

UNIVERSIDAD TÉCNICA DEL NORTE



Facultad de Ingeniería en Ciencias Aplicadas

Carrera de Ingeniería en Sistemas Computacionales

**DESARROLLO DE LOS MÓDULOS E-LEARNING Y EVALUACIONES COMO
COMPONENTES DEL ENTORNO VIRTUAL DE APRENDIZAJE INTEGRADO (EVAI)
PARA LA EMPRESA IEREC**

Trabajo de grado previo a la obtención del título de Ingeniero en Sistemas Computacionales

Autor:

Erik Omar Morocho Troya

Director:

Msc. Cosme MacArthur Ortega Bustamante

Ibarra – Ecuador

2022



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO		
CÉDULA DE IDENTIDAD:	1726103631	
APELLIDOS Y NOMBRES:	MOROCHO TROYA ERIK OMAR	
DIRECCIÓN:	TABACUNDO, EL ROSARIO	
EMAIL:	eomorochot@utn.edu.ec	
TELÉFONO FIJO:	N/A	TELÉFONO MÓVIL: 0983508239

DATOS DE LA OBRA	
TÍTULO:	DESARROLLO DE LOS MÓDULOS E-LEARNING Y EVALUACIONES COMO COMPONENTES DEL ENTORNO VIRTUAL DE APRENDIZAJE INTEGRADO (EVAI) PARA LA EMPRESA IEREC.
AUTOR (ES):	MOROCHO TROYA ERIK OMAR
FECHA: DD/MM/AAAA	17/06/2022
SOLO PARA TRABAJOS DE GRADO	
PROGRAMA:	<input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSGRADO
TITULO POR EL QUE OPTA:	INGENIERIA EN SISTEMAS COMPUTACIONALES
ASESOR /DIRECTOR:	MSC. COSME MACARTHUR ORTEGA BUSTAMANTE

2. CONSTANCIAS

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 21 días del mes de Julio de 2022.

EL AUTOR:

ERIK OMAR
MOROCHO
TROYA

Firmado
digitalmente por
ERIK OMAR
MOROCHO TROYA
Fecha: 2022.07.21
16:19:31 -05'00'

Nombre: Erik Morocho

CERTIFICADO DEL DIRECTOR DE TRABAJO DE GRADO

UNIVERSIDAD TÉCNICA DEL NORTE



FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN DEL DIRECTOR

Por medio del presente yo Ing. Cosme Ortega, MSc, certifico que el Sr. Erik Omar Morocho Troya, portador de la cédula de ciudadanía Nro. 1726103631. Ha trabajado en el desarrollo del proyecto de tesis **“Desarrollo de los módulos E-learning y evaluaciones como componentes del Entorno Virtual de Aprendizaje Integrado (EVAI) para la empresa IERec.”**, previo a la obtención del título de Ingeniería en Sistemas Computacionales, lo cual ha realizado en su total responsabilidad.

Es todo cuanto puedo certificar en honor a la verdad.

Atentamente:



Ing. Cosme Ortega, MSc
DIRECTOR DE TESIS

CERTIFICADO

Me permito informar a Ustedes que el señor **ERIK OMAR MOROCHO TROYA**, con cédula de ciudadanía Nro. **1726103631**. Estudiante de la Universidad Técnica del Norte, ha realizado su Trabajo de Grado con el tema: “**DESARROLLO DE LOS MÓDULOS E-LEARNING Y EVALUACIONES COMO COMPONENTES DEL ENTORNO VIRTUAL DE APRENDIZAJE INTEGRADO (EVAI) PARA LA EMPRESA IEREC**” cumpliendo con todos los requisitos reglamentarios de aprobación de la empresa, con cualidades de responsabilidad y profesionalismos.

Para efecto, se extiende el presente CERTIFICADO DE CULMINACIÓN DEL SOFTWARE, en la ciudad de Quito a los 3 días del mes de febrero de 2022.

Agradezco su atención.

Atentamente,



Ing. Guillermo Pérez Msc
CEO IER ECUADOR



0987 127 655
+593987127655



ier.energiasrenovables@gmail.com



Los Juncos 471 y Eloy Alfaro
Quito - Ecuador

DEDICATORIA

A mi madre Rosario, que ha sido parte integral de mi vida y siempre ha velado por mi salud, educación y economía, convirtiéndola en mi motivo de orgullo e inspiración para lograr mis metas y objetivos.

AGRADECIMIENTO

A mi docente tutor el Msc. Cosme Ortega que fue la guía y apoyo constante en el transcurso de la realización de mi proyecto de grado.

A la empresa IERec por haber confiado en mis habilidades y aptitudes para la realización de mi proyecto de titulación.

A mis docentes de trabajo de grado que fueron de gran ayuda en el planteamiento y seguimiento de mi proyecto de titulación.

A mi familia que a pesar de los tiempos difíciles me han apoyado en la culminación de mis estudios.

TABLA DE CONTENIDO

Introducción.....	XIX
Antecedentes	XIX
Situación actual.....	XIX
Prospectiva.....	XX
Problema	XX
Objetivos.....	XXI
Alcance.....	XXII
Justificación.....	XXIX
Riesgos.....	XXX
Contexto	XXXIII
CAPITULO I	1
MARCO TEÓRICO	1
1.1 Herramientas para la implantación de plataformas de E-Learning y Evaluaciones..	1
1.1.1 Concepto de E-Learning.....	1
1.1.2 Herramientas y tecnologías para E-Learning.....	1
1.1.3 Concepto de evaluación	2
1.1.4 Importancia de una evaluación.....	3
1.1.5 Herramientas y tecnologías para la toma de evaluaciones	3

1.2	Backend y Frontend	4
1.2.1	Concepto de Backend	4
1.2.2	¿Qué es API REST?	5
1.2.3	JavaScript y TypeScript	5
1.2.4	Node.Js y Express	7
1.2.5	MongoDB	8
1.2.6	Concepto de Frontend	8
1.2.7	Concepto de Programación Orientada a Objetos	9
1.2.8	React Js	10
1.2.9	React Hooks.....	11
1.2.10	Redux.....	11
1.3	DevOps y Computación en la nube	12
1.3.1	Concepto de DevOps	12
1.3.2	Que es un gestor de versiones.....	14
1.3.3	Git.....	15
1.3.4	GitHub	15
1.3.5	Docker.....	15
1.3.6	Concepto de computación en la nube.....	16
1.3.7	Que es DigitalOcean.....	18

1.4	Metodología XP y Norma ISO 25010 enfocada a la usabilidad.....	18
1.4.1	Metodología de desarrollo XP	18
1.4.2	Norma ISO 25010	22
CAPITULO II		23
DESARROLLO DE LA aplicación WEB		23
2.1	Planificación	23
2.1.1	Historias de usuario	24
2.2	Diseño	31
2.2.1	Arquitectura de la aplicación web.....	32
2.2.2	Diagrama de procesos.....	32
2.2.3	Modelo de la base de datos	40
2.2.4	Casos de uso	41
2.3	Codificación.....	47
2.3.1	Configuración del entorno de desarrollo	47
2.3.2	Creación y ejecución de la base de datos	48
2.3.3	Creación del backend de los módulos de E-learning y evaluaciones	49
2.3.4	Configuración y preparación para el desarrollo frontend.....	54
2.4	Pruebas	58
2.4.1	Pruebas con Postman en el backend.....	59

2.4.2	Pruebas con React Developer Tools en el frontend	61
2.4.3	Pruebas con Redux DevTools en el frontend	62
2.4.4	Pruebas con DevTools del navegador web en el frontend	63
Capitulo III	65
VALIDACIÓN de resultados	65
3.1	Metodología.....	65
3.1.1	Método de evaluación.....	65
3.2	Análisis de resultados	67
3.3	Interpretación de resultados	75
Conclusiones	78
Recomendaciones	78

ÍNDICE DE FIGURAS

Figura 1 Árbol de problemas	XXI
Figura 2 Flujograma Módulo de E-learning	XXIV
Figura 3 Flujograma Módulo de Evaluaciones	XXV
Figura 4 Arquitectura de software	XXVIII
Figura 5 Metodología Ingenieril	XXIX
Figura 6 Configuración del umbral para la identificación de los niveles de riesgo.....	XXXI
Figura 7 Valores para representación de la matriz de riesgos	XXXII
Figura 8 Grafica de la matriz de riesgos	XXXIII
Figura 9 metodología Extreme Programming	19
Figura 10 Norma ISO 25010. Fuente: (ISO 25010, 2018.).....	23
Figura 11 Programación Extrema.....	24
Figura 12 Arquitectura de la aplicación web	32
Figura 13 Diagrama del subproceso de inicio de sesión	33
Figura 14 Diagrama del subproceso de pago de cursos	34
Figura 15 Diagrama del subproceso de categorías	35
Figura 16 Diagrama del subproceso de cursos	36
Figura 17 Diagrama del subproceso de evaluaciones.....	37

Figura 18 Diagrama de procesos del usuario en los módulos de E-learning y Evaluaciones	38
Figura 19 Diagrama de procesos del instructor en los módulos de E-learning y Evaluaciones	39
Figura 20 Diagrama de procesos del administrador en los módulos de E-learning y Evaluaciones.....	40
Figura 21 Caso de uso del usuario administrador	41
Figura 22 Caso de uso del usuario instructor	43
Figura 23 Caso de uso del usuario	45
Figura 24 Modelo de la base de datos en MongoDB.....	48
Figura 25 Creación de una nueva app con Node.Js.....	49
Figura 26 Archivo de configuración de una aplicación echa con Node.Js y TypeScript.....	50
Figura 27 Estructura de carpetas de los módulos web.....	50
Figura 28 Archivos de los modelos de la base de datos	51
Figura 29 Estructura para la creación de modelos en MongoDB.....	51
Figura 30 Archivos que conforman los controladores de la API REST	52
Figura 31 Estructura para la creación de controladores de una API REST	53
Figura 32 Archivos que conforman las rutas de la API REST	53
Figura 33 Estructura para la creación de rutas de una API REST	54
Figura 34 Estructura de carpetas al crear una nueva app de React con TypeScript	54

Figura 35 Archivos de configuración para establecer reglas y estandarización de código limpio	55
Figura 36 Estructura diseño atómico en React, Redux y TypeScript.....	56
Figura 37 Hooks customizados en React.....	56
Figura 38 Estructura de la carpeta Pages en el frontend	57
Figura 39 Componentes visuales del frontend.....	57
Figura 40 Redux como control del stado de React.....	58
Figura 41 Validación de una API con Postman.....	61
Figura 42 Herramienta para validación de componentes en React	62
Figura 43 Herramienta para la validación de estado de React con Redux	63
Figura 44 Validación de carga de contenido con herramientas del navegador	64
Figura 45 Resultados de la pregunta 1 de la encuesta SUS.....	67
Figura 46 Resultados de la pregunta 2 de la encuesta SUS.....	68
Figura 47 Resultados de la pregunta 3 de la encuesta SUS.....	69
Figura 48 Resultados de la pregunta 4 de la encuesta SUS.....	70
Figura 49 Resultados de la pregunta 5 de la encuesta SUS.....	70
Figura 50 Resultados de la pregunta 6 de la encuesta SUS.....	71
Figura 51 Resultados de la pregunta 7 de la encuesta SUS.....	72
Figura 52 Resultados de la pregunta 8 de la encuesta SUS.....	73

Figura 53 Resultados de la pregunta 9 de la encuesta SUS.....	73
Figura 54 Resultados de la pregunta 10 de la encuesta SUS.....	74
Figura 55 Resumen de la encuesta SUS aplicada a los módulos de E-learning y evaluaciones.	74
Figura 56 Grado de calificación SUS Fuente: (Sauro & Lewis, 2016)	77

ÍNDICE DE TABLAS

Tabla 1 Contexto de temas de tesis	XXXIII
Tabla 2 Participantes de la metodología.	24
Tabla 3 Historia de usuario diseño de la base de datos	24
Tabla 4 Historia de usuario desarrollo del backend	25
Tabla 5 Historia de usuario diseño y construcción de las vistas de inicio	25
Tabla 6 Historia de usuario diseño y construcción de la vista para autenticación de usuarios.....	26
Tabla 7 Historia de usuario diseño y construcción de la vista de inicio del usuario	27
Tabla 8 Historia de usuario diseño y construcción de la vista del perfil de usuario	27
Tabla 9 Historia de usuario diseño y construcción de un dashboard para visualizar un curso	28
Tabla 10 Historia de usuario diseño y construcción del dashboard para el usuario instructor	28
Tabla 11 Historia de usuario diseño y construcción del componente visual de los cursos...29	
Tabla 12 Historia de usuario diseño y construcción del dashboard del administrador.....	30
Tabla 13 Historia de usuario diseño y construcción del componente visual carrito de compras	30
Tabla 14 Historia de usuario diseño y construcción del componente visual para realizar pagos.....	31
Tabla 15 Caso de uso del usuario administrador	41

Tabla 16 Caso de uso del usuario instructor	43
Tabla 17 Caso de uso del usuario	45
Tabla 18 Resultados de la encuesta en la plataforma Microsoft Forms	66
Tabla 19 Valoración de cada una de las opciones de una encuesta SUS	75
Tabla 20 Sumatoria de los valores de cada pregunta para 58 encuestados.....	75
Tabla 21 Resultados de X0, Y0, y grado SUS por persona para 58 encuestas.....	76

RESUMEN

El proyecto es una solución de software con un desarrollo a medida que planea solucionar los problemas de atracción y obtención de clientes en la empresa IERec mediante la implementación de un sistema que ayude a la oferta, publicación y venta de cursos en línea mediante una aplicación web.

