



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

DESARROLLO DE PLATAFORMA WEB PARA LA EVALUACIÓN DE
SOFTWARE BASADOS EN LA METODOLOGÍA SCRUM

SARANGO YUNGA DARIO XAVIER
INGENIERO DE SISTEMAS

MACHALA
2020



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

Desarrollo de plataforma web para la evaluación de software
basados en la metodología SCRUM

SARANGO YUNGA DARIO XAVIER
INGENIERO DE SISTEMAS

MACHALA
2020



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

TRABAJO TITULACIÓN
PROPUESTAS TECNOLÓGICAS

Desarrollo de plataforma web para la evaluación de software basados en la
metodología SCRUM

SARANGO YUNGA DARIO XAVIER
INGENIERO DE SISTEMAS

VALAREZO PARDO MILTON RAFAEL

MACHALA, 01 DE MAYO DE 2020

MACHALA
2020

Desarrollo de plataforma web para la evaluación de software basados en la metodología SCRUM

INFORME DE ORIGINALIDAD

10%

INDICE DE SIMILITUD

6%

FUENTES DE
INTERNET

1%

PUBLICACIONES

6%

TRABAJOS DEL
ESTUDIANTE

FUENTES PRIMARIAS

1

oa.upm.es

Fuente de Internet

<1%

2

Piedad Chilito, Daniel Viveros, Cesar Pardo, Francisco J. Pino. "Scrum+: An agile guide for the global software development (GSD) multi-model project management", 2018 IEEE

Colombian Conference on Communications and Computing (COLCOM), 2018

Publicación

<1%

3

Submitted to Universidad Senor de Sipan

Trabajo del estudiante

<1%

4

Submitted to Universidad Ort

Trabajo del estudiante

<1%

5

www.docstoc.com

Fuente de Internet

<1%

6

recibe.cucei.udg.mx

Fuente de Internet

<1%

CLÁUSULA DE CESIÓN DE DERECHO DE PUBLICACIÓN EN EL REPOSITORIO DIGITAL INSTITUCIONAL

El que suscribe, SARANGO YUNGA DARIO XAVIER, en calidad de autor del siguiente trabajo escrito titulado Desarrollo de plataforma web para la evaluación de software basados en la metodología SCRUM, otorga a la Universidad Técnica de Machala, de forma gratuita y no exclusiva, los derechos de reproducción, distribución y comunicación pública de la obra, que constituye un trabajo de autoría propia, sobre la cual tiene potestad para otorgar los derechos contenidos en esta licencia.

El autor declara que el contenido que se publicará es de carácter académico y se enmarca en las disposiciones definidas por la Universidad Técnica de Machala.

Se autoriza a transformar la obra, únicamente cuando sea necesario, y a realizar las adaptaciones pertinentes para permitir su preservación, distribución y publicación en el Repositorio Digital Institucional de la Universidad Técnica de Machala.

El autor como garante de la autoría de la obra y en relación a la misma, declara que la universidad se encuentra libre de todo tipo de responsabilidad sobre el contenido de la obra y que asume la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.

Aceptando esta licencia, se cede a la Universidad Técnica de Machala el derecho exclusivo de archivar, reproducir, convertir, comunicar y/o distribuir la obra mundialmente en formato electrónico y digital a través de su Repositorio Digital Institucional, siempre y cuando no se lo haga para obtener beneficio económico.

Machala, 01 de mayo de 2020



SARANGO YUNGA DARIO XAVIER
0750179004



UNIVERSITAS
MAGISTRO-
RUM
ET SCHOLARIUM

DEDICATORIA

Este trabajo significa una evidencia de mi trayectoria en la Carrera de Ingeniería de Sistemas, por esto quiero dedicarlo a mi familia quienes me brindaron su apoyo incondicional durante esta etapa. A mis padres que fueron mi sustento y principal motivación para alcanzar una meta más, colocando su confianza sobre mí en cada paso que doy. A mis hermanos siendo confidentes y amigos siempre.

Dedico este trabajo a todas aquellas personas que brindaron su confianza sobre mí, brindándome su respeto y me incentivaron a cumplir cada objetivo que me propuse.

AGRADECIMIENTO

Quiero empezar agradeciendo a mi madre, por ser una amiga y guía durante mi etapa universitaria, por siempre estar presente en los momentos buenos y malos, por sus consejos y confianza puestos sobre mí, y por todo su amor incondicional en todo momento. A mi padre por infundir el respeto y la responsabilidad, y que a pesar de cualquier circunstancia debemos esforzarnos y seguir adelante dando lo mejor de uno mismo.

Agradezco a la Universidad Técnica de Machala, que con sus autoridades y gestión permitieron que mi transcurso en sus instalaciones sea placentero. A los docentes quienes gracias a sus enseñanzas lograron incrementar mis facultades académicas para así obtener el título universitario como Ingeniera de Sistemas.

RESUMEN

Las tecnologías se han convertido en parte fundamental de las personas, siendo herramientas de trabajo, progreso y de constante innovación, así también la actualización continua del hardware, conllevando al crecimiento del software, permitiendo un correcto funcionamiento y manejo de los recursos, satisfaciendo las necesidades que demanda la sociedad. De igual manera las empresas y organizaciones hacen uso de software como herramientas para mejorar la optimización de procesos, manejo de recursos y reducción de costos en las distintas áreas de la organización; es por eso que en la actualidad el desarrollo de software ha adquirido una mayor importancia mejorando el proceso de cómo se realiza, y estratégicamente obteniendo una mejor competitividad dentro de diferentes empresas u organizaciones.

Debido a esto, surgen las metodologías para el desarrollo de software cuyo propósito es que el producto final sea eficiente y de mejor calidad, proponiendo cada una de ellas distintas etapas, fases y componentes, buscando llegar al mismo objetivo, entre estas metodologías se ubica SCRUM, planteada como una herramienta que permite el trabajo en equipo de un conjunto de iteraciones o ciclos de trabajo también llamado Sprint. Es necesario realizar una evaluación del cumplimiento de la metodología luego de culminar el desarrollo del proyecto, con el fin de tener mejores resultados y que cumplan con los estándares de calidad, sin embargo, este proceso llega a ser extenso y tedioso, afectando a la realización del proyecto e influyendo en el cronograma planificado. Ante lo expuesto se realizará una plataforma web para la evaluación de software basados en la metodología SCRUM, mejorando el rendimiento del equipo de trabajo y el cumplimiento de las fases de la metodología en dicho proyecto.

Para el desarrollo de la plataforma web se hace uso de tecnologías y lenguajes modernos como son JavaScript, lenguaje que sirve de base para la mayoría de páginas y aplicaciones web; Node.js, el cual es un entorno para la parte del servidor en la web que permite la funcionalidad de todo lo que el usuario no observa como procesos lógicos y cálculos; los Frameworks como Express, que complementa a node.js para realizar múltiples configuraciones como la creación de las rutas de la página web, y Vue.js el cual se usa para elaborar las interfaces del usuario y es complementado, con Vuetify que es un potenciador que tiene varias interfaces codificadas, simples de usar; y para el almacenamiento, el uso de MongoDB, una base

de datos NoSQL que realiza el almacenamiento de datos en documentos y no en registros.

El principal objetivo del presente proyecto es la implementación de una plataforma web para que el usuario lleve a cabo una evaluación subjetiva de las fases y componentes necesarios en el desarrollo de software bajo los estándares de la metodología SCRUM, identificando los distintos niveles dependiendo del tamaño del mismo, mediante el planteamiento de preguntas necesarias por cada una de las fases planificación, desarrollo, revisión y retrospectiva; para llegar a una conclusión respecto al cumplimiento de la metodología, permitiendo así al usuario optimizar el tiempo que conlleva este proceso de evaluación y revelando posibles faltas ocurridas durante el desarrollo del software.

Palabras claves:

Scrum, Metodología de Software, Evaluación de Software, Desarrollo Web, Software

ABSTRACT

The technologies have become a fundamental part of the people, being tools of work, progress and constant innovation, as well as the continuous updating of the hardware, leading to the growth of the software, allowing a correct operation and management of the resources, satisfying the needs that society demands. Similarly, companies and organizations make use of software as tools to improve process optimization, resource management and cost reduction in different areas of the organization; That is why software development has become more important today, improving the process of how it is done, and strategically obtaining better competitiveness within different companies or organizations.

Due to this, the methodologies for the development of software arise whose purpose is that the final product is efficient and of better quality, proposing each of them different stages, phases and components, seeking to reach the same objective, among these methodologies is SCRUM , raised as a tool that allows teamwork of a set of iterations or work cycles also called Sprint. It is necessary to carry out an evaluation of the fulfillment of the methodology after completing the development of the project, in order to have better results and that meet the quality standards, however, this process becomes extensive and tedious, affecting the completion of the project and influencing the planned schedule. Given the above, a web platform for the evaluation of software based on the SCRUM methodology will be carried out, improving the performance of the work team and the fulfillment of the phases of the methodology in said project.

For the development of the web platform, current technologies and languages are used, such as JavaScript, which is the basis for most web pages and applications; Node.js, which is an environment for the part of the server on the web that allows the functionality of everything that the user does not observe as logical processes and calculations; Frameworks as Express, which complements node.js to perform multiple configurations such as the creation of web page paths, and Vue.js which is used to develop user interfaces and is complemented, with Vuetify which is an enhancer which has several coded interfaces, simple to use; and for storage, the use of MongoDB, a NoSQL database that performs data storage in documents and not in records.

The main objective of this project is the implementation of a web platform for the user to carry out a subjective evaluation of the phases and components necessary in the development of software under the standards of the SCRUM methodology, identifying the different levels depending on the size of the same, by raising the necessary questions for each of the planning, development, review and retrospective phases; to reach a conclusion regarding compliance with the methodology, thus allowing the user to optimize the time involved in this evaluation process and revealing possible failures occurred during the development of the software.

Keywords:

Scrum, Software Methodology, Software Evaluation, Web Development, Software

CONTENIDO

DEDICATORIA	1
AGRADECIMIENTO	2
RESUMEN	3
ABSTRACT	5
INTRODUCCIÓN	10
1. CAPÍTULO I. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS	12
1.1. Ámbito de aplicación: descripción del contexto y hechos de interés	12
1.2. Establecimiento de requerimientos	13
1.3. Justificación del requerimiento a satisfacer	14
2. CAPÍTULO II. DESARROLLO DEL PROYECTO	15
2.1. Definición del prototipo tecnológico	15
2.1.1. Capa Cliente	16
2.1.2. Capa Servidor Web	16
2.1.3. Capa de Datos	17
2.2. FUNDAMENTACIÓN TEÓRICA DEL PROTOTIPO	17
2.2.1. La industria del desarrollo de software	17
2.2.2. Aplicaciones web	18
2.2.3. Metodologías de desarrollo de software	18
2.2.4. Metodologías Tradicionales	18
2.2.5. Metodologías Ágiles	19
2.2.6. SCRUM	19
2.2.6.1. Principios	20
2.2.6.2. Características	21
2.2.6.3. Equipo Scrum	21
2.2.6.4. Sprint	22
2.2.6.5. Procesos Scrum	22
2.2.6.5.1. Planificación	22
2.2.6.5.1.1. Planificación del Sprint	22
2.2.6.5.2. Desarrollo	22
2.2.6.5.2.1. Scrum Diario.	22
2.2.6.5.2.2. Refinamiento del Backlog	22
2.2.6.5.3. Revisión	23
2.2.6.5.4. Retrospectiva	23
2.2.7. Plataforma web	23
2.2.8. JavaScript	23
2.2.8.1. Node.js	24
2.2.8.2. Express	24
2.2.8.3. Vue.js	24
2.2.9. Base de datos NoSQL	24

2.2.9.1.	MongoDB	24
2.3.	OBJETIVOS DEL PROTOTIPO	26
2.3.1.	Objetivo General	26
2.3.2.	Objetivos Específicos	26
2.4.	DISEÑO DEL PROTOTIPO	27
2.4.1.	Prototipo de interfaces de usuario	28
2.5.	EJECUCIÓN Y/O ENSAMBLAJE DEL PROTOTIPO	33
2.5.1.	Pantalla principal	33
2.5.2.	Registro de usuarios	34
2.5.3.	Inicio de sesión de usuario	34
2.5.4.	Pantalla de inicio	35
2.5.5.	Configuración de la cuenta	35
2.5.6.	Creación de evaluaciones	36
2.5.7.	Evaluaciones del usuario	37
2.5.8.	Edición de evaluación	38
2.5.9.	Pantalla de evaluación	39
2.5.10.	Resultados de evaluación	41
3.	CAPÍTULO III. EVALUACIÓN DEL PROTOTIPO	43
3.1.	PLAN DE EVALUACIÓN	43
3.1.1.	Evaluación de calidad	43
3.2.	RESULTADOS DE LA EVALUACIÓN	44
3.2.1.	Validator HTML	44
3.2.2.	Validator CSS	44
3.2.3.	Funcional Accessibility Evaluator 2.0	44
3.2.4.	GtMetrix	44
3.2.5.	Moz Bar	45
3.2.6.	Metricspot	45
3.2.7.	Websitte grader	45
3.2.8.	ReadyMobi	45
3.3.	CONCLUSIONES	46
3.4.	RECOMENDACIONES	47
	BIBLIOGRAFÍA	48
	ANEXOS	52

ÍNDICE DE TABLAS

Tabla 1. Comparación de metodologías.	19
---------------------------------------	----

