga
wget
https://github.com/wildfly/wildfly/releases/download/26.1.1.Final/wildfly-26.1.1.Final.zip

# Descompresión
unzip wildfly-26.1.1.Final.zip
```

2. Copiar el contenido al directorio “opt”.

Una vez se haya realizado la copia del contenido, es recomendable crear un enlace simbólico para así poder controlar mejor las versiones del software a futuro.

```
Manual de Despliegue - WildFly

# Creación de la copia en OPT
cp -r wildfly-26.1.1.Final /opt/wildfly-26.1.1.Final

# Creación del enlace simbólico
ln -s /opt/wildfly-26.1.1.Final wildfly-26
```

3. Crear el directorio para la configuración del controlador de base de datos.

```
Manual de Despliegue - WildFly

# Dirigirse al directorio ORG
cd /opt/wildfly-26/modules/system/layers/base/org

# Creación del directorio para el controlador
mkdir -p postgresql/main

# Dirigirse al directorio creado
cd $_
```

4. Dentro de la carpeta “main” anteriormente creada, tener los siguientes archivos:

- El controlador JDBC descargado.

```
Manual de Despliegue - WildFly
wget https://jdbc.postgresql.org/download/postgresql-42.4.0.jar
```

- El archivo “module.xml”. Cabe resaltar que, dependiendo de la versión del controlador descargado, se deberá cambiar el valor del path en “resource-root”.

```
Manual de Despliegue - WildFly
<?xml version='1.0' encoding='UTF-8'?>
<module xmlns="urn:jboss:module:1.1" name="org.postgresql">

  <resources>
    <resource-root path="postgresql-42.4.0.jar" />
  </resources>

  <dependencies>
    <module name="javax.api" />
    <module name="javax.transaction.api" />
  </dependencies>
</module>
```

5. Agregar la configuración al archivo “/opt/wildfly-26/standalone/configuration/standalone.xml”.

Dentro de este archivo se establecerán las configuraciones para que WildFly se conecte con la base de datos. Para ello se deberá especificar la dirección URL y el puerto en donde se encuentre la base de datos.

Además, se deberá establecer las credenciales de un usuario que tenga permisos sobre la base de datos a utilizarse.

```
Manual de Despliegue - WildFly

<!--
Agregar dentro de:
<subsystem xmlns="urn:jboss:domain:datasources:7.0">
  <datasource jndi-name="java:jboss/datasources/NameDS" pool-name="poolDS"
  enabled="true" use-java-context="true">
    <connection-
url>jdbc:postgresql://localhost:PORT/databasename</connection-url>
    <driver>postgresql</driver>
    <security>
      <user-name>USERNAME</user-name>
      <password>PASSWORD</password>
    </security>
    <validation>
      <valid-connection-checker class-
name="org.jboss.jca.adapters.jdbc.extensions.postgres.PostgreSQLValidConnec-
tionChecker" />
      <exception-sorter class-
name="org.jboss.jca.adapters.jdbc.extensions.postgres.PostgreSQLExceptionSo-
rter" />
    </validation>
  </datasource>
  <driver name="postgresql" module="org.postgresql">
    <driver-class>org.postgresql.Driver</driver-class>
  </driver>
-->
```

6. Copiar los archivos de configuración de WildFly 26 (CentOS 7).

```
Manual de Despliegue - WildFly

mkdir -p /etc/wildfly

# Se copian las configuraciones
cp /opt/wildfly-26/docs/contrib/scripts/systemd/wildfly.conf
/etc/wildfly/wildfly-26.conf

# Se copia el launch
cp /opt/wildfly-26/docs/contrib/scripts/systemd/launch.sh /opt/wildfly-
26/bin/

# Se asignan permisos
sh -c 'chmod +x /opt/wildfly-26/bin/*.sh'

# Se copia el servicio
cp /opt/wildfly-26/docs/contrib/scripts/systemd/wildfly.service
/etc/systemd/system/wildfly-26.service
```

7. Configurar el archivo “/etc/wildfly/wildfly-26.conf”.

En este caso, se dejaron los valores por defecto, y solo se cambiaron los valores de “HOME” y “BIND”, según se configuró anteriormente.