Para la automatización y mejora continua de los procesos se propuso la realización de un módulo de E-learning y un módulo de evaluaciones como componente de un entorno virtual de aprendizaje integrado (EVAI), el cual estará basado en base en los requerimientos de usuario final la empresa IERec.

Para el desarrollo de los módulos web se utilizó una arquitectura cliente servidor dividida en dos secciones backend y frontend. El desarrollo del servidor (backend) fue realizado con las siguientes tecnologías: Git, JavaScript, TypeScript, Node.js, Express.js, MongoDB y para la creación de interfaces de usuario (frontend) se utilizó tecnologías como: Git, TypeScript, React.js, Redux, Sass, HTML. Para su puesta en marcha y subida a producción o entorno de trabajo real se lo realizo en un servidor en la nube llamado DigitalOcean mediante la utilización de las siguientes tecnologías: GitHub, Docker y Mongo Atlas.

Para la validación de resultado se procedió a realizar una encuesta de usabilidad enfocada en la ISO 25010 mediante la utilización de un cuestionario SUS, en el cual se pudo obtener un 82,67, que representa un grado B, lo cual significa que la aplicación a nivel de usabilidad tiene un grado de aceptación por parte de los usuarios.

ABSTRACT

The project is a software solution with a custom development that plans to solve the problems of attracting and obtaining clients in the IERec company by implementing a system that helps the offer, publication and sale of online courses through a web application.

For the automation and continuous improvement of the processes, it was proposed to carry out an E-learning module and an evaluation module as a component of an integrated virtual learning environment (EVAL), which will be based on the end user requirements.

For the development of the web modules, a client-server architecture divided into two sections, backend and frontend, was used. The development of the server (backend) was carried out with the following technologies: Git, JavaScript, TypeScript, Node.js, Express.js, MongoDB and for the creation of user interfaces (frontend) technologies such as: Git, TypeScript, React were used. .js, Redux, Sass, HTML. For its start-up and upload to production or real work environment, it is done on a cloud server called DigitalOcean using the following technologies: GitHub, Docker and Mongo Atlas

For the validation of the result, a usability survey focused on ISO 25010 was carried out through the use of a SUS questionnaire, in which a score of 82.67 could be obtained, which represents a B grade, which means that the application to level of usability has a good acceptance by users.

Introducción

Antecedentes

La empresa IERec, fue fundada como una empresa que brinda servicios de consultoría y capacitación de Ingeniería en Energías Renovables, utilizando procesos de negocio tradicionales de manera presencial en el Ecuador, ganando reconocimiento en el mercado, pero debido a la reciente pandemia y cambio en el estilo de vida, la empresa IERec toma la decisión de incluir la red social Facebook, como una herramienta para promocionar y difundir sus cursos en línea mediante la plataforma Zoom.

Uno de los problemas que afronta la empresa IERec, es la validación de procesos de pago mediante comprobantes enviados, a través de la Red Social WhatsApp, lo que genera mucha desconfianza, provocando que la empresa pierda futuros clientes.

La empresa IERec al no contar con una plataforma que permita crear, subir, ofertar y vender cursos en línea (modulo E-learning) relacionados con la Ingeniería en Energías Renovables provoca que otras empresas que si cuentan con medios tecnológicos abarquen gran parte del mercado nacional e internacional, sumado a esto la poca existencia de empresas dedicadas a brindar este tipo de cursos en línea.

Situación actual

La empresa IERec actualmente se dedica a dar capacitaciones y asistencia técnica de Ingeniería en Energías Renovables, utilizando varias aplicaciones como Zoom, Meet y Redes Sociales como Facebook y WhatsApp, así que, al no contar con una plataforma la cual contenga varios módulos integrados como por ejemplo pagos en línea, cursos online, evaluaciones dinámicas de cada curso, videoconferencias, gestión académica, la empresa tiende a perder futuros clientes, por ende generar una ventaja competitiva ante la sociedad y el mercado en el que se desenvuelve.

Prospectiva

Mediante la implementación del módulo de E-learning se pretende incrementar la ventaja competitiva de la empresa IERec, ya que le permitirá crear, ofertar y vender cursos en línea.

También se espera que la empresa pueda ampliar su tasa de clientes de manera nacional e internacional, por el hecho de que al estar en internet cualquier persona del mundo puede acceder y conocer los cursos que la empresa IERec oferta.

Con el módulo de toma de evaluaciones se pretende dar un apoyo al módulo de E-learning, puesto que este contara con pruebas a final de cada sección o a un examen de todo el curso que ayude a evaluar los conocimientos de los usuarios, además podrá ayudar como componente individual que mejore el proceso de gestión para la toma de pruebas y exámenes.

Problema

La empresa IERec, tiendo a perder clientes y expansión de mercado debido a que no cuenta con una plataforma la cual permita integrar varios módulos como E-learning y evaluaciones.

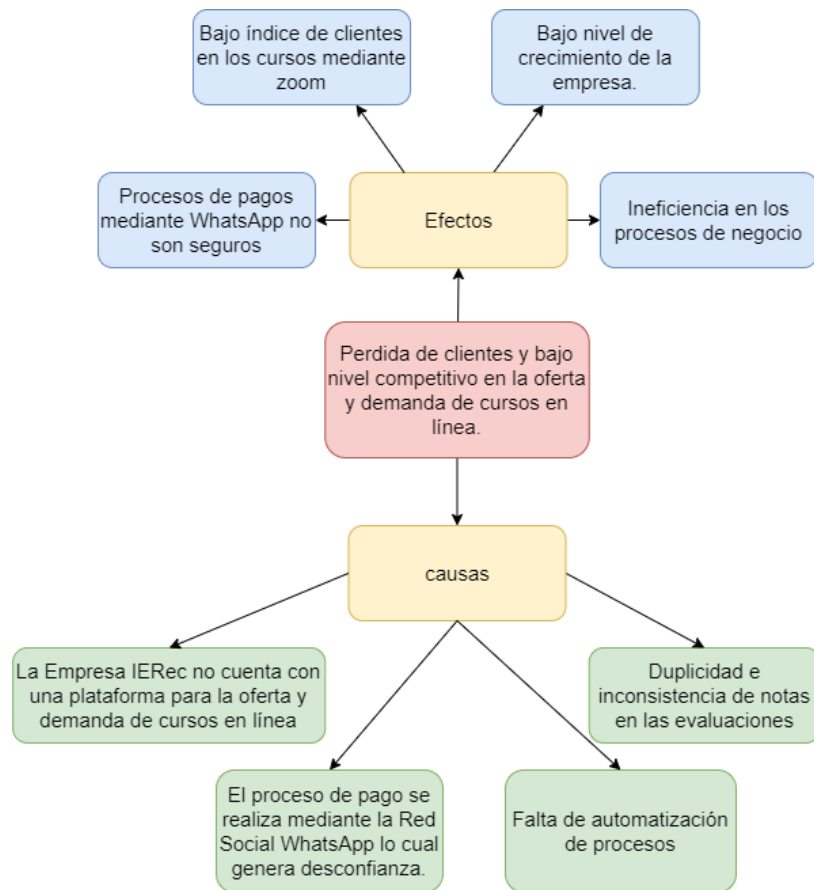


Figura 1 Árbol de problemas

Objetivos

Objetivo General

Desarrollar los módulos de E-learning y evaluación como componentes del Entorno Virtual de Aprendizaje Integrado (EVAI) para la empresa IERec.

Objetivos Especifico

- Desarrollar un marco teórico sobre conceptos, herramientas y software para la implementación de los módulos de E-learning y evaluaciones.

- Implementar los módulos E-learning y evaluaciones como componentes del Entorno Virtual de Aprendizaje Integrado (EVAI).
- Validar el software mediante la norma ISO 25010 enfocada hacia la usabilidad.

Alcance

Se realizará dos módulos uno de E-learning y el otro de evaluaciones, los cuales formarán parte del Entorno Virtual de Aprendizaje Integrado (EVAI).

Módulo de E-learning

Este módulo debe contar con las siguientes funcionalidades creación, publicación, actualización, venta y eliminación de cursos proporcionados por la empresa, este módulo debe tener una sección donde los cursos comprados se pueden ver por separado en secciones que contengan una serie de videos y pruebas tomadas en algunos módulos del curso.

El módulo de E-learning contará con tres tipos de roles:

Administrador

- Podrá visualizar clientes registrados.
- Podrá crear, visualizar, actualizar y eliminar usuarios, cursos y categorías.
- Podrá visualizar el historial de ventas.

Instructor

- Podrá actualizar la información acerca del curso.
- Podrá crear, visualizar, actualizar y eliminar módulos, contenidos de cada módulo.

Usuario

- Podrá ver un catálogo de cursos.
- Podrá filtrar los cursos por categoría o hacer una búsqueda por el nombre de un curso.
- Podrá ver detalles de cada curso.
- Podrá hacer compras de los cursos en línea.
- El usuario podrá visualizar los cursos comprados y tener acceso a todo el contenido del curso.

El módulo de toma de evaluaciones contará solo con un rol:

Instructor

- Podrá crear, visualizar, actualizar y eliminar evaluaciones, preguntas y respuestas de una pregunta.
- Las evaluaciones podrán ser asignadas a los módulos de cada curso.