ÍNDICE DE FIGURAS

Figura 1. Arquitectura del Modelo Cliente-Servidor	15
Figura 2. Funcionamiento del Modelo Cliente-Servidor	16
Figura 3. Mapa conceptual de la Arquitectura del prototipo	17
Figura 4. Roles, artefactos y eventos principales de SCRUM	20
Figura 5. Arquitectura de la plataforma web.	23
Figura 6. Prototipo de interfaz de usuario - Pantalla principal	28
Figura 7. Prototipo de interfaz de usuario - Pantalla de inicio de sesión	29
Figura 8 . Prototipo de interfaz de usuario - Pantalla de registro de usuario	29
Figura 9. Prototipo de interfaz de usuario - Pantalla de inicio de la plataforma	30
Figura 10. Prototipo de interfaz de usuario - Formulario para agregar evaluaciones	30
Figura 11. Prototipo de interfaz de usuario - Pantalla de evaluaciones del usuario	31
Figura 12. Prototipo de interfaz de usuario - Formulario de edición de evaluación	31
Figura 13. Prototipo de interfaz de usuario - Pantalla de evaluación	32
Figura 14. Prototipo de interfaz de usuario - Formulario de configuración de la cuenta	32
Figura 15. Ejecución del prototipo - Pantalla principal	33
Figura 16. Ejecución del prototipo - Patanlla de registro de usuario	34
Figura 17. Ejecución del prototipo - Pantalla de inicio de sesión	34
Figura 18. Ejecución del prototipo - Pantalla de inicio	35
Figura 19. Ejecución del prototipo - Formulario de configuración de la cuenta	35
Figura 20. Ejecución del prototipo - Menú del usuario	36
Figura 21. Ejecución del prototipo - Formulario para agregar evaluación	36
Figura 22. Ejecución del prototipo - Pantalla de evaluaciones	37
Figura 23. Ejecución del prototipo - Listado de evaluaciones	37
Figura 24. Ejecución del prototipo - Vista de evaluación en proceso	38
Figura 25. Ejecución del prototipo - Vista de evaluación finalizada	38
Figura 26. Ejecución del prototipo - Formulario de edición de evaluación	38
Figura 27. Ejecución del prototipo - Pantalla de evaluación	39
Figura 28. Ejecución del prototipo - Opciones de calificación	39
Figura 29. Ejecución del prototipo – Controles paginación de evaluación	40
Figura 30. Ejecución del prototipo - Controles de evaluación en proceso	40
Figura 31. Ejecución del prototipo - Controles de evaluación finalizada	40
Figura 32. Ejecución del prototipo - Visualización de evaluaciones finalizadas	41
Figura 33. Ejecución del prototipo - Pantalla de resultados de evaluación	41
Figura 34. Ejecución del prototipo - Información de evaluación	42
Figura 35. Ejecución del prototipo - Calificación del cumplimiento de la metodología	42
Figura 36. Ejecución del prototipo - Controles de visualización de resultados	42
Figura 37. Ejecución del prototipo - Tabla de resultados	42
Figura 38. Ejecución del prototipo - Gráfico de resultados	43

INTRODUCCIÓN

La tecnología juega un papel importante, aportando beneficios significativos para la humanidad, de tal manera que se la considera como pilar fundamental para el avance de la sociedad, debido a esto se requiere de una constante producción de dispositivos tecnológicos y software que se adapte a las necesidades de los usuarios, las cuales actualmente son muy diversificadas, desde plataformas de entretenimiento hasta sistemas relacionados a distintas áreas de salud.

La alta necesidad de programas conlleva a que crear software requiere de un proceso más coordinado, que involucre diferentes responsabilidades y actividades repartidas a un equipo de trabajo, para ello surgen las metodologías de desarrollo de software, que ayudan a estos equipos a mejorar la comunicación, los tiempos de entrega y con ello la creación de un producto calificado para los usuarios finales.

El uso de una metodología para el desarrollo de software, asegura un mayor porcentaje de éxito del proyecto y aumento en la calidad del producto final, entre estas metodologías se encuentra Scrum, como una herramienta que permite trabajar a través de una serie de interacciones en equipo, para cumplir con los objetivos planteados y lograr mejores resultados.

Para asegurar que las interacciones que plantea Scrum se han llevado a cabo correctamente y que se está cumpliendo en su totalidad con la metodología, se requiere evaluar lo realizado, sin embargo, es un proceso que puede llegar a ser extenso, para lo cual es necesario el uso de herramientas que automaticen este proceso, y proporcionen un resultado en el cual basarse para tomar decisiones sobre qué mejorar en el desarrollo de nuevos proyectos.

Al ser Scrum una de las metodologías con mayor aceptación y uso en el área de desarrollo de software, la implementación de una plataforma que agilice el proceso de evaluar el cumplimiento de la metodología es necesaria, ya que ayuda a los equipos de trabajo a tomar decisiones acertadas sobre lo realizado, ya sea para mantener ciertos procesos o realizar cambios en cómo se desarrollan.

El presente informe describe el desarrollo e implementación de una plataforma web para la evaluación de software basados en la metodología Scrum, facilitando una

herramienta a personas relacionadas al desarrollo de software, automatizando el proceso de evaluar el cumplimiento de la metodología.

Este documento se encuentra estructurado de la siguiente forma.

En el **Capítulo 1**, se describe la necesidad de implementar la plataforma, describiendo de primer plano el enfoque del tema y sus antecedentes, seguido por los requerimientos y justificación, en los que se describe porque es importante dar solución a la necesidad planteada.

El **Capítulo 2**, describe la definición del prototipo, la fundamentación teórica, objetivos, diseño y ejecución del prototipo.

El **Capítulo 3**, se recopila información sobre el plan y los resultados de la evaluación, además se describen las conclusiones y recomendaciones que se obtuvieron a partir de la solución planteada.

1. CAPÍTULO I. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS

1.1. Ámbito de aplicación: descripción del contexto y hechos de interés

El desarrollo de software es un proceso de producción no tradicional, se considera, una actividad por completo intelectual y creativa, en donde el principal recurso es el capital humano especializado en diferentes tecnologías y el uso de herramientas que permitan una adecuada ejecución del proyecto. Sin embargo, la presencia o no de dicho personal no significa que nuevas compañías no ingresen a este espacio [1]. Este hecho se refleja en el impacto sobre la educación y el acceso a nuevos mercados alrededor del mundo, así como importantes avances en países en vías de desarrollo y economías alternas.

En el 2001 surge la metodología para el desarrollo de software Agile, resultado de una reunión de un grupo independiente de desarrolladores que no estaban a gusto con las metodologías existentes en aquel entonces [2]. Donde se da una mayor relevancia a los individuos, a hacer que el software funcione correctamente, la colaboración con el cliente y la respuesta a cambios [3].

Scrum es una metodología diseñada para el trabajo en equipo, basándose en ciclos de trabajo llamados "Sprint" y reuniones de trabajo, brindando así un enfoque ágil con mayores beneficios, con ello se obtiene mayor aceptación en la industria del desarrollo de software.

El software que está enfocado a la web debido a las diferentes características que deben cumplirse respecto a la duración del ciclo de vida, presenta un grado de dificultad en su desarrollo y diseño [4]. Una de los problemas que se presenta con mayor frecuencia es la accesibilidad de las aplicaciones, la cual se debe al desarrollo por igual del sistema y el equipo.

La presente propuesta tecnológica tiene como propósito desarrollar una plataforma web para la evaluación de software basados en la metodología Scrum, haciendo uso como lenguaje principal JavaScript, con sus Frameworks Node.js y Express para el servidor, MongoDB para el almacenamiento de datos y Vue.js para la creación de interfaces.

1.2. Establecimiento de requerimientos

Actualmente, se fuerza a que las compañías desarrollen sus productos y servicios en un tiempo menor, esto debido a las presiones por parte de las competencias; Para ello como una alternativa se presentan las metodologías de desarrollo ágil como Scrum y XP [5].

Aunque Scrum sea aplicado de manera exitosa, no existe una manera clara de cómo asegurar el cumplimiento total de la metodología, por lo cual es necesaria una plataforma, ya que facilitará a desarrolladores, grupos independientes u organizaciones a evaluar los procesos Scrum y sus categorías.

La plataforma para evaluar software basado en la metodología Scrum permitirá a desarrolladores, involucrados y personas afines al desarrollo de software evaluar de manera subjetiva el cumplimiento de la metodología de forma automatizada, durante o luego del proceso de desarrollo, ya sean proyectos a grande, mediana o pequeña escala.

La persona a cargo de la evaluación será quién valide la información que se ingresa, depende de cada uno que los datos ingresados sean los correctos. Al ser una evaluación subjetiva, la plataforma se encargará de la parte del análisis de los datos ingresados y expresarlos a través de una representación cuantitativa el nivel de cumplimiento de la metodología, permitiendo así tomar decisiones sobre los procesos con un déficit, mejorando el rendimiento del equipo de trabajo, tiempos de desarrollo, cumplimiento de la metodología y de los objetivos a corto y largo plazo.

1.3. Justificación del requerimiento a satisfacer

Con la implementación de las metodologías dentro del ciclo de vida de las aplicaciones Web se ha logrado no solo mejorar el proceso de creación y desarrollo de software, sino también se ha disminuido el nivel de riesgo, puesto que cada metodología ofrece una guía compuesta por etapas y procesos efectivos que permiten obtener resultados de calidad [6].

El desarrollo ágil necesita un mecanismo fundamental que aporte en los procesos de la metodología y apoye los ciclos que tiene para el desarrollo como una plataforma de desarrollo [7].

La presente propuesta tecnológica tiene como objetivo principal el desarrollo de una plataforma web que permita la evaluación de software basados en la metodología SCRUM. Permitiendo que personas relacionadas al desarrollo de software puedan llevar a cabo una comprobación del cumplimiento de la metodología, definiendo qué procesos o categorías se ven afectados y cuales cumplen en su totalidad con Scrum, ya sea durante o después del proceso de desarrollo del software.

2. CAPÍTULO II. DESARROLLO DEL PROYECTO

2.1. Definición del prototipo tecnológico

La arquitectura de la plataforma está basada en el modelo cliente-servidor, el cual permite el acceso de múltiples clientes que solicitan información a uno o más servidores centrales en un entorno multiplataforma.

Figura 1. Arquitectura del Modelo Cliente-Servidor

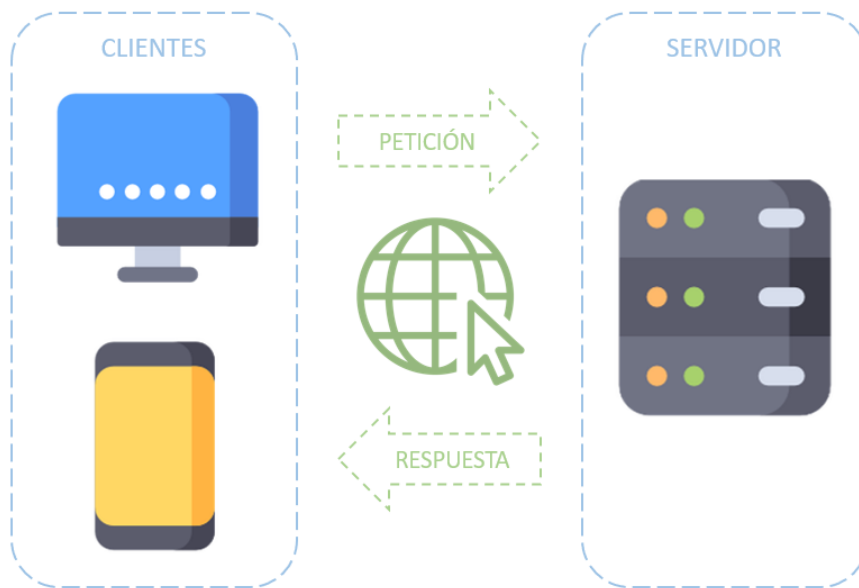


Fuente: Elaboración Propia

Basados en el modelo mencionado surge la figura 2 en la que se representa su funcionamiento.

1. El cliente enviará una petición al servidor por internet.
2. El servidor recibe la petición del cliente y la procesa.
3. El servidor enviará una respuesta al cliente por internet.
4. El cliente recibe el resultado y lo procesa.

Figura 2. Funcionamiento del Modelo Cliente-Servidor



Fuente: Elaboración propia

El objetivo principal del modelo es tratar de dar un permiso general de tal manera que el usuario logre acceder desde cualquier ordenador, reduciendo el mínimo de recursos de la plataforma. Una de las ventajas de esta arquitectura es que está basada en el modelo de tres capas que son: Presentación, Negocio y Datos, esto quiere decir que sus tareas son realizadas de forma independiente teniendo un elevado grado de flexibilidad y alto rendimiento.

2.1.1. Capa Cliente

Según [8], es considerada la capa final, capa cliente o capa de presentación la cual puede ser presentada en navegadores u otro tipo de formato, ciertos desarrolladores limitan sus sistemas a uno o dos navegadores, la comunicación es realizada por medio del protocolo HTTP.

Este será el punto intermedio entre los clientes y la plataforma web comunicándose mediante los métodos de envío de datos usando URL y envío de forma en segundo plano.

2.1.2. Capa Servidor Web

En esta capa el servidor web se encuentra diseñado para recibir peticiones por el protocolo HTTP, haciendo uso de los métodos GET, POST, PUT y DELETE

para enviar respuestas en formato JSON, implementando las tecnologías node.js y express.