```
Manual de Despliegue - WildFly

# Path del directorio de WildFly
WILDFLY_HOME=/opt/wildfly-26/

# El perfil de configuración
WILDFLY_CONFIG=standalone.xml

# El modo de despliegue
WILDFLY_MODE=standalone

# La dirección IP
WILDFLY_BIND=XXX.X.X.X
```

8. Configurar el archivo “/opt/wildfly-26/bin/launch.sh”.

Dentro de este archivo se deja todo por defecto, excepto el primer “WILDFLY_HOME”, con la ubicación real donde se encuentran todos los archivos necesarios.

```
Manual de Despliegue - WildFly

if [ "x$WILDFLY_HOME" = "x" ]; then
    WILDFLY_HOME="/opt/wildfly-26"
fi

if [[ "$1" = "domain" ]]; then
    $WILDFLY_HOME/bin/domain.sh -c $2 -b $3
else
    $WILDFLY_HOME/bin/standalone.sh -c $2 -b $3
fi
```

9. Configurar el archivo “/etc/systemd/system/wildfly-26.service”.

En este archivo se cambian los paths: “EnvironmentFile”, “PIDFile”, “ExecStart” según la configuración realizada anteriormente. Además se cambia el usuario a “root”.

```
Manual de Despliegue - WildFly

[Unit]
Description=The WildFly 26 Application Server
After=syslog.target network.target
Before=nginx.service

[Service]
Environment=LAUNCH_JBOSS_IN_BACKGROUND=1
EnvironmentFile=-/etc/wildfly/wildfly-26.conf
User=root
LimitNOFILE=102642
PIDFile=/var/run/wildfly/wildfly-26.pid
ExecStart=/opt/wildfly-26/bin/launch.sh $WILDFLY_MODE $WILDFLY_CONFIG
$WILDFLY_BIND
StandardOutput=null

[Install]
WantedBy=multi-user.target
```

10. Ejecutar las configuraciones para activar el marco de seguridad Elytron.

A fin de implementar seguridad dentro de la herramienta web, se necesita ejecutar las configuraciones de Elytron. Para ello, se utiliza el archivo “configure-elytron.cli”.

```
Manual de Despliegue - WildFly

# Acceso al proyecto
cd /opt/source-path/backend

# Ejecución de las configuraciones Elytron
/opt/wildfly-26/bin/jboss-cli.sh --connect --file=configure-elytron.cli
```

El contenido y las configuraciones del archivo descrito, es el siguiente:

```
## Consulta para validar el login del usuario y obtener su rol
## Además, se establece el algoritmo de cifrado de las contraseñas
/subsystem=elytron/jdbc-realm=jdbcRealm:add(principal-query=[{sql="SELECT
u.password, u.role from user u where u.username = ? and u.enabled = 1
",data-source="nameDS",simple-digest-mapper={password-index="1",
algorithm="simple-digest-sha-512", hash-encoding="hex"}, attribute-
mapping=[{index=2,to=groups}]}])

## Creación el sistema de administración (Realm) para validar los Roles
/subsystem=elytron/security-domain=jdbcdomain:add(default-
realm=jdbcRealm,realms=[{realm="jdbcRealm", role-decoder=groups-to-
roles}], default-realm=jdbcRealm, permission-mapper=default-permission-
mapper)

## Creación del dominio de seguridad para los subsistemas EJB3
/subsystem=ejb3/application-security-domain=nameSD:add(security-
domain=jdbcdomain)

## Creación de la autenticación HTTP
/subsystem=elytron/http-authentication-factory=http-db-auth:add(security-
domain=jdbcdomain,mechanism-configurations=[{mechanism-
name=BASIC,mechanism-realm-configurations=[realm-name="jdbcRealm"}]},http-
server-mechanism-factory=global)

## Configuración del método de autenticación
/subsystem=undertow/server=default-server/host=default-host/setting=http-
invoker:write-attribute(name=http-authentication-factory, value=http-db-
auth)

## Creación del dominio de seguridad para aplicaciones para permitir el
login con HTTP
/subsystem=undertow/application-security-domain=nameSD:add(http-
authentication-factory=http-db-auth)