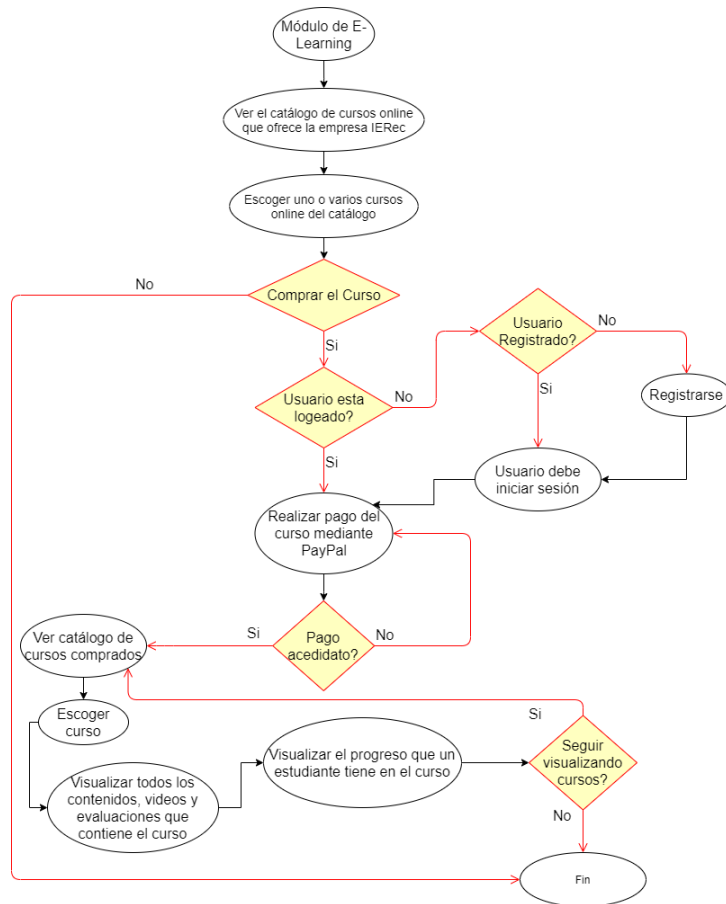


Figura 2 Flujograma Módulo de E-learning

El módulo de toma de evaluaciones contara solo con un rol:

Profesor

- Podrá crear preguntas las cuales podrán ser de opción múltiple, una sola opción entre varias o respuesta mediante texto a las cuales se les deberá poner un grado de dificultad entre alto, medio o bajo, así poder crear un banco de preguntas.
- Podrá visualizar, filtrar y buscar preguntas.
- Podrá actualizar y eliminar preguntas.

- Podrá crear pruebas o exámenes aleatorios con el banco de preguntas que tenga el profesor o podrá seleccionar las preguntas que el profesor desee tomar en cada prueba o examen.
- Se podrá visualizar y buscar las pruebas que tiene el profesor.
- Se podrá actualizar y eliminar las pruebas o exámenes.

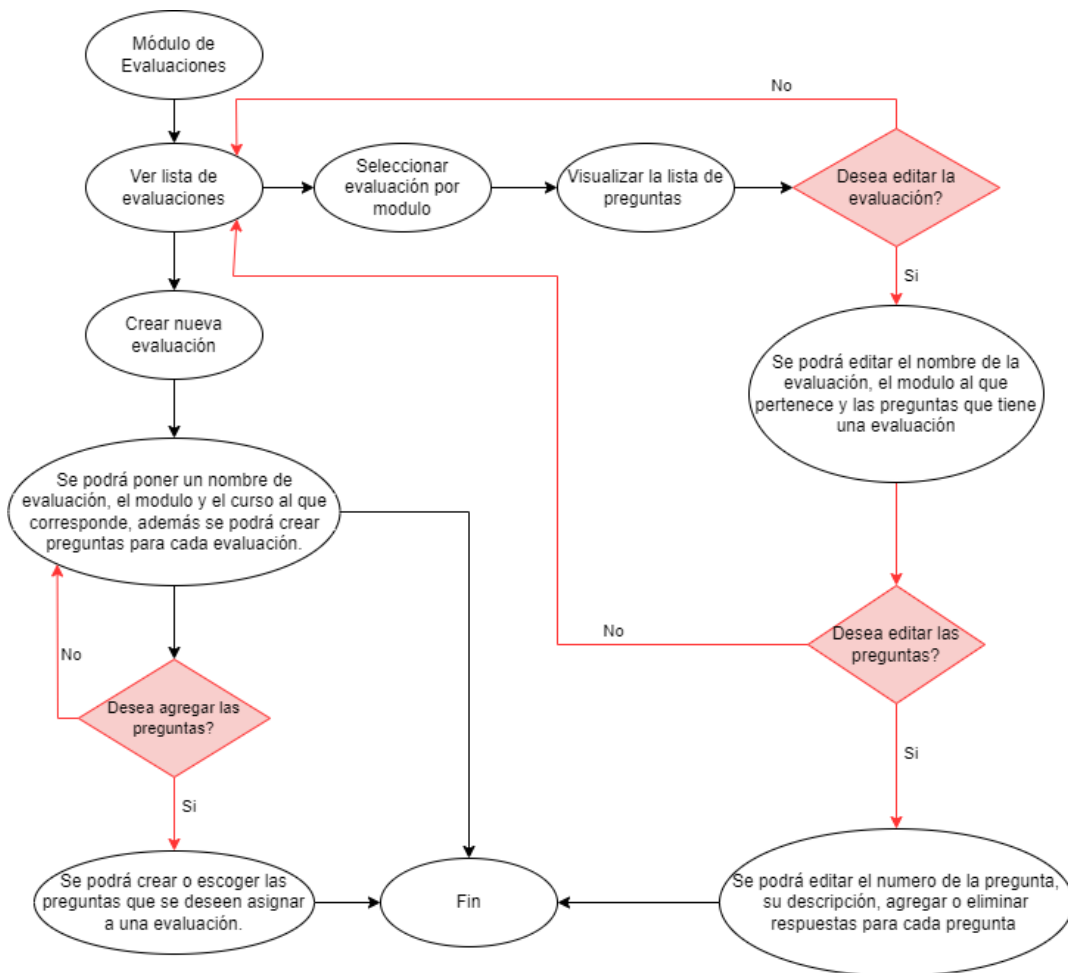


Figura 3 Flujograma Módulo de Evaluaciones

Arquitectura

La aplicación será creada como una arquitectura cliente–servidor o también llamado Backend y Frontend.

Backend

El back-end será construido en Node.js mediante el uso de paquetes y librerías necesarias para el desarrollo (Node.js, 2020), ExpressJS esta librería nos ayuda a la configuración del servidor (*Direccionamiento de Express*, 2018), MongoDB nos ayudará como base de datos en donde almacenaremos toda la información de la aplicación web (*Welcome to the MongoDB Documentation — MongoDB Documentation*, 2019), TypeScript nos ayudará como lenguaje de tipado, ya que JavaScript no maneja tipos de datos por ende complica un poco la detección de errores y acceso a los datos, a través de la integración de todas estas tecnologías se lograra construir una API REST la cual podrá ser consumida por cualquier tipo de cliente (*Typescript: The Starting Point for Learning Typescript*, 2019).

Aquí también se procederá a la utilización de ESLint y Prettier los cuales nos permiten establecer una serie de reglas para controlar las buenas prácticas de programación (JS Foundation and other contributors, 2019) y código limpio que es muy fundamental a la hora de construir software de calidad (*What Is Prettier? · Prettier*, 2019).

Docker permite crear contenedores ligeros y portables para las aplicaciones de software que puedan ejecutar en cualquier máquina con Docker instalado independientemente del sistema operativo de una máquina o servidor en la nube, facilitando así los despliegues (Hykes, 2016).

Frontend

El frontend será construido en ReactJS una librería de JavaScript que nos ayuda a simplificar la programación del lado del cliente y crear componentes reactivos basados en pequeñas porciones de código reutilizable haciendo que la navegación sea más rápida que otras aplicaciones web y eficiente con la capacidad de crear aplicaciones wpa permitiendo controlar el estado de una aplicación sin necesidad de refrescar la página (*React – Una Biblioteca de JavaScript Para Construir Interfaces de Usuario*, 2017).

Otra tecnología que nos ayuda a manejar los estados de ReactJS es Redux que mediante esta librería podemos crear un store donde podemos almacenar las variables que se requieran en uno o más componentes de la aplicación haciendo que el manejo de estados

de react sean manejados por acciones con un tipo de instrucción para ser devueltos por un selector el cual podrá traer las variables que uno se solicita (Srinivasa & Muppalla, 2015).

TypeScript nos ayudará como lenguaje de tipado de JavaScript para simplificar el trabajo de detección de errores y acceso a los datos (*Typescript: The Starting Point for Learning Typescript*, 2019).

Aquí también se procederá a la utilización de ESLint y Prettier los cuales nos permiten establecer una serie de reglas para controlar las buenas prácticas de programación (JS Foundation and other contributors, 2019) y código limpio que es muy fundamental a la hora de construir software de calidad (*What Is Prettier? · Prettier*, 2019).

Para la construcción de los módulos se utilizará las siguientes tecnologías:

- Base de datos no relacional MongoDB.
- Lenguaje de programación JavaScript con el runtime de Node.js para la construcción del Back-end.
- Framework de programación React para la construcción del Frontend.
- Integración de TypeScript como lenguaje de tipado en JavaScript.
- Tecnología de manejo de estado ReduxJs.
- Framework de programación Express para la construcción del Back-end.
- Tecnología para la creación de contenedores con el runtime de Docker.
- Git y Github.

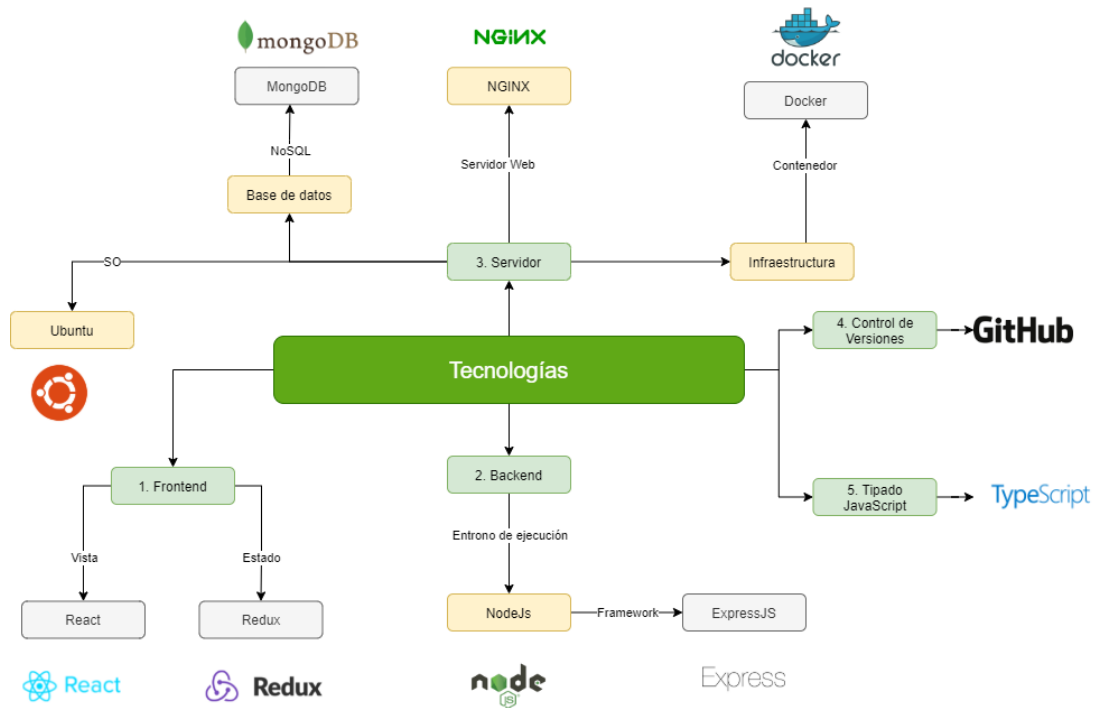


Figura 4 Arquitectura de software

Metodología

Para el desarrollo y la implantación de los módulos de E-learning y de toma de evaluaciones se utiliza la investigación aplicada la cual nos permitirá mediante herramientas tecnologías crear un sistema aporte a la competitividad en venta y oferta de cursos en línea de la empresa IERec.

También se pretende seguir la metodología ingenieril y en específico los siguientes puntos de la metodología.