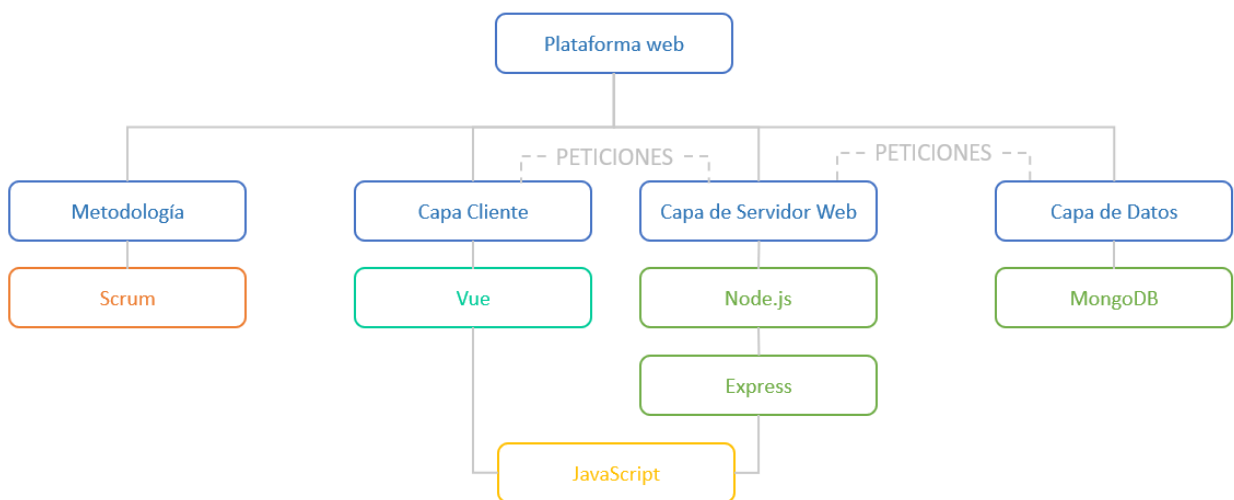
El método GET solicita una representación de un recurso específico. Para mayor seguridad no se debería usar en aplicaciones que puedan causar efectos, ya que se transmite información a través de la URL, haciendo uso de los parámetros en la misma [9].

2.1.3. Capa de Datos

Representa a la base de datos en sí misma, que se encuentra asociada a un Sistema Manejador de Base de Datos, la cual interactúa con la Capa Cliente a través de una conexión nativa de base de datos [10]. En la presente propuesta tecnológica se hace uso de la base de datos NoSQL MongoDB, agilizando la obtención de información y facilitando la incorporación con la Capa del servidor.

2.2. FUNDAMENTACIÓN TEÓRICA DEL PROTOTIPO

Figura 3. Mapa conceptual de la Arquitectura del prototipo



Fuente: Elaboración propia

2.2.1. La industria del desarrollo de software

Dentro de la industria, el desarrollo de software es un proceso mayormente colaborativo en el que su éxito depende de la adquisición de conocimiento, el intercambio de información y la minimización de la ruptura en el proceso de comunicación [1].

En la actualidad constituye la denominada economía de la información, generando el asentamiento para las tecnologías de la comunicación, siendo así para los países una de las contribuciones más grandes tanto a nivel de empleo, producción y economía.

2.2.2. Aplicaciones web

Según [11] se puede considerar como un sistema que generalmente se conforma por un servidor, una base de datos (o back-end) y la interfaz de usuario (el front-end), con el cual los usuarios interactúan a través de una red usando un navegador.

A diferencia de una página web común, las aplicaciones web hacen uso de tecnologías más robustas que permiten mayor interacción con el usuario, control de la información, mejor escalabilidad y rendimiento.

2.2.3. Metodologías de desarrollo de software

El desarrollo de software ha evolucionado, desde la forma de codificar y luego arreglar, en la que un equipo de desarrollo posee la idea general de lo que quiere realizar, pasando a metodologías formales, que por lo general son el conjunto de una planeación detallada del desarrollo, documentación, requerimientos, diseño de alto nivel e inspección [12], formando un marco de trabajo para apoyar los procesos que involucra el desarrollo de software.

2.2.4. Metodologías Tradicionales

Las metodologías tradicionales de desarrollo de software son orientadas por planeación. Inician el desarrollo de un proyecto con un riguroso proceso de elicitación de requerimientos, previo a etapas de análisis y diseño. Con esto tratan de asegurar resultados con alta calidad limitados a un calendario [13].

Se trabaja en un proyecto a la vez, el cual debe tener una estructura y dimensiones definidas, además el proceso a seguir debe ser secuencial sin opción a volver, es decir, avanzar en cuanto se haya finalizado en su totalidad cada proceso, los requerimientos se definen una sola vez para todo el proyecto, por lo que conlleva a largos tiempos de planeación y poca interacción con el cliente.

2.2.5. Metodologías Ágiles

Surgen como una alternativa a las tradicionales, priorizando los elementos que agregan más valor en el proceso de desarrollo del proyecto. Según [14] los principios de las metodologías ágiles son: El individuo y la interacción del equipo por encima del proceso y las herramientas; El software funcional por encima de la documentación; La colaboración con y del cliente por encima de la negociación del contrato; La respuesta a los cambios por encima del seguimiento a un plan.

A diferencia de las metodologías tradicionales de desarrollo de software, estas implementan nuevas formas de desarrollo, esto conlleva a que las empresas prefieran usarlas durante el ciclo de vida de un proyecto [15].

Tabla 1. Comparación de metodologías.

Metodologías Ágiles	Metodologías tradicionales
Se basan en heurísticas provenientes de prácticas de producción de código	Se basan en normas provenientes de estándares seguidos por el entorno de desarrollo
Preparados para cambios durante el proyecto	Cierta resistencia a los cambios
Impuestas internamente por el equipo	Impuestas externamente
Proceso menos controlado, con pocos principios	Proceso muy controlado, numerosas normas
Contrato flexible e incluso inexistente	Contrato prefijado
El cliente es parte del desarrollo	Cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (<10)	Grupos grandes
Pocos artefactos	Más artefactos
Menor énfasis en la arquitectura del software	La arquitectura del software es esencial

Fuente: [16]

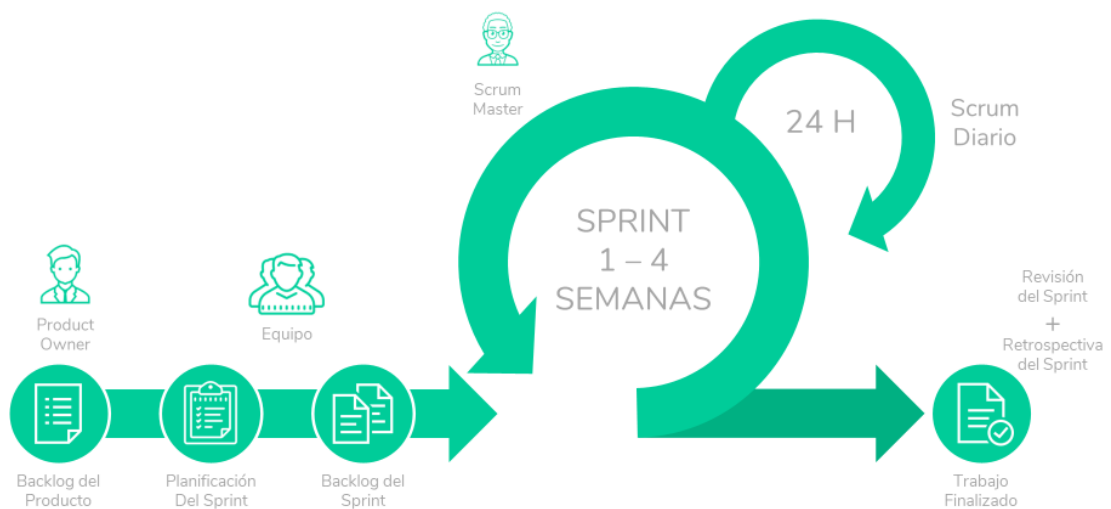
2.2.6. SCRUM

Se presenta como un marco de trabajo que presenta como característica principal el tener un desarrollo incremental e iterativo en cuanto a las funcionalidades que el usuario necesita; Donde los equipos de desarrollo tienen las destrezas para organizarse por sí mismos y gestionar sus propias tareas durante el proceso de desarrollo [17].

Se define a Scrum como una estructura donde las personas pueden abarcar problemas adaptables, con capacidad de producción y creatividad para obtener como resultado productos de gran valor, sin embargo, es una metodología ligera pero difícil de dominar. Según [18] las prácticas empleadas por SCRUM para mantener un control ágil en el proyecto son:

- Revisión de las iteraciones
- Desarrollo incremental
- Desarrollo evolutivo
- Autoorganización del equipo
- Colaboración

Figura 4. Roles, artefactos y eventos principales de SCRUM



Fuente: Elaboración propia

2.2.6.1. Principios

Basados en [19] los principios con los que se ha desarrollado Scrum son:

- El principio de incertidumbre de Ziv en la ingeniería del software: la incertidumbre es inherente e inevitable para los equipos de trabajo durante el proceso de desarrollo de software.

- Principio de requisitos indefinidos de Humphrey: los requerimientos de un sistema nuevo no se conocerán del todo hasta que un usuario no lo haya probado.

2.2.6.2. Características

Las características principales del proceso Scrum son:

- La primera y última fase, requieren que los procesos, entradas y salidas estén bien definidos. El conocimiento de cómo realizar los procesos es claro y se intenta plantear una recopilación de todas las actividades a realizar [19], a esto se lo conoce en Scrum como Backlog.
- Se desarrollan iteraciones llamadas Sprint, donde el equipo decide que funcionalidad se va a incluir en cada iteración, y planteando un tiempo estimado para su desarrollo. Dentro de los Sprint pueden existir procesos que no sean controlados.
- Los Sprint son lineales y flexibles, donde los errores y las pruebas se usan para crear procesos de conocimiento.
- Se realizan reuniones diarias con un máximo de tiempo de 15 minutos donde se deben responder tres preguntas: ¿Qué se hizo desde la última reunión?, ¿Qué va a hacer hasta la próxima reunión diaria? y ¿Qué dificultades encontró en el desarrollo de la tarea?
- La entrega del proyecto puede ser replanificada en el transcurso de las fases, dependiendo de factores como complejidad, competencia, tiempo, dinero, entre otros.
- Está creada para ser flexible en el proceso de desarrollo, a través de mecanismos para el manejo de recursos y planificar la entrega del producto, dando lugar a que el equipo de desarrollo entregue la versión más óptima del proyecto.

2.2.6.3. Equipo Scrum

Según [20] el equipo Scrum está conformado por tres roles:

- **Product Owner.** Es quién se encarga de priorizar las tareas del backlog, además se considera la voz del cliente.
- **Scrum Master.** Es quien se asegura de que el trabajo siga las bases de Scrum, debe tener el conocimiento suficiente para explicar todo sobre la metodología al equipo.

- **Equipo de desarrollo.** Son quienes realizan las tareas de codificación y pruebas.

2.2.6.4. Sprint

Es la base de esta metodología, la cual es una iteración realizada por el equipo, del cual se obtiene un entregable, un sprint tiene un mínimo de tiempo de una semana y un máximo de cuatro.

2.2.6.5. Procesos Scrum

2.2.6.5.1. Planificación

En el proceso de planificación se realizan las actividades iniciales de un proyecto de desarrollo de software. Durante esta fase se genera la perspectiva que tiene el proyecto; Se definen requisitos para el desarrollo que se establecerá de manera distribuida, detallando las necesidades tanto para la distribución como el acuerdo con la visión en general del proyecto [21].

2.2.6.5.1.1. Planificación del Sprint

Se reúne el equipo Scrum para designar los requerimientos y las actividades a realizar durante el Sprint, el equipo determina cuánto puede lograr y basado en esto se determina la duración del Sprint.

2.2.6.5.2. Desarrollo

En esta fase se lleva a cabo la implementación de las funcionalidades con el respectivo Backlog por Sprint, el cual tiene como objetivo la coordinación e integración del equipo, esto gracias a las distintas reuniones que se realizan durante el proceso de desarrollo.

2.2.6.5.2.1. Scrum Diario.

Se realizan reuniones diarias para responder las preguntas planteadas en los principios.

2.2.6.5.2.2. Refinamiento del Backlog

Se revisan los elementos del backlog para resolver dudas por parte del equipo.

2.2.6.5.3. Revisión

Se realiza una reunión para observar el trabajo elaborado, presentando los logros alcanzados, revisando las implementaciones, soluciones y cambios, siendo aceptados o describiendo cambios agregando tareas al backlog.

2.2.6.5.4. Retrospectiva

El equipo Scrum se reúne para analizar qué se hizo mal, qué se hizo bien y que dificultades se presentaron que no permitieron continuar según lo planeado.

2.2.7. Plataforma web

Figura 5. Arquitectura de la plataforma web.



Fuente: Elaboración propia

2.2.8. JavaScript

Es un lenguaje que se basa en objetos, llegando a ser un poderoso lenguaje muy concurrido en la actualidad por los desarrolladores, ya que permite la creación de páginas, programas y aplicaciones flexibles y la adición y adaptación de miembros a los objetos [22]. JavaScript se escribe de manera dinámica, es decir, cuando se detecta un acceso en tiempo de ejecución a un usuario que no existe se provoca un error.

Es un lenguaje que no requiere compilación, es decir es un lenguaje interpretado, donde los navegadores son los encargados de hacer este trabajo. Puede ser utilizado para crear páginas, aplicaciones o multiplataformas, ya que

tiene un alto nivel de escalabilidad lo que permite a desarrolladores inexpertos empezar desde lo más básico con posibilidades de formar un software robusto.

Antes JavaScript se usaba únicamente del lado del cliente, sin embargo, en la actualidad gran cantidad de servidores están desarrollados bajo este lenguaje con implementación de Frameworks.

2.2.8.1. Node.js

Se plantea como un entorno desarrollado para el lenguaje JavaScript, el cual se usa para desarrollar aplicaciones escalables por el lado del servidor, ya que es un entorno de tiempo de ejecución [23].