## Actualizar y cargar las configuraciones realizadas
reload
```

11. Instalar NodeJS (NodeJS: 14.17.3).

```
Manual de Despliegue - WildFly

# Dirigirse al directorio OPT
cd /opt/

# Descarga de Node
wget https://nodejs.org/dist/v14.17.3/node-v14.17.3-linux-x64.tar.xz

# Descompresión y eliminación del archivo
tar -xf node-v14.17.3-linux-x64.tar.xz
rm node-v14.17.3-linux-x64.tar.xz

# Renombrar la carpeta
mv node-v14.17.3-linux-x64 node14
```

12. Inicializar el servicio.

```
Manual de Despliegue - WildFly

# Carga las configuraciones
systemctl daemon-reload

# Inicia el servicio
systemctl start wildfly-26
```


1. Configurar el archivo “/etc/nginx/nginx.conf”.

Aquí se agregan configuraciones de proxy para agregar seguridad al servidor web. Además, se define la ruta con el que se enlazan las cookies generadas en el servidor de aplicaciones.

```
Manual de Despliegue - Nginx

server {
    . . .
    root /usr/share/nginx/html;

    location /owa/ {
        try_files $uri /owa/index.html;
    }

    location /api/owa/ {
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
        proxy_set_header X-Forwarded-Port $server_port;
        proxy_cookie_path ~*^/* /api/owa/;
        proxy_pass http://localhost:PORT/owa/;
    }
    . . .
}
```

2. Reiniciar el servicio.

```
Manual de Despliegue - Nginx

systemctl restart nginx
```

1. Compilar el backend del proyecto de manera local.

Los parámetros necesarios en las propiedades secretas de configuración están documentados dentro de “/src/main/resources/config.properties”.

```
Manual de Despliegue - Backend

# Configurar el proyecto con los parámetros necesarios en:
# /src/main/resources/config.properties
# /src/main/resources/config-secret.properties
# dentro de la máquina local
cd /opt/owa-src/backend

# Compilación con Maven
mvn clean install
mvn package
```

2. Subir el archivo “WARNAME.war” al servidor de aplicaciones.

```
Manual de Despliegue - Backend

scp ./target/WARNAME.war <user>@<host>:/opt/wildfly-
26/standalone/deployments/WARNAME.war
```

1. Compilar el proyecto en Angular.

```
Manual de Despliegue - Frontend

cd /opt/source-path/frontend

# Configuración previa
npm config set legacy-peer-deps true

# Instalación de módulos
npm i

# Compilación con Angular
ng build --prod --base-href /owa/ --deploy-url /owa/

# Subida del proyecto al servidor
# (Optional, caso contrario, compilar en el servidor)
cd dist
scp -r owa <user>@<host>:/usr/share/nginx/html
```

2. Asignar permisos al proyecto.

```
Manual de Despliegue - Frontend

# Acceso a la carpeta por defecto de nginx
cd /usr/share/nginx/html

# Permisos necesarios
chmod -R 0755 owa/
```

1. Instalar el módulo de contraste en un directorio del servidor.

```
Manual de Despliegue - Conformidad

cd /opt/modules-path/conformance

# Instalación
npm i

# Test para probar si está funcionando correctamente
/opt/node14/bin/node /opt/modules-path/conformance/analyzer.js
http://google.com 1366 768 /opt/modules-path/conformance-images/
```

1. Compilar el módulo de Contraste con CMake.

```
Manual de Despliegue - Contraste

cd /opt/modules-path/contrast

# Creación del directorio "build"
mkdir -p build
cd $_

# Crear los archivos de configuración con CMAKE
/usr/bin/cmake --no-warn-unused-cli \
-DCMAKE_EXPORT_COMPILE_COMMANDS:BOOL=TRUE \
-DCMAKE_BUILD_TYPE:STRING=Release \
..

# Instalación del script cont
cmake --build /opt/modules-path/contrast/build --config Release --target
all

# La salida del script estará en
# /opt/modules-path/contrast/cont
```

2. Desplegar el “contrast-backend”.

```
Manual de Despliegue - Contraste

# Acceso al directorio
cd /opt/source-path/contrast-backend

# Instalación de módulos
npm i

# Compilación en producción
npm run build-prod

# Despliegue
pm2 start dist/server.js --name "contrast-api"
```