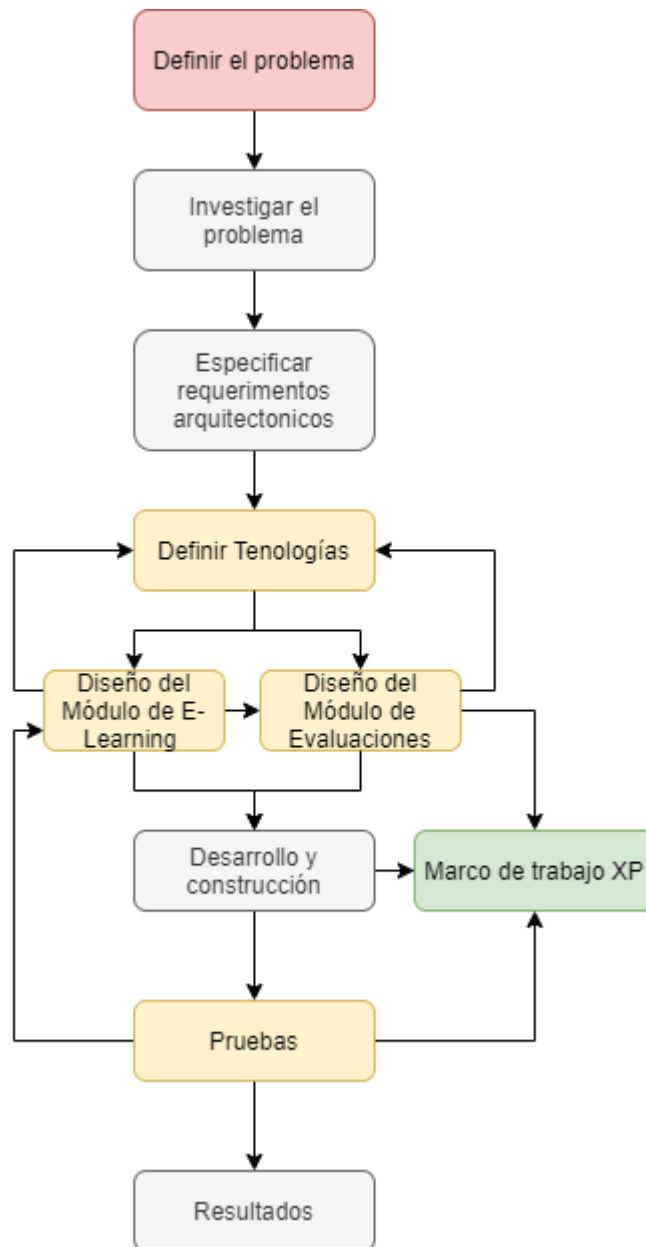


Figura 5 Metodología Ingenieril

Justificación

La empresa IERec, requiere una solución de software vertical enfocado a sus objetivos de negocio que vayan de acorde a la misión y visión de la empresa, debido a que en la actualidad los procesos de negocios son realizados de manera manual, la promoción, publicidad y difusión de contenidos se realiza a través de Facebook. El pago de

capacitaciones o cursos en línea se realiza vía transferencia bancaria y el comprobante de pago es verificado mediante la red social WhatsApp.

Además, el presente proyecto se enfoca en el cumplimiento de dos Objetivos de desarrollo sostenible (ODS) los cuales son:

Objetivo N°4: Educación de Calidad: Se pretende el cumplimiento de este objetivo el cual estipula que se espera “Garantizar una educación inclusiva, equitativa y de calidad y promover las oportunidades de aprendizaje para todos de aquí a 2030” (UNESCO, 2019).

Objetivo N°9: Industria, Innovación e Infraestructura: Según el objetivo número nueve se espera el cumplimiento “La industrialización inclusiva y sostenible, junto con la innovación y la infraestructura, pueden dar rienda suelta a las fuerzas económicas dinámicas y competitivas que generan el empleo y los ingresos. Estas desempeñan un papel clave a la hora de introducir y promover nuevas tecnologías, facilitar el comercio internacional y permitir el uso eficiente de los recursos” (ONU, 2015).

Justificación Tecnológica. – Con la evolución de la tecnología constantemente, los sistemas informáticos han ido cambiando, dejando atrás sistemas monolíticos, sin escalabilidad, difíciles de mantener. Con el apareamiento de la era de la web y la industria 4.0, se entiende que aquellas empresas que no tienen presencia en internet tienden a no desarrollarse apropiadamente y por consecuente desaparecer por falta de presencia en el mercado (Demartini & Benussi, 2017).

Riesgos

R1. Versiones de paquetes desactualizados o ya no compatibles.

R2. DigitalOcean suba los precios de los servidores para la aplicación web.

R3. La compatibilidad de todas las herramientas tecnológicas que se pretende usar para el desarrollo de esta aplicación web.

R4. La empresa IERec ya no desee que se le realice la aplicación web.

R5. Que algunos de mis compañeros de que también realizan módulos no terminen el mismo tiempo que yo.

R6. Que algunos de mis compañeros de que también realizan módulos no terminen el mismo tiempo que yo.

R7. Que no se encuentre una manera económica de tener videos privados en la sección de cursos del módulo de E-learning.

Impacto			Probabilidad			Umbral de nivel		
Alto	10		Alto	10		Alto	≥	21 
Medio alto	7		Medio alto	7		Medium	≥	11 
Medio bajo	4		Medio bajo	4		Bajo	≥	1 
Bajo	1		Bajo	1				

Figura 6 Configuración del umbral para la identificación de los niveles de riesgo

Matriz de evaluación de riesgos

Conjunto de riesgos		Mi conjunto de riesgos	
R1	Impacto: Bajo (1) Probabilidad: Medio alto (7)	Nivel: Bajo (7)	Ver matriz
R2	Impacto: Bajo (1) Probabilidad: Medio bajo (4)	Nivel: Bajo (4)	Ver matriz
R3	Impacto: Medio bajo (4) Probabilidad: Medio bajo (4)	Nivel: Medium (16)	Ver matriz
R4	Impacto: Alto (10) Probabilidad: Medio bajo (4)	Nivel: Alto (40)	Ver matriz
R5	Impacto: Medio alto (7) Probabilidad: Bajo (1)	Nivel: Bajo (7)	Ver matriz
R6	Impacto: Medio bajo (4) Probabilidad: Medio bajo (4)	Nivel: Medium (16)	Ver matriz
R7	Impacto: Medio alto (7) Probabilidad: Medio alto (7)	Nivel: Alto (49)	Ver matriz

Figura 7 Valores para representación de la matriz de riesgos

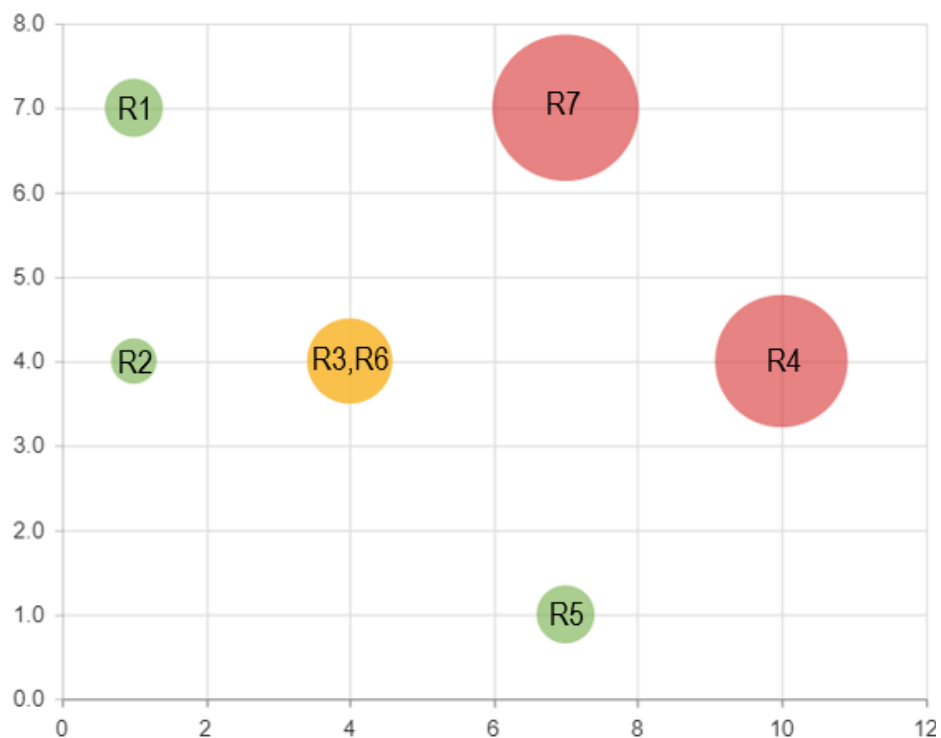


Figura 8 Grafica de la matriz de riesgos

Contexto

Tabla 1

Contexto de temas de tesis

Investigación	Enlace	Diferencia
E-learning para la creación de aula virtual en la Unidad Educativa "Monseñor Leónidas Proaño", en el año lectivo 2015-2016 del Cantón Espejo en la Provincia del Carchi	http://repositorio.utn.edu.ec/handle/123456789/5118	<ul style="list-style-type: none"> • Las herramientas de desarrollo no son las mismas. • El lenguaje de programación no es igual. • La arquitectura de software es mucho más robusta. • Se podrá tomar pruebas mediante un banco de preguntas.

<p>Estudio del contenedor Cloud Docker y propuesta de implementación para la plataforma Cloud FICA</p>	<p>http://repositorio.utn.edu.ec/handle/123456789/5775</p>	<ul style="list-style-type: none"> • Aquí se aplicará Docker. • El proyecto de titulación no es un estudio es algo práctico con el uso de varias tecnologías.
<p>Implementar una alternativa de preparación masiva para facilitar la formación en el área informática con un curso MOOC.</p>	<p>http://repositorio.utn.edu.ec/handle/123456789/6130</p>	<ul style="list-style-type: none"> • El módulo de E-learning se encargará de brindar cursos en línea, pero esos cursos en línea no serán de consumo gratuito sino de paga.
<p>Sistema de entorno virtual de enseñanza aprendizaje de la Universidad Técnica del Norte con los módulos de cursos, usuarios, recursos y tareas.</p>	<p>http://repositorio.utn.edu.ec/handle/123456789/3726</p>	<ul style="list-style-type: none"> • Se implementará un módulo específico para la toma de pruebas además el módulo de E-learning se podrá ver el progreso del curso tomado en línea.
<p>Sistema de entorno virtual enseñanza – aprendizaje de la universidad Técnica del Norte con la implementación de los módulos de cuestionarios, interacción estudiante – maestro, seguridad y auditoría.</p>	<p>http://repositorio.utn.edu.ec/handle/123456789/4355</p>	<ul style="list-style-type: none"> • Se utilizará un lenguaje de programación diferente. • Se utilizará una base de datos NoSql como es MongoDB • Se utilizará Docker para el uso de contenedores. • Se utilizará una librería de Frontend como React.

CAPITULO I

MARCO TEÓRICO

1.1 Herramientas para la implantación de plataformas de E-Learning y Evaluaciones

1.1.1 Concepto de E-Learning

El E-learning se lo puede entender cómo un proceso de enseñanza y aprendizaje a través de internet, lo cual permite que mediante esta modalidad exista una formación completamente a distancia donde uno o varios alumnos puedan acceder a distintos contenidos, actividades, tareas, entre otras cosas mediante el uso de internet (Guzmán Valle Alma Máter del Magisterio Nacional FACULTAD CIENCIAS & Pérez, 2019).

El negocio del E-Learning está convirtiendo de forma acelerada el modelo educativo tradicional sirviendo como herramienta que contribuye a mejorar la eficacia educativa. Al margen de eso, también se ha convertido en una potente industria con un gran futuro (Guzmán Valle Alma Máter del Magisterio Nacional FACULTAD CIENCIAS & Pérez, 2019).

1.1.2 Herramientas y tecnologías para E-Learning

Las herramientas para la construcción de una plataforma E-Learning realizan un papel fundamental en el cambio de los métodos de aprendizaje, obteniendo educación de calidad y oportunidad de aprender en un entorno interactivo. Estas herramientas son fáciles de personalizar por el instructor adecuando a las necesidades de los estudiantes que apoyan sus esfuerzos de aprendizaje electrónico (Guzmán Valle Alma Máter del Magisterio Nacional FACULTAD CIENCIAS & Pérez, 2019).