La meta de Node.js es brindar una forma sencilla para que desarrolladores puedan crear programas de red escalables.

2.2.8.2. Express

Es una infraestructura web cuyo propósito es ayudar a las aplicaciones desarrolladas con Node.js, de una forma rápida, flexible y minimalista, esto a través de un conjunto de métodos HTTP [24].

2.2.8.3. Vue.js

Es un Framework de JavaScript que sirve para desarrollar interfaces de usuario de una manera sencilla y rápida, ya que posee una curva de aprendizaje baja en comparación a otros Frameworks del mismo ámbito.

2.2.9. Base de datos NoSQL

Las bases de datos NoSQL poseen estructuras de datos más flexibles que no están regidos a un esquema. Lo cual permite un mejor almacenamiento de datos no estructurados y diversos. Además, proporcionan una escalabilidad horizontal más fácil y rápida que permiten guardar conjuntos de datos con posible expansión [25].

2.2.9.1. MongoDB

MongoDB es un Sistema Gestor de Bases de Datos (SGBD) NoSQL de tipo documento, que para el almacenamiento usa documentos en formato JSON el

cual es un formato para el intercambio de datos similar a XML con una estructura simplificada [26].

2.3. OBJETIVOS DEL PROTOTIPO

2.3.1. Objetivo General

Desarrollar una plataforma web mediante estándares de la metodología SCRUM para que usuarios lleven a cabo evaluaciones subjetivas de las fases y componentes necesarios en el desarrollo de software.

2.3.2. Objetivos Específicos

- Implementar una plataforma web para evaluar el desarrollo del software.
- Evaluar mediante criterios y preguntas claves que determinan el nivel de cada fase definición, planificación, desarrollo, revisión y retrospectiva.
- Automatizar el proceso de evaluación de las fases y componentes de la metodología SCRUM para optimizar el tiempo que conlleva el proceso de evaluación.

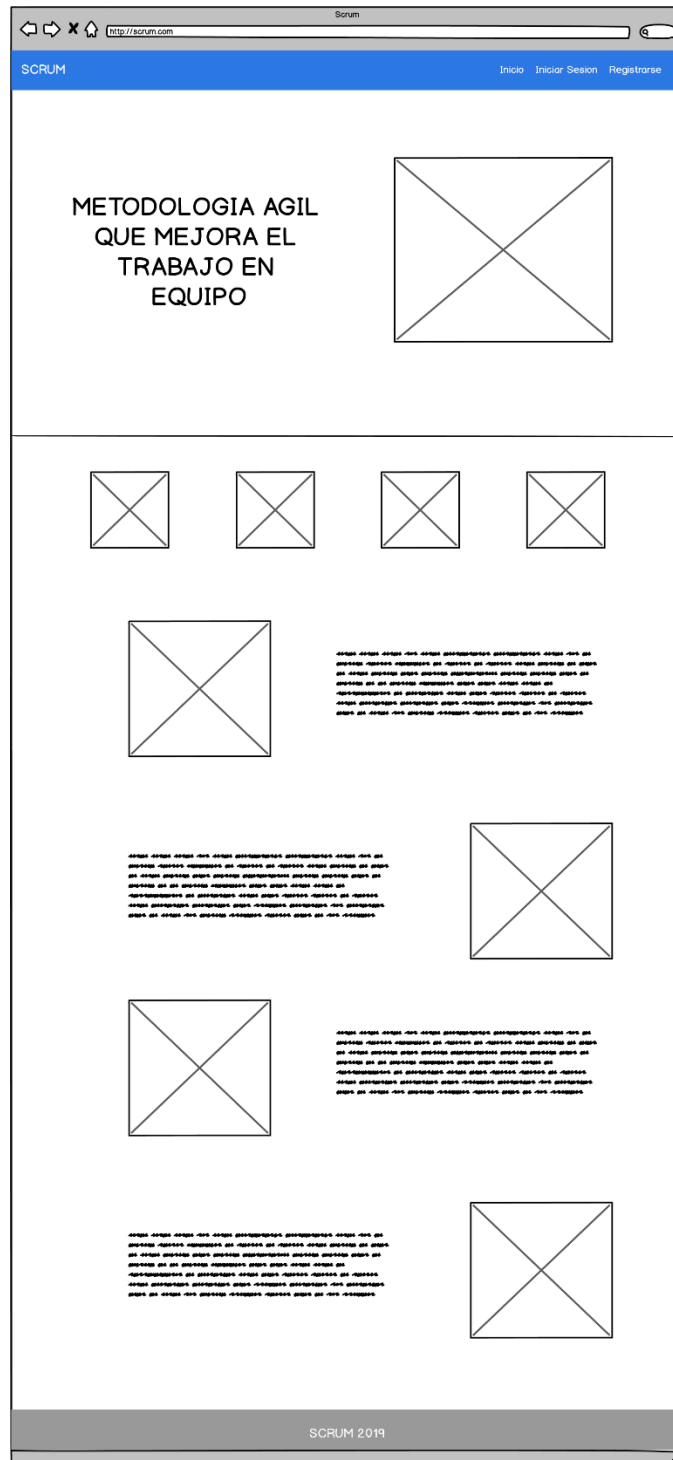
2.4. DISEÑO DEL PROTOTIPO

En la gestión de desarrollo de software se debe cumplir con el ciclo de vida basado en el análisis, diseño, codificación, pruebas y mantenimiento; para la aplicación del presente proyecto se ha implementado la metodología SCRUM por ser ágil y flexible lo que da énfasis en el desarrollo incremental, estableciendo los siguientes criterios de evaluación:

- **Análisis.** Este componente reúne las características de inicialización de un proyecto, que define parámetros de requerimientos, definición, modelado y los interesados dando enfoque y dirección al software.
- **Planeación.** Componente que está sujeto a generar valor a la organización mediante subcomponentes o tareas como historias de usuario, definición de entregables, gestión de comunicaciones, gestión de riesgos, gestión de cambios y calidad, gestión de cronograma.
- **Modelado.** Parte inherente del software que ayuda al diseño del modelo conceptual, diseño navegacional, diseño de interfaz abstracta de usuario y diseño de diagramas UML.
- **Implementación.** Gestión de las actividades, tiempo y recursos en el transcurso de la etapa, comprendiendo separación de tareas entre los miembros de trabajo, codificación de las funcionalidades y componentes, validación de las restricciones globales y generales, e implementación de métricas de usabilidad y calidad.
- **Pruebas y revisión.** Componente dedicado a la realización de pruebas de integración, detección y corrección de errores, y realización de pruebas de sistema para corregir posibles errores en el software.
- **Lanzamiento.** Finalmente, es el término del desarrollo del software generando la selección y reparación del dominio, selección y alojamiento de la aplicación de un hosting, configuración del certificado ssl y presentación de entregables.

2.4.1. Prototipo de interfaces de usuario

Figura 6. Prototipo de interfaz de usuario - Pantalla principal



Fuente: Elaboración propia

Figura 7. Prototipo de interfaz de usuario - Pantalla de inicio de sesión

SCRUM

Login

Correo

Contraseña

Detailed description: This is a wireframe of a login page for a website named 'SCRUM'. The page is displayed within a browser window with the URL 'http://scrum.com'. The main content is centered and consists of a box titled 'Login'. Inside this box, there are two input fields: one for 'Correo' (Email) and one for 'Contraseña' (Password). Below these fields is a button labeled 'Entrar' (Login).

Fuente: Elaboración propia

Figura 8. Prototipo de interfaz de usuario - Pantalla de registro de usuario

SCRUM

Registro

Nombre

Apellido

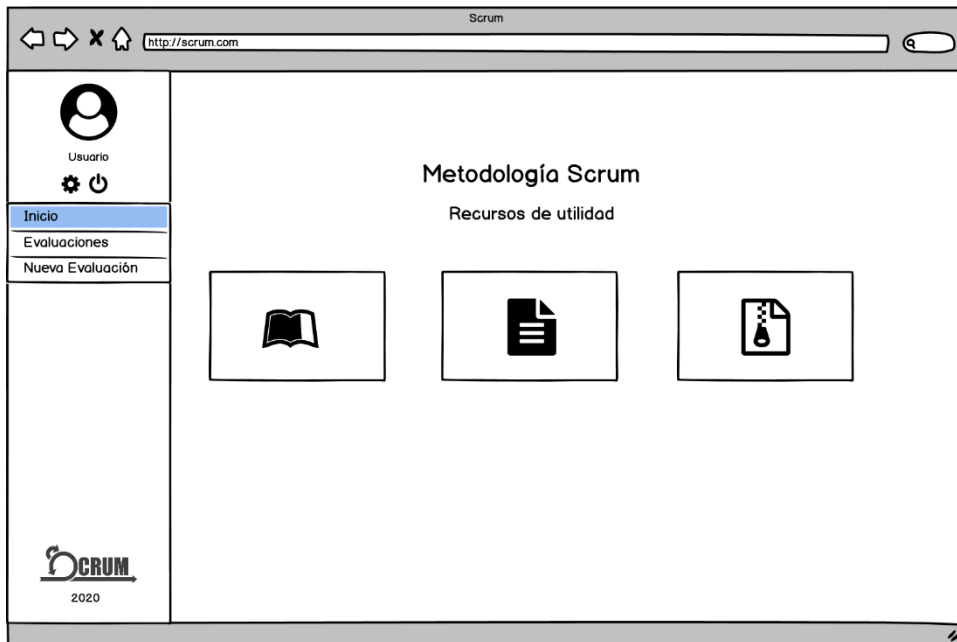
Correo

Contraseña

Detailed description: This is a wireframe of a user registration page for a website named 'SCRUM'. The page is displayed within a browser window with the URL 'http://scrum.com'. The main content is centered and consists of a box titled 'Registro'. Inside this box, there are four input fields: 'Nombre' (Name), 'Apellido' (Last Name), 'Correo' (Email), and 'Contraseña' (Password). Below these fields is a button labeled 'Registr' (Register).

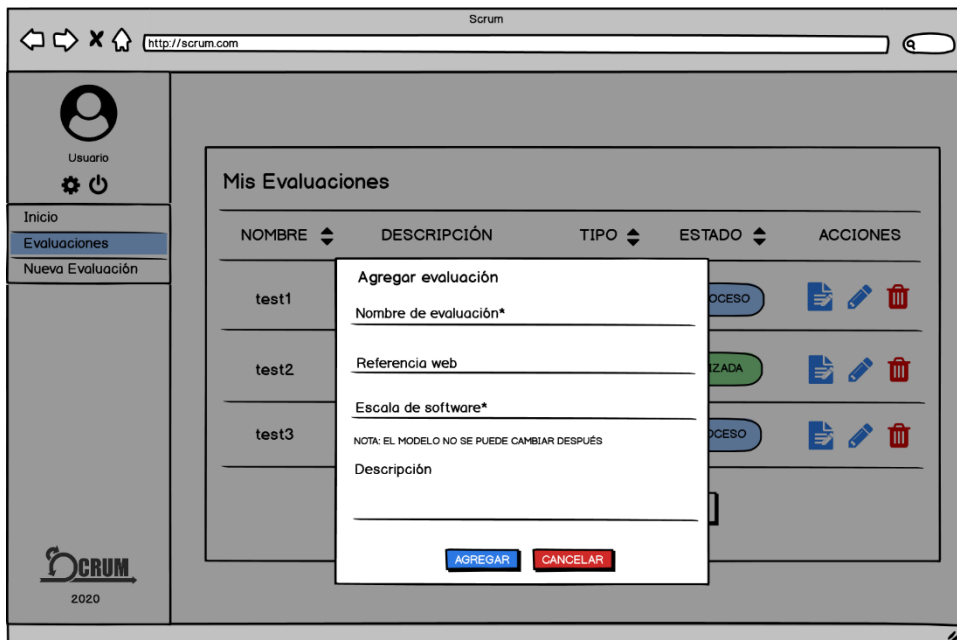
Fuente: Elaboración propia

Figura 9. Prototipo de interfaz de usuario - Pantalla de inicio de la plataforma



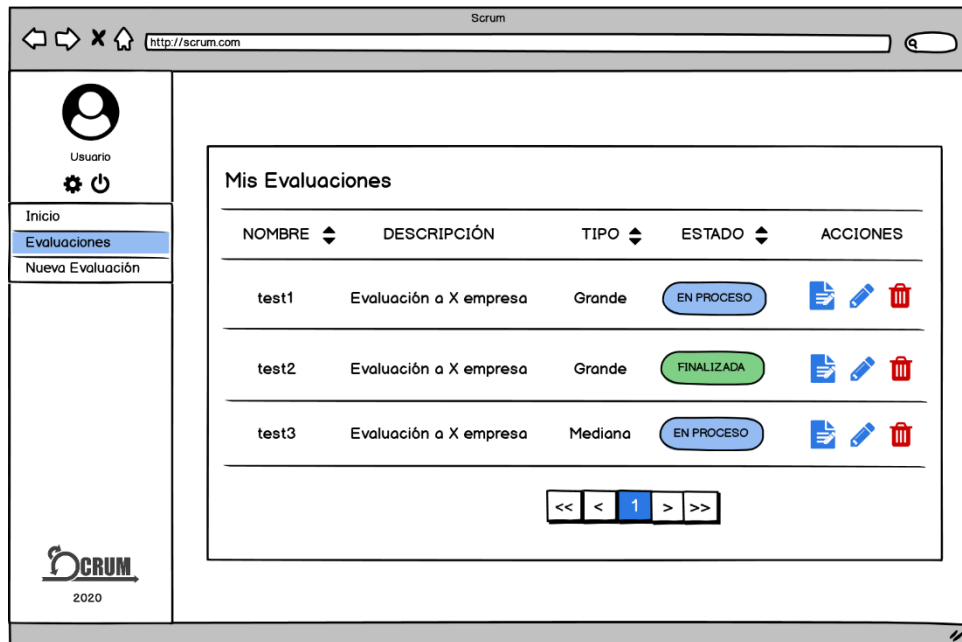
Fuente: Elaboración propia

Figura 10. Prototipo de interfaz de usuario - Formulario para agregar evaluaciones