Herramientas que ayudan a crear sistemas interactivos para educación formativa en línea conocida como E-learning:

CMS es un software de aplicación diseñado para apoyar la enseñanza y aprender en un entorno educativo. CMS proporciona herramientas y un marco que permiten la creación fácil de contenido de un curso en línea, la enseñanza posterior y gestión de ese curso, incluida la diversa interacción con los estudiantes que toman un curso el CMS proporciona un marco que ayuda en la creación de contenidos de E-Learning como son la enseñanza, gestión de contenido en un curso, además facilita interacción con los alumnos matriculados en un curso (Guzmán Valle Alma Máter del Magisterio Nacional FACULTAD CIENCIAS & Pérez, 2019).

MOODLE es un entorno de aprendizaje dinámico orientado a la educación en línea por lo que es considerada como una herramienta para creación de plataformas E-Learning de código abierto, este sistema también se lo considera como un sistema de gestión de aprendizaje a través del cual el usuario puede publicar información en la web sin conocer nada de tecnología (Guzmán Valle Alma Máster del Magisterio Nacional FACULTAD CIENCIAS & Pérez, 2019).

JOOMLA es una plataforma que permite realizar sistemas de gestión de contenido de código abierto en la web, lo cual ayuda a la gestión del aprendizaje que proporciona el educador al cargar el contenido dando así el mejor ambiente de aprendizaje (Guzmán Valle Alma Máster del Magisterio Nacional FACULTAD CIENCIAS & Pérez, 2019).

OLAT es un software de código abierto, funciona como un sistema de gestión de aprendizaje basado en Java que ha sido desarrollado en 1999 en la Universidad de Zúrich. Proporciona la facilidad de aprendizaje en línea, síncrono y asíncrono comunicación entre el instructor y el alumno, intercambio de archivos y proporcionar mensajes de instancia (Guzmán Valle Alma Máster del Magisterio Nacional FACULTAD CIENCIAS & Pérez, 2019).

JOJO CMS es un sistema de gestión de cursos de código abierto basado en PHP y fue desarrollado en Auckland, Nueva Zelanda para construir sitios web, distribuidos bajo la licencia LGP. JOJO CMS, proporciona un marco para que los desarrolladores generen Sitios web y una interfaz de administración para editar la página (Guzmán Valle Alma Máster del Magisterio Nacional FACULTAD CIENCIAS & Pérez, 2019).

DRUPAL es un marco y curso modular gratuito y de código abierto. Sistema de gestión escrito en lenguaje de programación PHP con MySQL como back-end. Se utiliza como un back end sistema para muchos tipos diferentes de sitios web, que van desde pequeños blogs personales a grandes sitios corporativos y políticos (Guzmán Valle Alma Máster del Magisterio Nacional FACULTAD CIENCIAS & Pérez, 2019).

1.1.3 Concepto de evaluación

El término evaluación tiene muchos significados, diferentes motivos y, por supuesto, diferentes métodos de implantación. Hablamos de la evaluación como estrategia de medición, coordinación, calificación, acreditación, selección, docencia, y también de tipologías de evaluación diferentes, como la inicial, la diagnóstica, la formativa, la formadora, la sumativa, la acreditativa (Sanmarti Puig, 2020).

Sin embargo, tras una determinada medida de valoración, dos profesores no hacen la misma valoración de los mismos conocimientos, ni siquiera sobre temas matemáticos que suelen considerarse los conocimientos más precisos. Una de las principales razones de esto es que la evaluación esconde una parte importante del idealismo. Es decir, comprender los factores que representan las metas de la escuela y dan forma al desempeño de los estudiantes (Sanmarti Puig, 2020).

1.1.4 Importancia de una evaluación

Las evaluaciones son importantes ya que permiten recibir una buena retroalimentación o una perspectiva positiva, como un entorno para tomar buenas decisiones u organizarse. La recompensa no es elogiar el trabajo o lo que salió mal, sino proporcionar estrategias y herramientas para ayudar a los estudiantes a encontrar formas de avanzar. La retroalimentación debe ser lo que Vygotsky llamó el entorno de aprendizaje por proximidad. Esto varía de persona a persona. Para obtener buenos resultados, un educador se debe poner en su lugar, comprender su pensamiento, equilibrar lo que se dice y cómo se dice, y cada alumno evaluará las ideas que ha obtenido de un educador, no todas ellas serán idóneas pero se podrá elaborar un plan de acción personal para poner en práctica las más interesantes (Sanmarti Puig, 2020).

Sin embargo, se debe recordar que los consejos de adultos y compañeros no son más que una sugerencia, que cada alumno debe transformar en acciones que será capaz de poner en práctica, ayudando a completar su autoevaluación. Los consejos útiles para el aprendizaje son los que toman los propios estudiantes para interactuar con sus compañeros y adultos, ya que los estudiantes son únicos que pueden solucionar el problema y dar el siguiente paso (Sanmarti Puig, 2020).

1.1.5 Herramientas y tecnologías para la toma de evaluaciones

CEREBRITI es una serie de juegos intelectuales, educativos y tradicionales. Esta empresa de origen español toca todos los palos, todos los ámbitos del conocimiento tanto en la educación como son materias de ciencias, matemáticas, historia, lengua, entre otras, además como del mundo más general como es el cine, la televisión, los deportes, entre otros, a través de preguntas de minijuegos que debemos responder. Cerebriti es sencilla e intuitiva de usar, también hay una sección donde los profesores pueden ver el registro de los estudiantes de un vistazo (Cerebriti, 2020).

KAHOOT! Es un juego de preguntas y respuestas que es obvio y tiene el poder de la simplicidad. El hecho de que contenga tantos paquetes de respuestas llamadas Kahoots y que contenga un color sugerido hace que todos los niños de todas las edades puedan usar Kahoot (Kahoot, 2020).

PLICKERS nació como una herramienta en línea que permite realizar un seguimiento de los estudiantes, encontrar respuestas y calificar en tiempo real. Todos los datos se recopilan en línea y se puede acceder a ellos desde un teléfono inteligente o tableta. El trabajo de un maestro es crear preguntas y respuestas relevantes, entregar tarjetas a los estudiantes y explorar el entorno (*Plickers*, 2019).

MOODLE es una gran herramienta en beneficios de un trabajo emocionante, basada en paquetes y elementos adicionales que agregan nuevas funciones. Y de todas estas existe la posibilidad de poner en marcha evaluaciones en el aula, los cuales se corregirán automáticamente. El maestro solo necesita preparar la prueba, establecer el tiempo para hacer la prueba y dar a los estudiantes instrucciones adecuadas, así entonces obtener todos los resultados en archivos que se podrán visualizar (Moodle.org, 2018).

TRIVINET es una plataforma que brinda la posibilidad de crear cuestionarios especiales de trivia. Pues eso es precisamente lo que permite crear acertijos en línea, compartir preguntas, respuestas y competir con amigos. Una de estas herramientas es una revisión que cubre juegos y entretenimiento para otros fines. Se puede crear un grupo de preguntas favoritas o seguidas por los estudiantes que sean adecuadas para preguntas más generales sobre lo que ha aprendido en un tema o clase en particular (*Trivinet.Com - Juego de Trivial Online Educativo*, 2020).

QUIZZIZZ es una plataforma de juegos que permite a los profesores crear sus propias preguntas en línea y enviarlas a los estudiantes para que las busquen a través de un navegador web y una plataforma fácil de entender. Finalmente, los resultados llegan al profesor para que los estudie. Si no desea hacer una pregunta, puede utilizar otras preguntas publicadas por otros profesores de forma gratuita (Quizizz, 2020).

1.2 Backend y Frontend

1.2.1 Concepto de Backend

El backend de una aplicación web funciona como un habilitador para una mejor experiencia en el frontend. Una aplicación web puede estar diseñada muy bien visualmente, pero si la aplicación en sí no funciona, la aplicación fallará. El backend de una aplicación es

responsable de varias actividades como la por ejemplo la contabilidad, la definición comercial, la interacción de datos y el rendimiento. La mayor parte del código de una aplicación web se lo realiza en el backend para garantizar su funcionamiento. El código backend se ejecuta en el servidor, a diferencia del cliente. Esto significa que los desarrolladores de backend no solo necesitan comprender los lenguajes de programación y las bases de datos, sino que también deben comprender la arquitectura del servidor. Si una aplicación es lenta, falla con frecuencia o causa constantemente errores a los usuarios, podría deberse a un problema de backend (*Web Desing and Development*, 2020).

Sin embargo, el desarrollo de backend no solo está constituido por programación y base de datos. Al igual que el desarrollo de frontend, el desarrollo de backend también consta de un aspecto humano. Debido a que la mayor parte del código de una aplicación está escrito en el backend. La mayoría de los lenguajes de backend, como Ruby y JavaScript, utilizan estilos y expresiones estándar que hacen que la lectura y escritura del código sea más eficiente y agradable (*Web Desing and Development*, 2020).

1.2.2 ¿Qué es API REST?

Una API o interfaz de programación de aplicaciones es un conjunto de instrucciones y protocolos que se utilizan para diseñar e integrar software de aplicación. Generalmente se considera como un contrato entre el informante y el usuario, que define el significado (llamada) requerido por el consumidor y el contenido (respuesta) requerido por el proveedor, es decir, al interactuar con una computadora o sistema para recuperar datos o realizar una tarea, la API se puede usar para enviar los datos relevantes al sistema, para que este comprenda la solicitud y la cumpla (RedHat, 2019).

Una API se puede considerarla como un medio entre los usuarios o clientes y los recursos que quieren obtener. También es una forma para que las empresas compartan recursos y datos mientras mantienen la seguridad y el control, lo que les permite definir oportunidades para cada usuario (RedHat, 2019).

1.2.3 JavaScript y TypeScript

1.2.3.1 JavaScript

JavaScript es un lenguaje de programación ligero, interpretado, o compilado justo a tiempo utilizando funciones de primera clase. Se lo conoce como un lenguaje de scripting (secuencias de comandos) para páginas web, y se utiliza en muchos entornos que no son de navegador, como Node.js, Apache CouchDB y Adobe Acrobat. JavaScript es un lenguaje de

programación basada en prototipos, multiparadigma, de un solo hilo, dinámico, con soporte para programación orientada a objetos, imperativa y declarativa (por ejemplo programación funcional) (*Замыкания - JavaScript | MDN, 2020*).

El estándar para JavaScript es ECMAScript. A partir del 2012, todos los navegadores modernos admiten ECMAScript 5.1. Desde el 17 junio de 2015, ECMA International lanzo oficialmente la sexta versión principal de ECMAScript, que se llama ECMAScript 2015. Originalmente se llamaba ECMAScript 6 o ES6. Desde entonces, las pautas de ECMAScript se han actualizado anualmente por ende esta información hace referencia a la última versión preliminar, que actualmente es ECMAScript 2020 (*Замыкания - JavaScript | MDN, 2020*).

JavaScript no se lo debe confundir con el lenguaje de programación Java, debido a que "Java" y "JavaScript" son marcas o marcas registradas de Oracle en los Estados Unidos y otros países, Sin embargo, los dos lenguajes de programación tienen sintaxis, semántica y usos muy diferentes (*Замыкания - JavaScript | MDN, 2020*).

1.2.3.2 TypeScript

TypeScript es un lenguaje de código abierto basa en JavaScript, una de las herramientas más utilizadas en el mundo, que agrega definiciones de tipo estático. Los tipos explican la funcionalidad de un objeto, proporcionan una mejor documentación y proporcionan una forma para que TypeScript valide que el código funciona correctamente. TypeScript permite que los tipos sean opcionales, ya que la inferencia de tipos le permite obtener mucho poder sin escribir código adicional (*TypeScript: Typed JavaScript at Any Scale., 2020*).

Cualquier código JavaScript válido es código TypeScript. Es posible que obtenga errores en la verificación de los tipos, pero eso no impedirá ejecución de JavaScript. TypeScript se maneja en base a reglas que pueden ir de un comportamiento sencillo a uno más estricto, pero eso no significa que tienes el control sobre el código JavaScript (*TypeScript: Typed JavaScript at Any Scale., 2020*).

El compilador TypeScript o Babel convierte el código TypeScript en código JavaScript limpio y simple, que se ejecuta en cualquier lugar donde se ejecute JavaScript ya sea en un navegador, en node.JS o en aplicaciones (*TypeScript: Typed JavaScript at Any Scale., 2020*).

TypeScript no es una opción binaria. Se puede comenzar aprendiendo JavaScript actual usando JSDoc y, a continuación, modificar algunos de los archivos que escanea TypeScript y con el tiempo, preparar su base de código para convertirlo completamente. Escribir

TypeScript significa que no tiene que definir tipos en el código hasta que esté más seguro (*TypeScript: Typed JavaScript at Any Scale.*, 2020).

1.2.4 Node.Js y Express

1.2.4.1 Node.Js

Node.Js se diseñó como un entorno de ejecución de JavaScript enfocado a eventos asíncronos, y creación de aplicaciones web escalables. En el ejemplo "hola mundo", pueden atenderse muchas conexiones simultáneamente. Las devoluciones de llamadas o callbacks están habilitadas para cada conexión, pero Node.js duerme mientras no hay ningún trabajo de ejecución (OpenJs Foundation, 2020).

Esto se identifica con el modelo de concurrencia más frecuente de hoy en día, en el que se ocupan hilos del Sistema Operativo. Las redes establecidas en hilos son relativamente ineficientes y difíciles de usar. Aparte de eso, los usuarios de Node.js no se preocupan por las fallas que pueden existir en el bloqueo de los procesos, ya que no existen. A menudo ninguna función en Node.js realiza directamente I/O, por lo que el proceso nunca se bloquea. Debido a eso, es muy frecuente desarrollar sistemas escalables en Node.js (OpenJs Foundation, 2020).

Node.js es similar en diseño y está influenciado por sistemas como Event Machine de Ruby y Twisted de Python. Sin embargo, Node.js lleva el modelo de eventos un poco más allá, que incluye un bucle de eventos como runtime de ejecución en lugar de una biblioteca. Algunos sistemas tienen una llamada de bloqueo para disparar un bucle de eventos. El comportamiento generalmente se define mediante devoluciones callbacks de llamada al iniciarse un script y finalmente iniciando un servidor mediante una llamada de bloqueo como `EventMachine::run()`. Node.js, no tiene una llamada de inicio del evento de bucle o `start-the-event-loop`. Node.js simplemente entra en un bucle de eventos después de la ejecución del script de entrada y sale cuando no hay más devoluciones callbacks de llamada para realizar. Funciona similar a JavaScript en el navegador, ya que el bucle de eventos está oculto al usuario (OpenJs Foundation, 2020).

HTTP es un elemento principal en Node.js, que fue diseñado tomando en cuenta la transmisión de operaciones con streaming y baja latencia. Por ende Node.js es muy adecuado para la base de una librería o un framework web (OpenJs Foundation, 2020).

1.2.4.2 Express

Express es una infraestructura de aplicaciones web Node.js fácil de usar que proporciona potentes funciones configuradas para servicios web y aplicaciones móviles. Con miles de utilidades e interfaces HTTP a su alcance, puede crear potentes API de forma rápida y sencilla con Express. Además, proporciona una capa de características de aplicación web básicas, que no ocultan la funcionalidad de Node.js que tanto le gusta y conoce (ExpressJS, 2020).

1.2.5 MongoDB

MongoDB es una base de datos basada en documentos que ofrece modelos de indexación y consultas avanzadas, además de una gran escalabilidad y flexibilidad. El modelo de documentos de MongoDB es fácil de aprender y usar, brindando a los desarrolladores todas las funcionalidades que necesitan para satisfacer sus necesidades más complejas a cualquier escala. MongoDB tiene soporte para más de diez lenguajes, y la comunidad ha desarrollado varias decenas más (*¿Qué Es MongoDB? | MongoDB, 2019*).

Debido a que MongoDB almacena datos en documentos flexibles similares a JSON, por lo que los campos pueden cambiar entre documentos, además que la estructura de datos puede cambiarse con el tiempo. Un factor beneficioso de MongoDB es que su modelo de documento se asigna mediante objetos en el código de una aplicación para facilitar el trabajo con los datos (*¿Qué Es MongoDB? | MongoDB, 2019*).

Las consultas ad hoc, la visualización de datos y la agregación en tiempo real brindan una forma poderosa de acceder y analizar datos, ya que MongoDB es una base de datos distribuida en su núcleo, por ende alta disponibilidad, la escalabilidad horizontal y la distribución geográfica están anexadas para una fácil implementación (*¿Qué Es MongoDB? | MongoDB, 2019*).

MongoDB es de uso gratuito. Las ediciones publicadas antes del 16 de octubre de 2018 tienen la licencia AGPL. Todas las versiones desde el 16 de octubre de 2018, incluidos los parches lanzados para funciones anteriores, se publicaron bajo la Licencia de participación pública (SSPL) v1 (SSPL) v1 (*¿Qué Es MongoDB? | MongoDB, 2019*).

1.2.6 Concepto de Frontend

El frontend es la parte de un proyecto que es visible para el usuario, es decir, se ocupa del lado del cliente. Pase lo que pase en la interfaz de usuario, el usuario puede recibirlo o

administrarlo. Es importante la interfaz de usuario del sitio web y la experiencia del usuario. La forma en que se presenta el sitio web al usuario es el objetivo principal del frontend. Sencillez, facilidad de uso, buena experiencia de usuario, funcionalidad clara e ideas son algunas de las características básicas que juegan un papel importante en la mejora de la creatividad (*Web Desing and Development*, 2020).

HTML es un lenguaje de marcado utilizado para estructurar un sitio web. Este lenguaje de marcado consta de una serie de cosas básicas para crear el frontend de cualquier sitio web. De igual manera existen muchas cosas que aprender, para la creación de una buena interfaz de usuario, y así crear webs mucho más sofisticada (*Web Desing and Development*, 2020).

Los pasos para el desarrollo web frontend, tienen que ver con la programación de una buena interfaz de usuario en una aplicación web. Esto generalmente se refiere al lenguaje de marcado de hipertexto (HTML), las hojas de estilo en cascada (CSS) y la parte de JavaScript como parte del diseño web, a diferencia del hardware y el sistema del servidor. Esto incluye desde la creación de páginas de texto HTML simples hasta sitios web complejos que responden a HTML5, diseñados para ser accedidos a través de múltiples navegadores, dispositivos y tamaños de pantalla (*Web Desing and Development*, 2020).

1.2.7 Concepto de Programación Orientada a Objetos

Es un enfoque basado en componentes, los componentes son similares a una clase. Es decir, un componente es definir y crear objetos. Tanto los componentes como las clases añaden la funcionalidad mediante descripciones de comportamiento llamadas interfaces. Sin embargo, a diferencia de las clases, los componentes pueden implementarse por una sola clase, múltiples clases o procedimientos no orientados a objetos. Por ende, los nombres de los componentes, a diferencia de los nombres de las clases, no se pueden utilizar como tipos nombres. En los componentes, las interfaces son tipos y están separadas de las implementaciones (Vu, 2016).

Elementos de un componente

- Interfaces. Una definición importante sobre el componente se debe, a que tiene una interfaz claramente definida. Se requiere una configuración estándar de interfaz la cual permitir que los componentes de software realicen transacciones, lo cual define lo que puede ser una interfaz. Si un componente contiene la implementación de una interfaz

proporcionada, el componente es compatible con todas implementaciones de operaciones definidas por la interfaz (Vu, 2016).

- Estándar de composición. Un componente necesita encapsular todos los datos y operaciones, y así realizar sus tareas de manera independiente y separada del sistema operativo (Vu, 2016).
- Implementación del modelo de componentes. Se trata de un conjunto de elementos de software ejecutables que son necesarios para permitir la ejecución de componentes en un modelo de componentes (Vu, 2016).

¿Por qué la programación orientada a objetos y a componentes es altamente aceptada y gana en popularidad? Una de las respuestas más importantes a esta pregunta es la "Reutilización del software". Orientado tanto a objetos como a componentes la programación lo cual brinda una ventaja y beneficio mediante la reutilización del software. El uso de la reutilización puede llegar a ser un concepto muy importante en Ingeniería de software, ahorrando tiempo, esfuerzos de los programadores, enfocando a los desarrolladores en la construcción de otras aplicaciones y así ahorrar costos para la empresa en el proceso de desarrollo de software. En lugar de crear una aplicación desde cero, podemos utilizar componentes existentes de software que ya se tengan disponibles e incorporarlos para formar un sistema de software, o una aplicación. Estos componentes de software pueden tener disponibilidad mediante el proveedor de software que se incluye con la descripción de la calidad, funcionalidad y requisitos del sistema de los componentes. El componente de software debe estar bien documentado y tener experiencia en el desarrollo de componentes para garantizar la calidad de los productos de software (Vu, 2016).

1.2.8 React Js

React ayuda a crear interfaces de usuario de manera sencilla e interactivas, mediante el diseño de vistas simples en una aplicación que maneja estados, además react se encarga de actualizar y renderizar de manera eficiente cuando existo u ocurre un cambio en los datos instanciados en un componente de una aplicación web (React, 2019).

Las vistas declarativas ayudan a que el código sea más predecible, por ende, fácil de depurar. Al crear componentes encapsulados que utilicen su propio estado, podemos conviértelos en interfaces de usuario complejas, ya que, la lógica de los componentes está escrita en JavaScript y no en plantillas, por lo que puedes pasar información de forma sencilla mediante una aplicación y mantener el estado fuera del DOM (React, 2019).

React utiliza muchas herramientas tecnológicas, por ende, se puede desarrollar nuevas características sin necesidad de reescribir el código existente. Mediante el uso de react también se puede renderizar desde el servidor usando Node.js, de igual manera potencializar aplicaciones móviles usando React Native (React, 2019).

1.2.9 React Hooks

Los Hooks son una nueva característica en React 16.8. Estos te permiten usar el estado y otras características de React sin necesidad de escribir una clase (React, 2020).

Los Hooks de React son:

- Completamente opcionales. Se Puede utilizar los Hooks en componentes sin reescribir código que ya exista. Debido a eso no es necesario aprender los Hooks, pero no estaría mal aprenderlos para próximos proyectos.
- Son 100% compatibles con versiones anteriores. Los Hooks no afecta el código ya escrito mucho menos ocurren rupturas con respecto a versiones existentes.
- Disponibles de inmediato. Los Hooks ya están disponibles con el lanzamiento de la versión v16.8.0.

Los Hooks no cambia el conocimiento de los conceptos de React. De hecho, los Hooks brindan una API más directa a los conceptos que ya se conoce de React: props, estado, contexto, referencias, y ciclo de vida. Además, los hooks ofrecen una nueva y poderosa forma de combinarlos (React, 2020).

Los Hooks resuelven una variedad de problemas en React que se han encontrado durante más de cinco años de escribir y mantener decenas de miles de componentes, ya sea que se esté aprendiendo React, usándolo o incluso utilizando una librería diferente con un modelo de componentes similar, por ende, es posible que se reconozca algunos de estos problemas (React, 2020).

1.2.10 Redux

Redux fue diseñado como un contenedor predecible del estado de aplicaciones JavaScript realizadas con react, permitiendo que aplicación que se comportan de manera consistente se las pueda escribir de manera más sencilla, además dichas aplicaciones corren en distintos ambientes (cliente, servidor y nativo), y son fáciles de probar. Además de eso,

brinda una gran experiencia de desarrollo, gracias a edición en vivo combinado con un depurador mediante una línea de tiempo (*Read Me · Redux En Español, 2020*).

Redux se puede usar para combinar con React, o cual cualquier otra librería de vistas. Su tamaño es muy pequeño (2kB) y no tiene dependencias. Debido a que los requisitos en aplicaciones JavaScript de una sola página se vuelven cada vez más complejas, afectando nuestro código, más que nunca, por ende, se debe manejar un estado en las aplicaciones web. Este estado incluye respuestas del servidor y datos cacheados, además datos creados localmente que aún no se han guardado en el servidor. El estado de las UI también se volvió más complejo, al necesitar mantener una ruta activa, el tab seleccionado, mostrar o no un spinner, si deben mostrarse los controles de paginación o no (*Read Me · Redux En Español, 2020*).