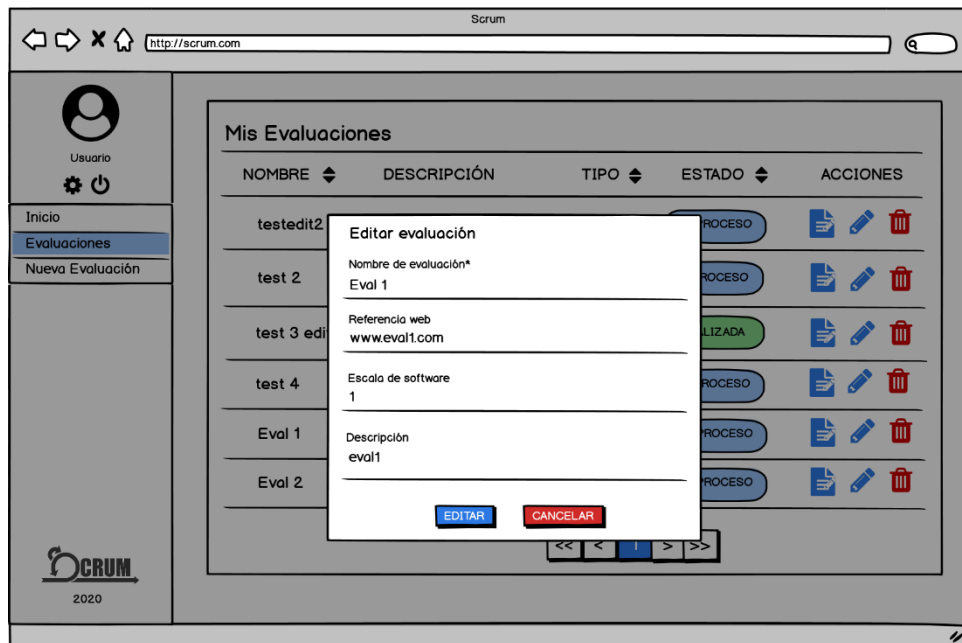
Fuente: Elaboración propia

Figura 11. Prototipo de interfaz de usuario - Pantalla de evaluaciones del usuario



Fuente: Elaboración propia

Figura 12. Prototipo de interfaz de usuario - Formulario de edición de evaluación



Fuente: Elaboración propia

Figura 13. Prototipo de interfaz de usuario - Pantalla de evaluación

Scrum

http://scrum.com

Evaluando

Nombre: Eval 1 Tipo de evaluación: Grande Referencia: www.eval1.com

Proceso: DEFINICIÓN Categoría: 1. DEFINICIÓN DE PRODUCT OWNER

Pregunta	Calificación
Es capaz de expresar claramente los elementos del backlog	Seleccione
Ordena los elementos del backlog para alcanzar misiones y objetivos	Seleccione
Asegura que el backlog del producto es visible, transparente y claro	Seleccione

CANCELAR

SCRUM 2020

Fuente: Elaboración propia

Figura 14. Prototipo de interfaz de usuario - Formulario de configuración de la cuenta

Scrum

http://scrum.com

Mis Evaluaciones

NOMBRE	DESCRIPCIÓN	TIPO	ESTADO	ACCIONES
testedit2	Configuración de la cuenta			PROCESO
test 2	Nombre* Usuario			PROCESO
test 3	Email* usuario@gmail.com			PROCESO
test 4	Género* Hombre			PROCESO
Eval 1	Nueva Contraseña			PROCESO
Eval 2				PROCESO

Configuración de la cuenta

Nombre* Usuario

Email* usuario@gmail.com

Género* Hombre

Nueva Contraseña

ACEPTAR CANCELAR

SCRUM 2020

Fuente: Elaboración propia

2.5. EJECUCIÓN Y/O ENSAMBLAJE DEL PROTOTIPO

En esta sección se describen los aspectos de la plataforma web para evaluar sistemas basados en metodología Scrum, tanto estáticos como dinámicos.

2.5.1. Pantalla principal

En la pantalla principal se muestra información acerca de la metodología Scrum, además de la barra de navegación con los botones para registro de usuarios e ingreso a la plataforma.

Figura 15. Ejecución del prototipo - Pantalla principal

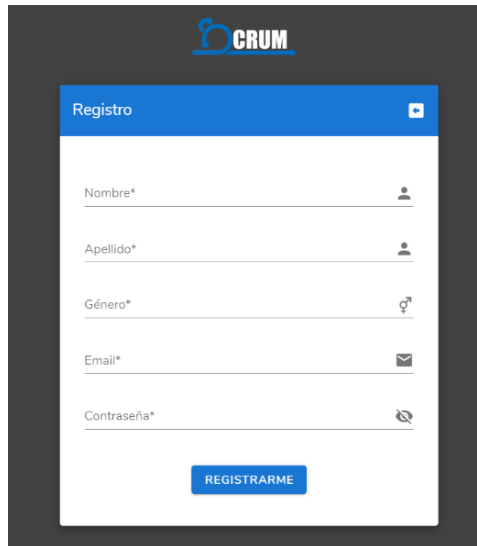


Fuente: Elaboración propia

2.5.2. Registro de usuarios

Para crear una cuenta en la plataforma se debe ingresar datos necesarios, entre ellos el correo el cual sirve para el inicio de sesión, la plataforma verifica que el correo sea único, en caso de no serlo se presentará un error al usuario.

Figura 16. Ejecución del prototipo - Pantalla de registro de usuario

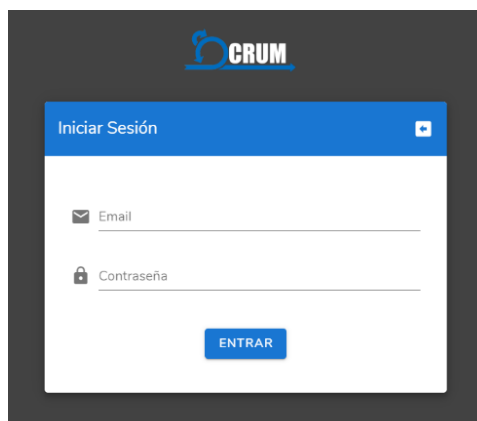


Fuente: Elaboración propia

2.5.3. Inicio de sesión de usuario

La plataforma solicita al usuario el correo y la contraseña con las que se registró, para poder iniciar la sesión, cada sesión tiene una duración de 6 horas, luego de este tiempo se volverá a solicitar que inicie su sesión.

Figura 17. Ejecución del prototipo - Pantalla de inicio de sesión

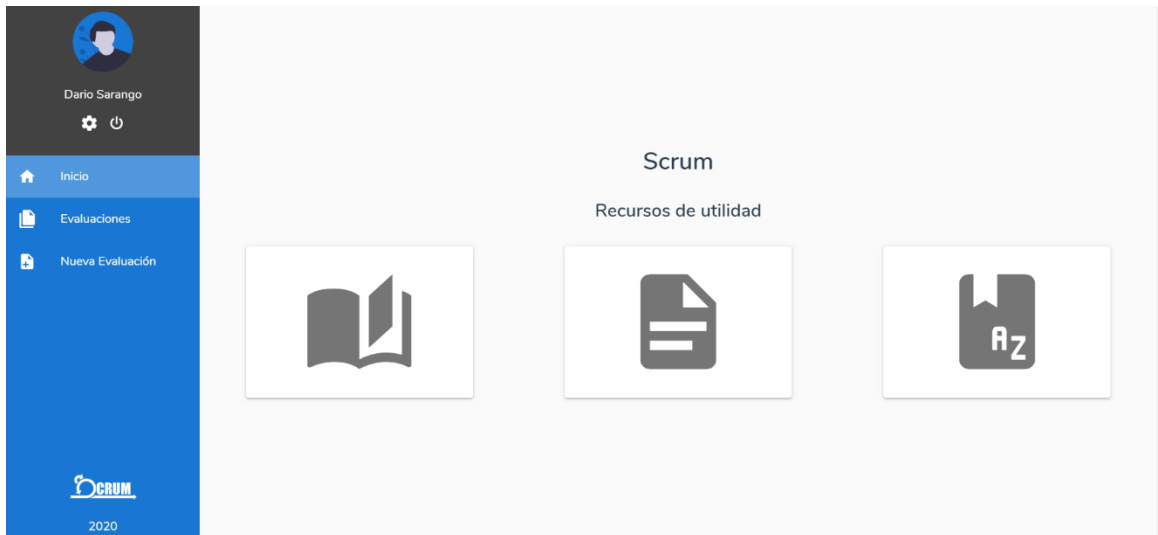


Fuente: Elaboración propia

2.5.4. Pantalla de inicio

Al iniciar la sesión se presenta una pantalla con recursos de utilidad para el usuario, entre ellos la guía scrum oficial, un artículo en español que explica detalladamente la metodología, y un diccionario en el que se explican conceptos determinados.

Figura 18. Ejecución del prototipo - Pantalla de inicio



Fuente: Elaboración propia

2.5.5. Configuración de la cuenta

La plataforma permite al usuario editar sus datos a través de un formulario de configuración.

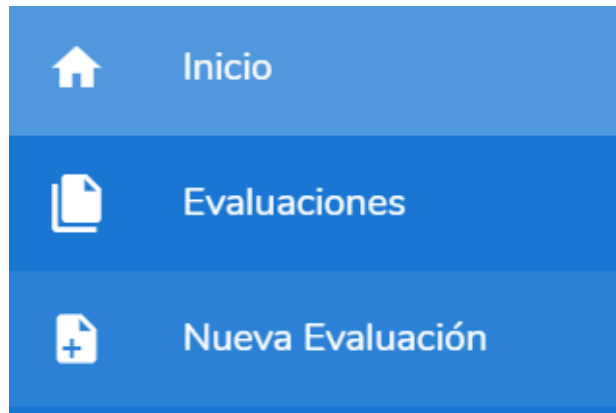
Figura 19. Ejecución del prototipo - Formulario de configuración de la cuenta

Fuente: Elaboración propia

2.5.6. Creación de evaluaciones

Para agregar una evaluación el usuario debe hacer click sobre Nueva Evaluación en el menú lateral, para abrir el formulario de creación.

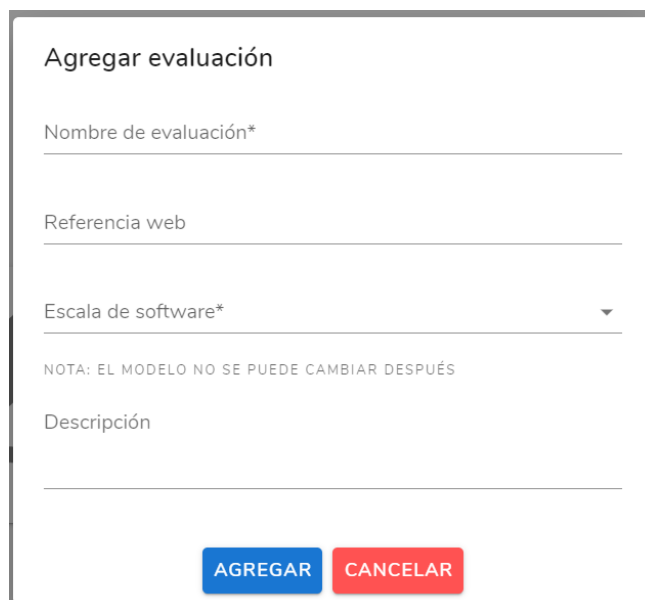
Figura 20. Ejecución del prototipo - Menú del usuario



Fuente: Elaboración propia

El formulario solicita datos requeridos entre ellos la escala del software, el cual determinará el tipo de evaluación que se realizará.

Figura 21. Ejecución del prototipo - Formulario para agregar evaluación

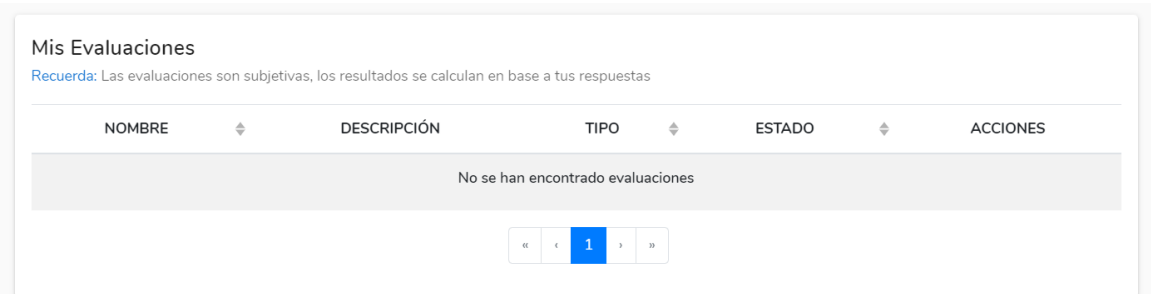
Un formulario con el título 'Agregar evaluación'. Tiene los siguientes campos: 'Nombre de evaluación*' (campo de texto), 'Referencia web' (campo de texto), 'Escala de software*' (menú desplegable), una nota que dice 'NOTA: EL MODELO NO SE PUEDE CAMBIAR DESPUÉS', y 'Descripción' (campo de texto). En la parte inferior hay dos botones: 'AGREGAR' (azul) y 'CANCELAR' (rojo).