Controlar un estado cambiante es difícil. Por lo que, si un modelo puede actualizar otro modelo, entonces una vista puede actualizar un modelo, el cual actualiza otro modelo, y esto causa que otra vista se actualice. En cierto punto, es complicado entender que está pasando en la aplicación debido a que siestas veces se puede perder control sobre el cuándo, el por qué y el cómo de un estado. Cuando un sistema es confuso y no intuitivo, es difícil reproducir errores o agregar nuevas características (*Read Me · Redux En Español, 2020*).

Además, hay que considerar que existen nuevos requisitos comunes en el desarrollo frontend. Un desarrollador debe saber manejar actualizaciones, renderizado en el servidor, obtención datos antes de realizar cambios de rutas, y más cosas (*Read Me · Redux En Español, 2020*).

Siguiendo los pasos de Flux, CQRS y Event Sourcing, Redux está trabajando para cambiar la naturaleza de las mutaciones imponiendo restricciones sobre el método y el momento de las actualizaciones. Esta especificación está bajo los tres principios de Redux (*Read Me · Redux En Español, 2020*).

1.3 DevOps y Computación en la nube

1.3.1 Concepto de DevOps

DevOps combina software y servicios de TI para crear equipos multiplataforma que trabajan juntos con herramientas y servicios mejorados. DevOps incluye planificación ágil, programación, integración y entrega continua y seguimiento de aplicaciones. Por ejemplo, Azure DevOps se basa en la filosofía DevOps. Este producto está integrado con Github para

brindar una experiencia completa al equipo de desarrollo (*How DevOps Improves the 4 Phases of an Application Lifecycle | Inova Solutions, 2020*).

DevOps tiene como objetivo involucrar a todo el proceso de desarrollo en cada fase. Cada parte depende una de la otra y no hay un puesto específico en ese nivel. En una verdadera cultura de DevOps, cada función va a cada sección durante un período de tiempo (*How DevOps Improves the 4 Phases of an Application Lifecycle | Inova Solutions, 2020*).

Comencemos por definir el rol de DevOps y la filosofía de Azure DevOps en el ciclo de vida de la aplicación. Por lo general, consta de cuatro etapas: planificación, desarrollo, entrega y operación (*How DevOps Improves the 4 Phases of an Application Lifecycle | Inova Solutions, 2020*).

- Planificar: Agilidad y visibilidad

En esta etapa, las organizaciones describen claramente las aplicaciones y los sistemas que crean. Se describen funciones y características. Los equipos pueden aprovechar la agilidad y la visibilidad, que, en esta etapa, mediante la creación de trabajos pendientes, el seguimiento de errores en este nivel navegando, usando Kanban y viendo el progreso en paneles. La herramienta más útil en este nivel entre Azure DevOps es Azure Boards, que informa. Además, puede conectar Power BI para insertar análisis y convertir sus datos en conocimientos sólidos (*How DevOps Improves the 4 Phases of an Application Lifecycle | Inova Solutions, 2020*).

- Desarrollar: Automatización

El desarrollo automatizado incluye la creación, prueba, revisión e integración de entornos de recopilación y archivo de datos. Los equipos que implementan Azure DevOps pueden aprovechar el uso de Visual Studio y Visual Studio Code para una codificación más rápida. Azure Pipelines automatiza las pruebas permitiendo la integración continua en la nube, además los entornos pueden aprovechar Azure DevTest Labs (*How DevOps Improves the 4 Phases of an Application Lifecycle | Inova Solutions, 2020*).

- Entregar: Fácilmente repetible

Al enviar su solicitud en producción, su equipo debe aprobarla. Al implementar la automatización durante el proceso de entrega de mensajes, puede hacerlos más consistentes, redistribuidos y manejables. Durante la entrega, los siguientes pasos pueden

ser útiles: canalización de Azure, servicios de GitHub, productos de Azure, instalación del paquete de GitHub los cuales nos ayudan a realizar una mejor integración continua sin necesidad de una persona en el deploy de una aplicación o actualización de cambios (*How DevOps Improves the 4 Phases of an Application Lifecycle | Inova Solutions, 2020*).

- Operar: Supervisión y seguridad

Las computadoras pueden usar Azure DevOps para acceder a la increíble seguridad que brindan los servicios en la nube de Microsoft. ¿Por qué es importante en el entorno laboral? Este nivel incluye el mantenimiento, la supervisión y la resolución de problemas de la aplicación en un entorno de producción. El equipo de desarrollo creó el objetivo de confiabilidad, sabiduría y lucha contra la corrupción sistemáticas al tiempo que mejoraba la seguridad y la gobernanza. Cuando surge un problema, el equipo quiere tener la oportunidad de identificar el problema antes de que afecte la experiencia del cliente y reducirlo rápidamente cuando ocurra. Azure DevOps permite que su organización realice una configuración completa, reciba alertas activas y acceda a registros y anuncios en Azure Monitor. Además, otras herramientas como Azure Automation, Azure Blueprints y ChefAutomate permiten el monitoreo en la nube y ayudan a las organizaciones a cumplir con las regulaciones. Utilizado junto con esta tecnología, Azure Security Center ayuda a limitar la exposición a amenazas y corregir vulnerabilidades (*How DevOps Improves the 4 Phases of an Application Lifecycle | Inova Solutions, 2020*).

1.3.2 Que es un gestor de versiones

Un control de versiones es un sistema de administración que registra los cambios realizados en un archivo o conjunto de archivos a lo largo del tiempo, para poder restaurar una versión en particular más adelante. Se pueden usar archivos de código fuente para controlar una determinada versión de una aplicación, pero en realidad se puede hacer lo mismo con casi cualquier tipo de archivo que encuentre en una computadora (git, 2018).

Un diseñador gráfico o diseñador web y puede conservar todo tipo de imágenes o diseños, utilizando el enfoque de un sistema de control de versiones (VCS) es una decisión acertada. Mediante este tipo de sistemas, se permite regresar a versiones anteriores de archivos, regresar a una versión anterior del proyecto completo, comparar cambios a lo largo del tiempo, ver quién modificó por última vez algo que pueda estar causando problemas, ver quién introdujo un problema y cuándo, entre otras cosas. Usar un VCS generalmente facilita la recuperación de archivos si están dañados o en caso de ser eliminados. Además, se puede

disfrutar de todos estos beneficios a un bajo coste de recursos computacionales o directamente en la nube con GitHub (git, 2018).

1.3.3 Git

Git consta de una serie de fundamentos para comprender como funciona, por ende, usar Git con eficacia probablemente será mucho más fácil de mantener y leer un código a través de los cambios entre varias personas a la vez. A medida que se comprende Git, se puede despejar la mente de las cosas que se puede saber sobre otros VCS, por ejemplo, CVS, Subversion o Perforce, así que, por ende, hacerlo te ayudará a evitar confusiones sutiles al usar la herramienta. La interfaz de usuario de Git es similar a otros VCS, pero Git almacena y recupera datos de una manera diferente. Comprender estas diferencias lo ayudará a evitar confusiones al usarlas (Git Inc, 2020).

1.3.4 GitHub

GitHub es una plataforma de recuperación de código para administrar versiones colaborativas. Esto le brinda múltiples oportunidades para colaborar en un proyecto desde cualquier lugar, diseñado para la colaboración. Permite que una computadora acceda a funciones adicionales y mejorar el rendimiento de una computadora (GitHub, 2020).

GitHub integra la revisión de código, mediante las solicitudes de extracción las cuales cubren todo el flujo de revisión: proponer cambios, examinar el código, pedir entrada, hacer una sugerencia y cerrar sesión en un solo lugar. Además, sigue trabajando de manera continua, lo cual permite revisar o combinar código, administrar notificaciones, examinar repositorios y mucho más con GitHub ya sea mediante dispositivos móviles, computadoras o cualquier dispositivo electrónico que tenga una conexión a internet. Se puede colocar una GUI con GitHub Desktop o permanecer en la línea mediante comandos con la CLI de GitHub. Disponible para macOS, Windows y Linux (GitHub, 2020).

1.3.5 Docker

Docker elimina la necesidad de servicios de configuración estándar y se utiliza durante todo el proceso de desarrollo, lo cual permite un desarrollo de aplicaciones más rápido, fácil y escalable para escritorio y nube. La plataforma de un extremo a otro de Docker incluye una interfaz de usuario, CLI, API y seguridad diseñadas para trabajar juntas en el ciclo de entrega de la aplicación (Hykes, 2016).

Ventajas de Docker

- La codificación es más sencilla aprovechando las imágenes de Docker para desarrollar de forma eficiente sus propias aplicaciones únicas en Windows y Mac (Hykes, 2016).
- Integrar herramientas a lo largo de la canalización de desarrollo: Docker se integra con todas las herramientas de desarrollo, incluidos VS Code, CircleCI y GitHub (Hykes, 2016).
- Empaqueta aplicaciones como imágenes de contenedor portátiles para ejecutar en cualquier entorno de forma coherente, utilizando Kubernetes locales hasta AWS ECS, Azure ACI, Google GKS y mucho más (Hykes, 2016).
- Se puede personalizar el acceso para desarrolladores a imágenes con la capacidad de usar registros de búsqueda de Docker Hub para acceder a su espacio de trabajo y conocer su historia de trabajo (Hykes, 2016).
- Proporciona múltiples aplicaciones sin problemas y ejecuta la misma aplicación en todos los entornos, incluido el diseño, las pruebas, la autorización y la producción (Hykes, 2016).
- Implemente las aplicaciones en contenedores independientes de forma independiente y en diferentes idiomas. Reducir el riesgo de conflicto entre idiomas, bibliotecas o marcos (Hykes, 2016).
- Acelere el desarrollo con la simplicidad de la CLI de Docker Compose y, con un solo comando, lance sus aplicaciones localmente y en la nube con AWS ECS, Azure ACI y Google GKS (Hykes, 2016).

1.3.6 Concepto de computación en la nube

En pocas palabras, la computación en la nube proporciona servicios informáticos como servidores, almacenamiento, bases de datos, comunicaciones, software, análisis e inteligencia a través de Internet para un crecimiento económico más rápido, más fácil y eficiente. Por lo general, paga por los servicios en la nube que utiliza. Esto ayuda a reducir los costos laborales, mejorar la gestión de la infraestructura y adaptarse a las necesidades de su negocio (Microsoft, 2016).

La computación en la nube es una desviación importante de la forma tradicional en que las empresas piensan sobre los recursos de TI. Aquí hay siete razones comunes por las que las organizaciones recurren a los servicios de computación en la nube:

- La computación en la nube elimina gastos en el requerimiento de hardware, software, configuración, ejecución de centros de datos, los bastidores de servidores, la electricidad las 24 horas del día para la alimentación, la refrigeración, expertos en TI para administrar la infraestructura (Microsoft, 2016).
- La mayoría de los servicios de computación en la nube se proporcionan independientes y bajo demanda, por ende, puede proporcionar una gran cantidad de herramientas de computación en minutos al alcance de algunos clics, por lo que brinda a una empresa mucha flexibilidad y quitando la presión de la planificación de la capacidad (Microsoft, 2016).
- Las ventajas de los servicios en la nube incluyen la capacidad de actualizarse fácilmente, provisión de una gran cantidad de recursos de TI, potencia de cómputo, almacenamiento y ancho de banda, cuando sea necesario y desde cualquier ubicación las 24 horas del día (Microsoft, 2016).
- Los centros de datos generalmente requieren una configuración de hardware extensa, parches de software y otras tareas de administración de TI que requieren mucho tiempo. La computación en la nube elimina muchas de estas tareas, lo que permite que las organizaciones de TI dediquen más tiempo a lograr objetivos comerciales críticos (Microsoft, 2016).
- El servicio líder de administración de computadoras opera en una red confiable de centros de datos y se actualiza constantemente a una nueva generación de hardware de rápido movimiento. Esto tiene varias ventajas, incluida la reducción de la latencia de la red para las aplicaciones y la economía en un solo centro de datos (Microsoft, 2016).
- La computación en la nube facilita, recuperaciones ante desastres y continuidad del negocio, ya que los datos se pueden reflejar en varios sitios redundantes de la red del proveedor de nube (Microsoft, 2016).
- Muchos proveedores de nube ofrecen múltiples políticas, tecnologías y controles que mejoran su sistema de seguridad en su conjunto y ayudan a proteger sus datos, aplicaciones e infraestructura de posibles amenazas (Microsoft, 2016).