Fuente: Elaboración propia

2.5.7. Evaluaciones del usuario

Mediante una tabla se presentan las evaluaciones del usuario, si es una cuenta nueva se muestra un mensaje indicando que no se han agregado evaluaciones.

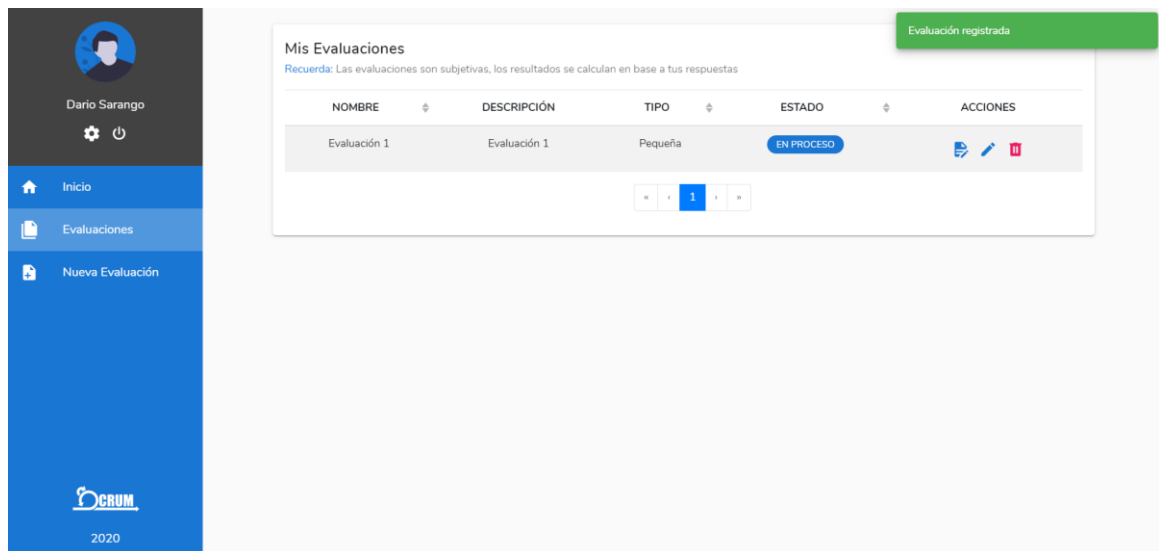
Figura 22. Ejecución del prototipo - Pantalla de evaluaciones



Fuente: Elaboración propia

Si el usuario ha creado evaluaciones se presentará una alerta de confirmación y se actualizará la tabla de evaluaciones.

Figura 23. Ejecución del prototipo - Listado de evaluaciones



Fuente: Elaboración propia

Por cada evaluación se presenta sus datos, el estado en que se encuentra ya sea en proceso o finalizada, además se incluyen tres botones para editar los datos de la evaluación, para eliminarla y dependiendo del estado en que se encuentre será el botón para evaluar o para revisar la evaluación.

Figura 24. Ejecución del prototipo - Vista de evaluación en proceso



NOMBRE	DESCRIPCIÓN	TIPO	ESTADO	ACCIONES
Evaluación 1	Evaluación 1	Pequeña	EN PROCESO	  

Figura 25. Ejecución del prototipo - Vista de evaluación finalizada

NOMBRE	DESCRIPCIÓN	TIPO	ESTADO	ACCIONES
Evaluación 1	Evaluación 1	Pequeña	FINALIZADA	  

Fuente: Elaboración propia

2.5.8. Edición de evaluación

Al ingresar a la edición de la evaluación se presenta un formulario que permite cambiar los datos, exceptuando la escala, ya que como se indica en la **Figura 21** cuando se agrega una evaluación no se puede cambiar.

Figura 26. Ejecución del prototipo - Formulario de edición de evaluación

Editar evaluación

Nombre de evaluación*
Evaluación 1

Referencia web
www.evaluacion.com

Escala de software
3

Descripción
Evaluación 1

EDITAR
CANCELAR

Fuente: Elaboración propia

2.5.9. Pantalla de evaluación

Esta es la sección principal de la plataforma, primero se presentan los datos de la evaluación que estamos realizando, luego dos seleccionadores, el primero muestra los procesos de la metodología, y el segundo las categorías del proceso que escogemos, una tabla presenta la preguntas a responder por cada categoría, y al final se incluyen botones de utilidad para el usuario.

Figura 27. Ejecución del prototipo - Pantalla de evaluación

The screenshot shows a web interface for an evaluation. At the top, there are three blue buttons: 'Nombre: Evaluación 1', 'Tipo de evaluación: Pequeña', and 'Referencia: www.evaluacion.com'. Below these are two dropdown menus: 'Proceso DEFINICIÓN' and 'Categoría 1. DEFINICIÓN DE PRODUCT OWNER'. The main content is a table with two columns: 'Pregunta' and 'Calificación'. The table contains three rows of questions, each with a corresponding dropdown menu for the rating.

Pregunta	Calificación
Es capaz de expresar claramente los elementos del backlog	Seleccione
Ordena los elementos del backlog para alcanzar misiones y objetivos	Seleccione
Asegura que el backlog del producto es visible, transparente y claro	Seleccione

Fuente: Elaboración propia

Por cada pregunta se debe proporcionar una calificación, escogiendo una opción del seleccionador con cinco opciones, basadas en la escala de Likert.

Figura 28. Ejecución del prototipo - Opciones de calificación

The screenshot shows a vertical list of five rating options for a Likert scale, each in a light gray box:

- Totalmente en desacuerdo (20)
- En desacuerdo (40)
- Ni de acuerdo, ni en desacuerdo (60)
- De acuerdo (80)
- Totalmente de acuerdo (100)

Fuente: Elaboración propia

Se presentan dos botones para navegar a través de los procesos y categorías para facilitar la evaluación al usuario.

Figura 29. Ejecución del prototipo – Controles paginación de evaluación



Fuente: Elaboración propia

Cuando la evaluación está en proceso se presentan los botones de guardar que permite guardar el avance realizado sin finalizar la evaluación, y cancelar que cierra la ventana de evaluación sin guardar el progreso.

Figura 30. Ejecución del prototipo - Controles de evaluación en proceso

GUARDAR CANCELAR

Fuente: Elaboración propia

Cuando se han respondido todas las preguntas de cada proceso, se presenta el botón de finalizar, que cierra el proceso de evaluación y permite ver los resultados de la evaluación.

Figura 31. Ejecución del prototipo - Controles de evaluación finalizada

FINALIZAR GUARDAR CANCELAR

Fuente: Elaboración propia

Al finalizar la evaluación se muestra una alerta de finalización, se actualiza el estado y se cambia el botón de evaluar a ver.

Figura 32. Ejecución del prototipo - Visualización de evaluaciones finalizadas

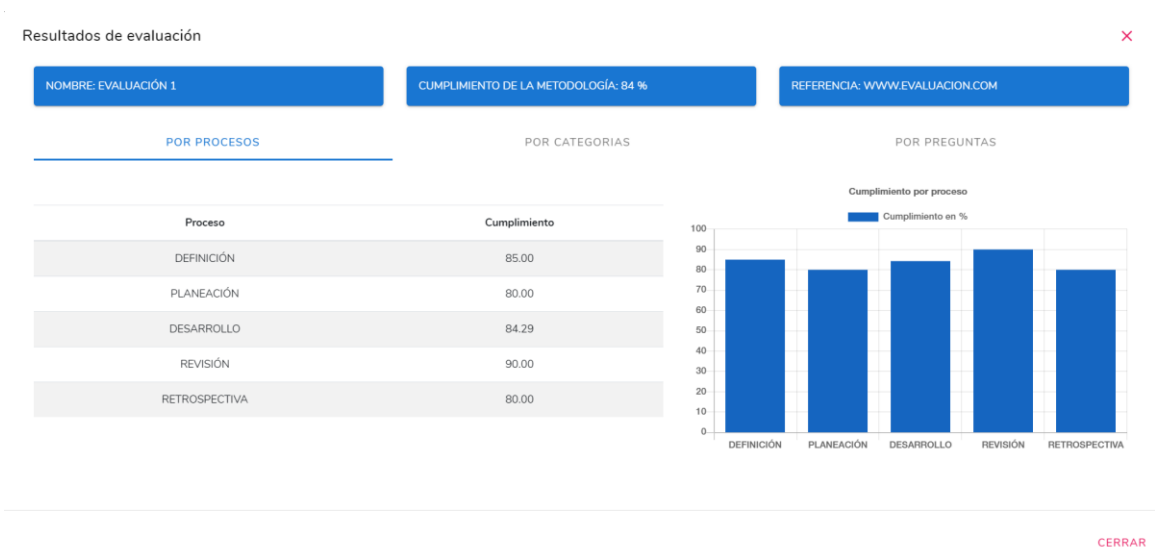


Fuente: Elaboración propia

2.5.10. Resultados de evaluación

En esta sección se presenta la evaluación finalizada, a través de gráficas generadas a partir de las respuestas del usuario, donde puede elegir distintas formas de ver las puntuaciones y un resultado cuantitativo final de la evaluación.

Figura 33. Ejecución del prototipo - Pantalla de resultados de evaluación



Fuente: Elaboración propia

Se presenta información de la evaluación y el porcentaje de cumplimiento de la metodología, calculado de manera general en base a las respuestas del usuario.

Figura 34. Ejecución del prototipo - Información de evaluación



Fuente: Elaboración propia

Figura 35. Ejecución del prototipo - Calificación del cumplimiento de la metodología

CUMPLIMIENTO DE LA METODOLOGÍA: 84 %

Fuente: Elaboración propia

El usuario puede escoger entre tres opciones de qué forma quiere ver la tabla y el gráfico, por defecto se presenta por procesos.

Figura 36. Ejecución del prototipo - Controles de visualización de resultados



Fuente: Elaboración propia

La tabla indica el cumplimiento de cada proceso, categoría o pregunta, según se escoja.

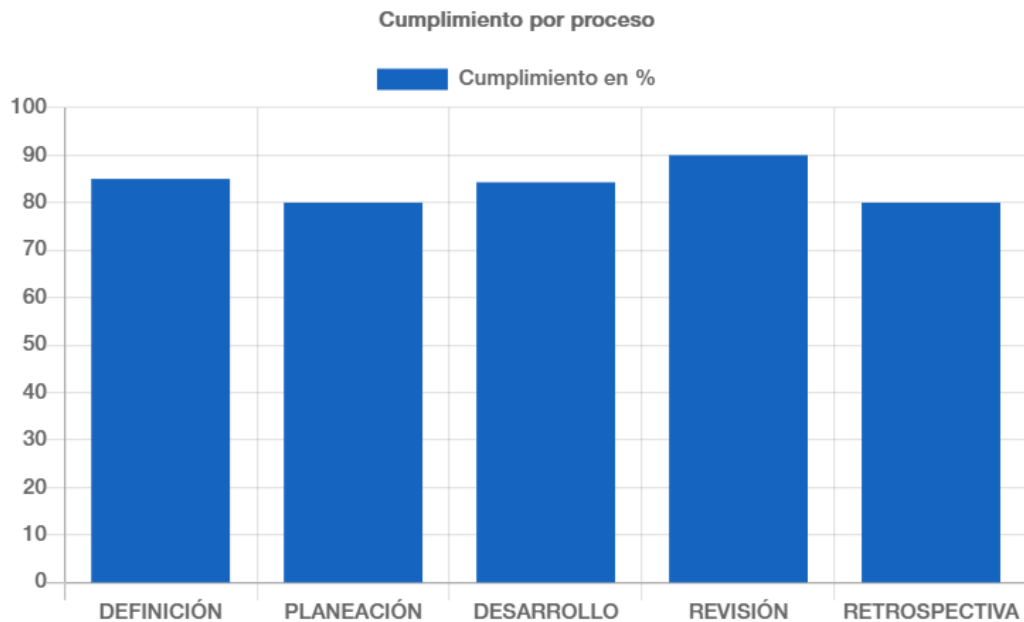
Figura 37. Ejecución del prototipo - Tabla de resultados

Proceso	Cumplimiento
DEFINICIÓN	85.00
PLANEACIÓN	80.00
DESARROLLO	84.29
REVISIÓN	90.00
RETROSPECTIVA	80.00

Fuente: Elaboración propia

El gráfico ayuda a los usuarios a una mejor visión de las fallas y aciertos de la metodología, presentando el nivel de cumplimiento basado en la selección para visualizar.

Figura 38. Ejecución del prototipo - Gráfico de resultados



Fuente: Elaboración propia

3. CAPÍTULO III. EVALUACIÓN DEL PROTOTIPO

3.1. PLAN DE EVALUACIÓN

La evaluación de software tiene gran relevancia en el proceso de desarrollo de software, ya que suponen el éxito del proyecto realizado, la capacidad de percibir errores, y generar posibles cambios.

3.1.1. Evaluación de calidad

La calidad de un sitio web puede ser evaluada mediante estándares, [27] propone ocho herramientas online para la evaluación de la calidad basándose en la norma ISO/IEC 9126, considerando los parámetros: accesibilidad, funcionalidad, eficiencia, usabilidad, portabilidad y posicionamiento, siendo estas herramientas: Validator HTML, Validator (HTML y CSS juntos), Functional accessibility evaluator 2.0, GtMetrix, Moz Bar, Metric Spot, Website grader y ReadyMobi. Las pruebas realizadas se pueden ver desde los **Anexo 1 - Anexo 8**.

3.2. RESULTADOS DE LA EVALUACIÓN

Las evaluaciones realizadas en distintas herramientas web online recomendadas fueron de gran ayuda para determinar el nivel de cumplimiento de procesos basados en los requerimientos esperados.

3.2.1. Validator HTML

Al utilizar esta herramienta (**Anexo 1**), para la evaluación del sitio web se obtuvo los resultados esperados al no encontrar algún tipo de error y/o advertencias.

3.2.2. Validator CSS

Esta herramienta analiza los archivos CSS que puede haber dentro del sistema (**Anexo 2**), se observa que los resultados son aceptables, porque existe 36 errores y 4059 advertencias, estos errores sintácticos se deben a que los estilos y extensiones desconocidas son provenientes de librerías de terceros.

3.2.3. Funcional Accessibility Evaluator 2.0

Esta herramienta permite la validación mediante URL respecto al nivel A y AA a su vez permite seleccionar HTML5 y técnicas ARIA o Técnicas HTML4.

La evaluación de accesibilidad del sitio web mediante el uso de esta herramienta donde los resultados se muestran por el alcance de la regla con un valor de 0 a 100 (**Anexo 3**). El puntaje que se obtuvo es de 33, siendo este un puntaje aceptable al aprobar 10 reglas en contra de 4.

3.2.4. GtMetrix

La página principal de la herramienta que se utilizó proporciona información organizada siendo intuitiva para el evaluador que la utiliza.

Al utilizar esta herramienta (¡Error! No se encuentra el origen de la referencia.), da como resultado los scores de PageSpeed con una calificación "F" (38%) en base a las reglas más importantes consideradas por Google y el de Yslow con una calificación "D" (65%) según las reglas consideradas por Yahoo!.

La herramienta también proporciona parámetros como; Fully Loaded Time (tiempo de carga completo) obteniendo un valor de 2.8 S, Total Page Size (tamaño total de la página) obteniendo un valor de 2.84 MB y Requests (peticiones) obtiene un valor de 20.

3.2.5. Moz Bar

Esta herramienta proporciona datos sobre el rendimiento del sistema web en los principales motores de búsqueda mediante técnicas de SEO (

Anexo 5).

Los resultados son bajos debido a que el sitio aún no se encuentra en producción.

3.2.6. Metricspot

Esta herramienta (**Anexo 6**) realiza la evaluación de análisis web y auditoría SEO la cual proporcionará datos para la mejora del sitio web.

Los datos que se obtuvo de la evaluación es una puntuación general de presencia online de 23.8%, indicando que existe mucho margen de mejora.

3.2.7. Websitte grader

Al utilizar esta herramienta para la evaluación del sitio web, se obtuvo como resultado (

Anexo 7), un alto rendimiento de 28/30, en el segundo parámetro refleja que es totalmente portable obteniendo un puntaje de 30/30.

3.2.8. ReadyMobi

Esta es una herramienta que permite comprobar si el sitio web cumple con la portabilidad que se requiere para el acceso del sistema (**Anexo 8**) mediante un conjunto de test específicos de 38 pruebas dedicadas al uso en dispositivos móviles.

Los resultados demuestran que el sistema es totalmente funcional en cualquier dispositivo con 33 pruebas pasadas, 3 fallas mayores y 2 fallas menores.

3.3. CONCLUSIONES

- Como resultado del presente trabajo, es posible concluir que, para llevar a cabo un desarrollo de software, se requiere de aspectos como comunicación, coordinación y un equipo de trabajo comprometido con el proyecto, para ello el uso de metodologías permiten a grupos de trabajo generar un mejor ambiente en el cual sentirse cómodo y asegurando una mayor probabilidad de éxito del proyecto.
- Mediante la plataforma para la evaluación de software basado en metodología Scrum, se podrá obtener mejores resultados, al poder visualizar los puntos más fuertes y débiles del equipo y el proyecto realizado, para así no cometer los mismos fallos en futuros proyectos.
- La plataforma da la posibilidad de ver a través de tablas y gráficos, el cumplimiento de cada proceso y categoría de la metodología, todo esto según lo requiera el usuario, además, se genera un dato cuantitativo final para tener una mejor visión del cumplimiento general.
- Evaluar un software no solo ayuda a mejorarlo, ayuda a los desarrolladores a ver los fallos cometidos en un proyecto, hacer uso de herramientas de evaluación permiten que este proceso sea mucho más sencillo, ahorrando tiempo y generando un estándar para los grupos de desarrollo.
- Actualmente existen Frameworks y herramientas para realizar interfaces de usuario de una forma más ordenada y rápida, tal es el caso de Balsamiq Mockups que es una herramienta que nos permite generar prototipos de manera rápida y Vue.js que da la posibilidad de generar interfaces en cortos tiempos de desarrollo.

3.4. RECOMENDACIONES

- En los trabajos de desarrollo de software es importante hacer uso de una metodología que permita al grupo de trabajo una buena comunicación, coordinación y compromiso para garantizar el éxito del producto, por ende, es recomendable aprovechar cada uno de los parámetros de la metodología empleada.
- En todo proyecto basado en la metodología Scrum la evaluación es parte fundamental del desarrollo ya que requiere de tiempo y dedicación, por lo cual se recomienda hacer uso del software para evaluar su cumplimiento, que ayuda a visualizar los puntos críticos y fuertes, permitiendo realizar mejoras del proyecto.
- El software cuenta con criterios y preguntas claves que determinan el nivel de cada fase por lo que el usuario final debe enfocarse y realizar la evaluación con atención, dependiendo de los datos que ingrese al sistema, se visualizan los resultados mediante tablas y gráficos, adicional mostrará un dato cuantitativo de toda la evaluación.
- Se debe conocer la metodología Scrum para implementar de manera correcta la automatización de los procesos que requieren las fases y sus componentes, de esta manera se podrá desarrollar un software que ayude a los desarrolladores a encontrar fallos del proyecto en menos tiempo y más fácil.
- Hoy en día se debe hacer uso de herramientas como Balsamiq Mockups que ayudan a crear prototipos de las interfaces de usuario y Vue.js que genera las interfaces en corto tiempo, es por ello que se implementa Frameworks y herramientas que permiten desarrollar un software de manera ordenada.

BIBLIOGRAFÍA

- [1] F. SÁENZ BLANCO, F. GUTIÉRREZ SIERRA, and J. C. RAMOS RIVERA, "CONFORMACIÓN DE EQUIPOS ÁGILES PARA EL DESARROLLO DE SOFTWARE: REVISIÓN DE LITERATURA," *Dimens. Empres.*, vol. 16, no. 2, pp. 39–54, 2018, doi: 10.15665/rde.v16i2.1042.
- [2] L. F. Muñoz Sanabria, J. A. Hurtado Alegría, and F. J. Álvarez Rodríguez, "Agile Architecture in Action (AGATA)," *Ing. y Univ.*, vol. 22, no. 1, pp. 33–51, Jan. 2018, doi: 10.11144/Javeriana.iyu22-1.aaaa.
- [3] L. E. V. A. Eliécer Herrera Uribe, "Del manifiesto ágil sus valores y principios. - Dialnet," 2007. [Online]. Available: <https://dialnet.unirioja.es/servlet/articulo?codigo=4809645>. [Accessed: 25-Feb-2020].
- [4] J. M. Ríos and N. Pedreira-Souto, "Approach of Agile Methodologies in the Development of Web-Based Software," *Information*, vol. 10, no. 10, p. 314, Oct. 2019, doi: 10.3390/info10100314.
- [5] G. Sobrevilla, J. Hernández, P. Velasco, and S. Soriano, "Aplicando Scrum y Prácticas de Ingeniería de Software para la Mejora Continua del Desarrollo de un Sistema Ciber-Físico," 2017.
- [6] J. R. Molina Ríos, M. P. Zea Ordóñez, M. J. Contenido Segarra, and F. G. García Zerda, "Comparación De Metodologías En Aplicaciones Web," *3C Technol. innovación Apl. a la pyme*, vol. 7, no. 1, pp. 1–19, 2018, doi: 10.17993/3ctecno.2018.v7n1e25.1-19.
- [7] L. M. Arana López, M. E. Ruiz Rivera, and N. La Serna Palomino, "Análisis de aplicaciones empleando la computación en la nube de tipo PaaS y la metodología ágil Scrum," *Ind. Data*, vol. 18, no. 1, p. 149, 2016, doi: 10.15381/idata.v18i1.12077.
- [8] R. DE Yunga Dickson Joel Ingeniero Sistemas, "Implementación de un servicio ecommerce utilizando la metodología xp enfocado a la gastronomía de Machala," Machala: Universidad Técnica de Machala, 2019.

- [9] F. E. Ríos Pérez, F. P. Carrillo, and V. Moreno Vega, "Servidor web empotrado en un FPGA para configurar un Controlador Maestro del Sistema Inteligente de Tráfico Cubano FPGA," *Rev. Cuba. Ciencias Informáticas*, vol. 11, no. 2, 2017.
- [10] L. VaLentina Gómez Fermín and T. raFaeL moreno PoGGio, "PROPUESTA DE MODELO EN CINCO CAPAS PARA APLICACIONES WEB," vol. 26, pp. 168–173, 2014.
- [11] Y. F. Li, P. K. Das, and D. L. Dowe, "Two decades of Web application testing - A survey of recent advances," *Information Systems*, vol. 43. Elsevier Ltd, pp. 20–54, 01-Jul-2014, doi: 10.1016/j.is.2014.02.001.
- [12] B. Abundis and C. Joaquín, "Metodologías para desarrollar software seguro," *ReCIBE*, vol. 2, no. 3, p. V, 2013.
- [13] A. N. Cadavid, "Revisión de metodologías ágiles para el desarrollo de software," *Prospectiva*, vol. 11, no. 2, p. 30, 2013, doi: 10.15665/rp.v11i2.36.
- [14] D. Tore and D. Torgeir, "Empirical studies of agile software development," *Inf. Softw. Technol.*, pp. 833–859, Sep. 2008.
- [15] J. Molina-Ríos and N. Pedreira-Souto, "Comparison of development methodologies in web applications," *Inf. Softw. Technol.*, vol. 119, p. 106238, Mar. 2019, doi: 10.1016/j.infsof.2019.106238.
- [16] R. Tinoco Gómez, López, P. Pablo, and U. N. M. de S. M. P. Bacalla, Salas, "Criterios de selección de metodologías de desarrollo de software," *Ind. Data*, vol. 13, no. 2, pp. 70–74, 2010.
- [17] L. O. Caballero Chi, "Factors Models of Scrum Adoption in the Software Development Process," p. 23, 2018.
- [18] S. I and P. L, "Implementación de SCRUM en el diseño del proyecto del Trabajo Final de Aplicación," 2014.
- [19] A. O. Duarte and M. Rojas, "Las Metodologías de Desarrollo Ágil como una Oportunidad para la Ingeniería del Software Educativo," 2008.
- [20] W. G. Barrios, M. G. Fernández, and F. M. Ferreira, "SCRUM: Application

- Experience in a Software Development PyME in the NEA.”
- [21] C. J. Pardo-Calvache, P. R. Chilito-Gómez, D. E. Viveros-Meneses, and F. J. Pino, “Scrum+: A scaled Scrum for the agile global software development project management with multiple models,” *Rev. Fac. Ing.*, no. 93, pp. 105–116, 2019, doi: 10.17533//udea.redin.20190519.
- [22] C. Anderson and P. Giannini, “Type checking for JavaScript,” *Electron. Notes Theor. Comput. Sci.*, vol. 138, no. 2, pp. 37–58, Nov. 2005, doi: 10.1016/j.entcs.2005.09.010.
- [23] L. You *et al.*, “JDap: Supporting in-memory data persistence in javascript using Intel’s PMDK,” *J. Syst. Archit.*, vol. 101, p. 101662, Dec. 2019, doi: 10.1016/j.sysarc.2019.101662.
- [24] D. E. A. Isom and R. Con, “Implementación de aplicaciones isomórficas con JavaScript,” pp. 143–161, 2015.
- [25] H. W. K. and O. M. S., “NoSQL databases: forensic attribution implications,” *SAIEE Africa Res. J.*, vol. 109, no. 2, pp. 119–132, 2006.
- [26] F. J. Moreno Arboleda, J. E. Quintero Rendón, and R. Rueda Vásquez, “Una comparación de rendimiento entre Oracle y MongoDB,” *Cienc. e Ing. Neogranadina*, vol. 26, no. 1, p. 109, Apr. 2016, doi: 10.18359/rcin.1669.
- [27] E. Serna and F. Arango, “Prueba del software: más que una fase en el ciclo de vida,” 2012.
- [28] W3C, “Validator,” 2020. [Online]. Available: <https://validator.w3.org/nu/>.
- [29] W3C, “Servicio de validación de CSS,” 2009. [Online]. Available: <https://jigsaw.w3.org/css-validator/about.html>.
- [30] University of Illinois, “Functional Accessibility Evaluator 2.1,” 2014. [Online]. Available: [https://fae.disability.illinois.edu/anonymous/?Anonymous Report=/
/](https://fae.disability.illinois.edu/anonymous/?Anonymous%20Report=/).
- [31] GT.net, “GTmetrix,” 2020. [Online]. Available: <https://gtmetrix.com>.
- [32] Moz, “Moz Bar,” 2020. [Online]. Available: <https://moz.com/>.
- [33] Metricspot, “Herramientas de Análisis Web y Auditoría SEO,” 2020.

[34] HubSpot, "Website Grader," 2020. [Online]. Available: <https://website.grader.com/>.

[35] A. T. Ltd, "Mobi Ready," 2016. [Online]. Available: <https://ready.mobi/>.

ANEXOS

Anexo 1: Evaluación de calidad - herramienta Validator html

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for <https://scrumevaluator.herokuapp.com/>

Checker Input

Show source outline image report

Check by address

<https://scrumevaluator.herokuapp.com/>

Document checking completed. No errors or warnings to show.

Used the HTML parser. Externally specified character encoding was UTF-8.
Total execution time 80 milliseconds.

[About this checker](#) • [Report an issue](#) • Version: 20.2.25

Anexo 2: Evaluación de calidad - herramienta Validator (html y css)

W3C El Servicio de Validación de CSS del W3C
Resultados del Validador CSS del W3C para <https://scrumevaluator.herokuapp.com/> (CSS versión 3 + SVG)

Ira: **Los Errores (36)** Las Advertencias (4059) Su Hoja de Estilo validada

Resultados del Validador CSS del W3C para <https://scrumevaluator.herokuapp.com/> (CSS versión 3 + SVG)

Disculpas! Hemos encontrado las siguientes errores (36)

URL: <https://scrumevaluator.herokuapp.com/css/chunk-vendors.5c259f7e.css>

7	[type="button"]::-moz-focus-inner, [type="reset"]::-moz-focus-inner, [type="submit"]::-moz-focus-inner, button::-moz-focus-inner	Propiedad no válida : border-style Faltan valores o no se reconocen los valores : 0
7	.v-btn--block	Propiedad no válida : max-width Faltan valores o no se reconocen los valores : auto
7	.v-chip--pill .v-chip__filter	Propiedad no válida : margin-right Faltan valores o no se reconocen los valores : 0 16px 0 0
7	.theme--light.v-calendar-daily .v-calendar-daily__intervals-body .v-calendar-daily__interval-text	Propiedad no válida : color Faltan valores o no se reconocen los valores : #424242 1px solid
7	.theme--dark.v-calendar-daily .v-calendar-daily__intervals-body .v-calendar-daily__interval-text	Propiedad no válida : color Faltan valores o no se reconocen los valores : #eee 1px solid

Anexo 3: Evaluación de calidad - herramienta Functional Accessibility Evaluator 2.1

Run FAE

Page Title: **Scrum** February 25, 2020 13:03

URL: <https://scrumevaluator.herokuapp.com/> Ruleset: [HTML5 and ARIA Techniques](#)

Reports

Page Summary [E-mail Report](#)


Page 1: [Scrum](#)


	Violations	Warnings	Manual Checks	Passed
Number of Rules	4	-	23	10

Rule Category [WCAG Guidelines](#) Rule Scope

Rule Group	Number of Rules				Score	Status
	V	W	MC	P		
Website	-	-	5	-	0	R
Page	-	-	10	3	23	NI-R
Element	4	-	8	7	49	NI-R
All Rule Groups	4	-	23	10	33	na

Anexo 4: Evaluación de calidad - herramienta GTmetrix



Features Resources Blog GTmetrix PRO





Latest Performance Report for:

<https://scrumevaluator.herokuapp.com/>

Report generated: Tue, Feb 25, 2020 11:05 AM -0800

Test Server Region:  Vancouver, Canada

Using:  Chrome (Desktop) 75.0.3770.100, PageSpeed 1.15-gt1.2, YSlow 3.1.8

 Looks like you might not be using a CDN
[Why should I use a CDN?](#)

Performance Scores

PageSpeed Score

F (38%)

YSlow Score

D (65%)

Page Details

Fully Loaded Time

2.8s

Total Page Size

2.84MB

Requests

20

PageSpeed
YSlow
Waterfall
Timings
Video
History

RECOMMENDATION	GRADE	TYPE	PRIORITY
Enable compression	F (0)	SERVER	HIGH
Minify JavaScript	F (9)	JS	HIGH
Optimize images	A (99)	IMAGES	HIGH
Minify CSS	A (99)	CSS	HIGH
Avoid bad requests	A (100)	CONTENT	HIGH
Avoid landing page redirects	A (100)	SERVER	HIGH
Defer parsing of JavaScript	A (100)	JS	HIGH
Enable Keep-Alive	A (100)	SERVER	HIGH
Inline small CSS	A (100)	CSS	HIGH
Inline small JavaScript	A (100)	JS	HIGH
Leverage browser caching	A (100)	SERVER	HIGH
Minimize redirects	A (100)	CONTENT	HIGH
Minimize request size	A (100)	CONTENT	HIGH
Optimize the order of styles and scripts	A (100)	CSS:JS	HIGH
Put CSS in the document head	A (100)	CSS	HIGH
Serve resources from a consistent URL	A (100)	CONTENT	HIGH
Serve scaled images	A (100)	IMAGES	HIGH
Specify a cache validator	A (100)	SERVER	HIGH
Combine images using CSS sprites	A (100)	IMAGES	HIGH
Avoid CSS @import	A (100)	CSS	MEDIUM
Prefer asynchronous resources	A (100)	JS	MEDIUM
Specify a character set early	A (100)	CONTENT	MEDIUM
Specify image dimensions	A (100)	IMAGES	MEDIUM
Avoid a character set in the meta tag	A (100)	CONTENT	LOW
Minify HTML	A (100)	CONTENT	LOW
Specify a Vary: Accept-Encoding header	A (100)	SERVER	LOW

What do my scores mean?

Rules are sorted in order of impact upon score
 Optimizing rules at the top of the list can greatly improve your overall score.

Not every recommendation will apply to your page
 The recommendations are meant to be generic, best practices; some things will be out of your control (eg. external resources) or may not apply to your page.


[Learn more about PageSpeed/YSlow scores and how they affect performance.](#)

Need optimization help?

Read our [how to guides](#) and [optimization explained articles](#) for additional direction on improving your page performance.


Want an expert to help you with optimization? We can recommend partners to further assist you with optimizing your site. [Contact us](#) and we'll point you in the right direction.

Anexo 5: Evaluación de calidad - herramienta Moz Bar

MOZ  Get Link Metrics with a FREE Moz Community Account [Create Account](#)


On-Page Elements General Attributes Link Metrics Markup Http Status

Tag/Location	Content	# of Ch
URL	 https://scrumevaluator.herokuapp.com/	3
Page Title	Scrum	5
Meta Description	Not found	--
Meta Keywords	Not found	--
H1	MEJORA TUS PRÁCTICAS DE DIRECCIÓN Y DESARROLLO	4
H2	AUMENTA LA EFICIENCIA DE TU GESTIÓN DE PRODUCTOS	4
Bold/Strong	Scrum	5
Italic/em	* * * *	7
Alt Text	Not found	--

MOZ  Get Link Metrics with a FREE Moz Community Account [Create Account](#)

On-Page Elements **General Attributes** Link Metrics Markup Http Status

Tag/Location	Content
Meta Robots	Not found
Rel="canonical"	Not found
Page Load Time	0.871 seconds
Google Cache URL	http://google.com/search?q=cache:https://scrumevaluator.herokuapp.com/
IP Address	52.22.170.144
Country	United States

MOZ  PA: 1 0 links DA: 1 Spam Score: 0/10 [Unlock More Features with MozBar Premium](#) [Try Free](#)

On-Page Elements General Attributes **Link Metrics** Markup Http Status

Page Attributes	URL	Subdomain	Root Domain	Unlock More Features with MozBar Premium Try Free
Page Authority (PA)	1	--	--	
Domain Authority	--	--	1	
External Followed Links	0			
Linking Root Domains				
Total Links	0			

Anexo 6: Evaluación de calidad - herramienta Metric Spot

Análisis Web de scrumevaluator.herokuapp.com

[Descargar PDF](#)
[Seguimiento](#)
[Historial](#)

23.8%

PUNTUACIONES

Autoridad SEO:	0.0%	SEO Básico:	27.5%
Contenido:	28.6%	Usabilidad:	43.2%
Aspectos Técnicos:	50.0%		

Analiza la estrategia de tus competidores con nuestro análisis comparativo de webs. ¡Desmárcate de la competencia!

[Análisis Comparativo \(Pro\)](#)

On-Page Optimization

Desktop Speed Score 34.0%

Mobile Speed Score 34.0%

Mobile Usability Score 100.0%

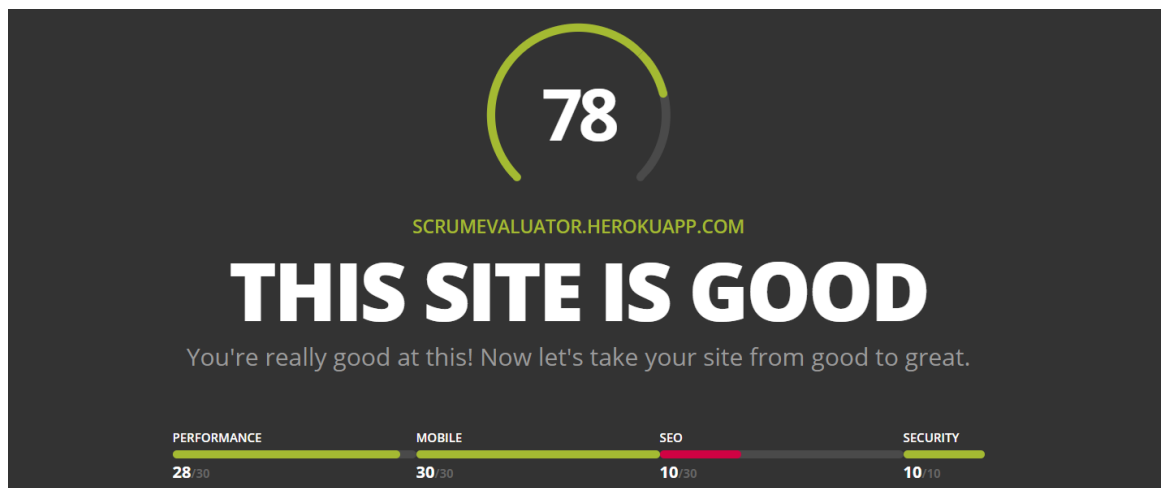
This website loads **15** resources across **1** unique hosts

7 JavaScript resources loaded

7 CSS resources loaded

TYPE	SIZE	PERCENT
HTML	1.7 KB	0.1%
CSS	898.6 KB	34.8%
Scripts	1.6 MB	65.2%
Images	0.0 B	00%
Other	0.0 B	00%
TOTAL	2.5 MB	100.0%

Anexo 7: Evaluación de calidad - herramienta Website grader



28/30

PERFORMANCE

Optimizing your website's performance is crucial to increasing traffic, improving conversion rates, generating more leads, and increasing revenue.

PAGE SIZE



60_{KB}

So fast! So light!

The heavier the site page, the slower the load. For optimal performance, try to keep page size below 3MB.

[READ MORE >](#)

PAGE REQUESTS



5

You make this look easy.

The more HTTP requests your website makes, the slower it becomes. Combine files to minimize the number of requests made.

[READ MORE >](#)

PAGE SPEED



0.3_{Sec}

Zoom zoom. Nice work.

Best-in-class webpages should load within 3 seconds. Any slower and visitors will abandon your site, reducing conversions and sales.

[READ MORE >](#)



BROWSER CACHING

Wowee. Your web caching is world class.



PAGE REDIRECTS

Perfect! No page redirects detected.



COMPRESSION

This website needs to go on a diet.



RENDER BLOCKING

Can't ask for a much faster site than that.

30 /30

MOBILE

Traffic from mobile devices is growing fast. Optimize your website for mobile or you'll miss out on valuable traffic, leads, and revenue.



RESPONSIVE



Fantastic! You're ready to face the future.

Responsive design gives you a bump in search rankings for searches on mobile devices.

[READ MORE >](#)

VIEWPORT



Now that's a good looking viewport.

The viewport lets you control your page width and scale on different device types.

[READ MORE >](#)

10 /30

SEO

Make sure your website is easy for users to discover—and easy for search bots to understand—with better page titles, headings, and meta descriptions.



PAGE TITLES

Splendid page titles.

Page titles should be no longer than 70 characters in length and not repeat keywords.

[READ MORE >](#)



META DESCRIPTION

The only way is up.

Meta descriptions should be no longer than 300 characters in length and should be relevant to the page.

[READ MORE >](#)



HEADINGS

Heads up! There's something missing.

Heading tags distinguish headings from core page content.

[READ MORE >](#)



SITEMAP

Hello? Where am I?

Site maps help users navigate your site quickly and easily.

[READ MORE >](#)

10 /10

SECURITY

SSL CERTIFICATE



Safe and secure. Nice job.

SSL certificates protect websites from attacks and give visitors confidence that your site is authentic and trustworthy.

[READ MORE >](#)

Anexo 8: Evaluación de calidad - herramienta ReadyMobi

Webpage Test Results

Click on a test below to view its details and recommended actions

MAJOR FAILS

3

MINOR FAILS

2

PASSES

33

- Gzip Encoding
- CSS Measurements
- Caching Control
- Favicon
- CSS Sprites
- Viewport Meta
- Image Crunch
- CSS Import
- Cookie Size
- Redirect Error
- DOM Too Large
- ETag Support
- Duplicate Resources
- DNS Lookups
- CSS Expression
- JavaScript Placement
- Stylesheet Placement
- Empty Image
- Table Nested
- Tables
- Internal CSS Directives
- Styled Markup
- Popups
- Page Title
- Inline JavaScript
- Applets
- Image Alt Tag
- Frames
- HTML Minimize
- JavaScript Minimize
- CSS Minimize
- Image Specify Size
- Image Resizing
- Image Map
- External Resources
- Input Type
- Valid Markup
- Charset

✘ MAJOR FAIL

gzip encoding

HTTP response should be compressed using server-side configuration of GZIP or similar compression technology

1 Gzip transport encoding not detected.

[How to fix it](#)