1.3.7 Que es DigitalOcean

Simplifique su trabajo en la nube y escale con actualizaciones con grupos de administración cuidadosos, precios predecibles, costos de equipos y más (Digital Ocean, 2020).

Implementar y escalar sin problemas

Funciona como un sistema de configuración eficiente permite a los equipos ahorrar tiempo trabajando e implementando aplicaciones distribuidas, experiencia técnica y de ingeniería, proyectos estructurados, redes de clientes o entornos comerciales (Digital Ocean, 2020).

Plataforma fiable

Más de 100,000 desarrolladores en todo el mundo han aprobado DigitalOcean, respaldando sus negocios con un SLA 99.99% más alto para todos los servicios (Digital Ocean, 2020).

Precios predecibles

Cuenta con una estructura de precios accesibles. Al conocer lo que siempre se paga al mes con una estructura de precios plana y líder en la industria (Digital Ocean, 2020).

Servicio de clase mundial

Soporte técnico gratuito las 24 horas del día para todos los clientes, con beneficios adicionales para los suscriptores de soporte premium. Sentirás el amor (Digital Ocean, 2020).

Seguridad y cumplimiento

Mantiene los datos protegidos. Nuestra plataforma está certificada en múltiples estándares internacionales, incluyendo SOC 1 Tipo II, SOC 2 Tipo II, ISO/IEC 27001 y PCI-DSS (Digital Ocean, 2020).

1.4 Metodología XP y Norma ISO 25010 enfocada a la usabilidad

1.4.1 Metodología de desarrollo XP

La metodología de desarrollo XP o (Metodología Extreme Programming) es un modelo de desarrollo de software que se centra en mejorar la calidad del software y su capacidad

para satisfacer las necesidades cambiantes de los clientes. Este es un tipo de modelo de desarrollo de software ágil. Admite lanzamientos frecuentes en ciclos de desarrollo cortos. Estas versiones tienen como objetivo mejorar la productividad y la calidad del software mediante la introducción de medios específicos para centrarse en el lanzamiento de puntos de prueba específicos que se pueden pasar y cumplir con los requisitos del cliente (Yasvi, 2019).

La programación extrema se centra en los procesos ligeros. Las principales etapas involucradas en el ciclo de XP son la planificación, el diseño, la codificación y las pruebas. Dado que es un modelo iterativo, el sistema se desarrolla dividiendo un proyecto común en funciones más pequeñas. El ciclo de desarrollo desde el diseño hasta la prueba es para la funcionalidad. Después de ejecutar una funcionalidad y depurar correctamente, el desarrollador pasa a la siguiente funcionalidad (Yasvi, 2019).

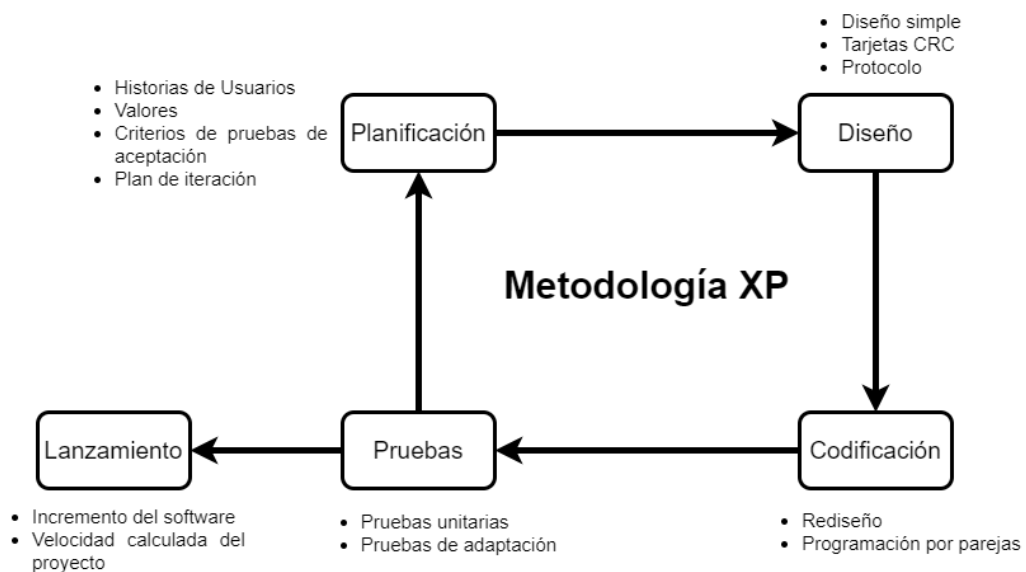


Figura 9 metodología Extreme Programming

1.4.1.1 Planeación

La metodología XP cataloga a la planificación como un diálogo continuo entre las partes interesadas del proyecto, incluidos los clientes, los desarrolladores y los coordinadores. El proyecto comienza recopilando historias de usuarios y así dar paso a los casos de uso. Una vez que se recopilan estas historias de usuario, los desarrolladores evalúan rápidamente el tiempo de desarrollo de cada una de estas historias (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).

Los Conceptos básicos de la planificación son:

- Historias de Usuario: son descritas por el cliente, en su propio idioma con una breve explicación de lo que se supone que debe hacer el sistema (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).
- Plan de entregas: establece que las historias de usuario deben agruparse para conformar entregas y garantizar el orden de estas. El cronograma será el resultado de reuniones entre las partes interesadas del proyecto (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).
- Plan de iteraciones: las historias de usuarios seleccionadas en un orden de entrega y probadas en un ciclo de iteración, de acuerdo con el cronograma establecido (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).

1.4.1.2 Diseño

La Metodología XP hace especial énfasis en los diseños simples y claros. Los conceptos más importantes de diseño en esta metodología son los siguientes:

- Simplicidad: los proyectos simples se completan más rápido que los proyectos complejos. Es por eso por lo que XP sugiere hacer el diseño más simple que funcionen (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).
- Soluciones: si encuentra un problema técnico, o si es difícil estimar el tiempo requerido para ejecutar su historia de usuario, puede usar un pequeño programa de prueba para encontrar diferentes soluciones (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).
- Recodificación: se trata de reescribir partes del código del programa para hacerlo más simple, conciso y fácil de entender, sin cambiar la funcionalidad del programa. Los métodos de XP recomiendan la recodificación según sea necesario (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).
- Metáforas: son una forma sencilla de explicar el propósito de un proyecto y orientar su estructura. Una buena metáfora debe ser comprensible para el cliente y tener suficiente contenido para orientar la arquitectura del proyecto (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).

1.4.1.3 Codificación

Disponibilidad del cliente, uno de los requisitos de XP es que el cliente esté disponible durante todo el proyecto. No solo para apoyar a los desarrolladores, sino también para ser miembro del equipo. La participación del cliente es fundamental para desarrollar un proyecto utilizando la metodología XP. Al comienzo del proyecto, debe proporcionar las historias de usuario. Sin embargo, estas historias son obviamente breves y de "alto nivel" y no contienen los detalles necesarios para el desarrollo del código. Estos detalles los proporciona el cliente durante la fase de desarrollo y deben discutirse con el desarrollador (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).

Uso de Estándares, XP facilita la programación basada en estándares de una manera que es fácil de entender y codificar para los equipos (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).

Programación Dirigida por las Pruebas, la metodología XP propone un modelo donde primero escribe las pruebas que el sistema debe pasar. En segundo lugar, el desarrollo debe hacer lo mínimo indispensable para pasar las pruebas definidas anteriormente (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).

Integraciones Permanentes, todos los desarrolladores deben trabajar siempre con "la última versión". Hacer cambios o mejoras a versiones anteriores puede causar serios problemas y retrasar el proyecto. Es por eso que XP recomienda lanzar una nueva versión lo antes posible, incluso si no está actualizada, a menos que sea defectuosa. Idealmente, debería lanzarse una nueva versión todos los días. Solo algunos desarrolladores pueden incrustar el código al mismo tiempo para evitar errores (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).

Propiedad Colectiva del código, un proyecto basado en la metodología XP, permite que todo el equipo presente nuevas ideas que se apliquen a cualquier parte del proyecto. Además, algunos programadores pueden cambiar el código según sea necesario para solucionar problemas, agregar funcionalidad o volver a codificar (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).

Ritmo Sostenido, la metodología XP indica que se necesita trabajar a un ritmo constante. El concepto que se debería establecer en esta práctica es planificar su trabajo para mantener un ritmo constante y razonable sin sobrecargar a su equipo (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).

1.4.1.4 Pruebas

Pruebas Unitarias, todos los módulos deben aprobar las pruebas unitarias antes de que puedan ser liberadas o publicadas. Por otro lado, como se mencionó anteriormente, debe definir una prueba antes de poder ejecutar su código. El hecho de que todo el código publicado pase las pruebas unitarias es lo que hace que la copropiedad del código funcione (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).

Detección y Corrección de Errores, si se encuentra un error, se debe solucionar de inmediato y se debe tener cuidado para evitar que se repitan errores similares. Del mismo modo, se creará una nueva prueba para confirmar que el error se ha corregido (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).

Pruebas de Aceptación, estas se crean en función de las historias de los usuarios en cada ciclo iterativo de desarrollo. Los clientes deben especificar uno o más escenarios para asegurarse de que la historia del usuario se implemente correctamente. Del mismo modo, si fallan varias comprobaciones, deben indicar la prioridad de resolución. Una historia de usuario no puede considerarse completa hasta que haya pasado todas las pruebas de aceptación (Meléndez Valladares & Pérez Reyes, Neldin Noel Elizabeth Gaitan, 2016).

1.4.2 Norma ISO 25010

El modelo de calidad constituye la base para el establecimiento de un sistema de evaluación de la calidad del producto. Este modelo define las características de calidad que se deben tener en cuenta al evaluar los atributos de un producto de software en particular (ISO 25010, n.d.)

La calidad de un producto de software puede entenderse como el grado en que el producto cumple con los requisitos del usuario y proporciona valor. Son estos requisitos (funcionalidad, rendimiento, seguridad, mantenibilidad, etc.) (ISO 25010, 2018).



Figura 10 Norma ISO 25010. Fuente: (ISO 25010, 2018.)

1.4.2.1 Usabilidad

La capacidad de un producto de software para comprender, aprender, utilizar y atraer usuarios cuando se utiliza en determinadas condiciones. Esta función se puede dividir en las siguientes subcaracterísticas:

- Capacidad para reconocer su adecuación: las características del producto permiten a los usuarios comprender si el software satisface sus necesidades (ISO 25010, 2018).
- Capacidad de aprendizaje: las características del producto permiten a los usuarios aprender la aplicación (ISO 25010, 2018).
- Capacidad para ser usado: las funciones del producto deben ser fáciles de operar y controlar para los usuarios (ISO 25010, 2018).
- Protección contra errores de usuario: las funciones del sistema deben proteger a los usuarios de errores (ISO 25010, 2018).
- Estética de la interfaz de usuario: las características de la interfaz de usuario responden y satisfacen la interacción del usuario (ISO 25010, 2018).

CAPITULO II

DESARROLLO DE LA APLICACIÓN WEB

2.1 Planificación

El desarrollo de sistemas web debe tomar en cuenta la necesidad de aplicaciones intuitivas y fáciles de usar. La aplicación se divide en varios bloques y las partes como se presenta en la figura 11.

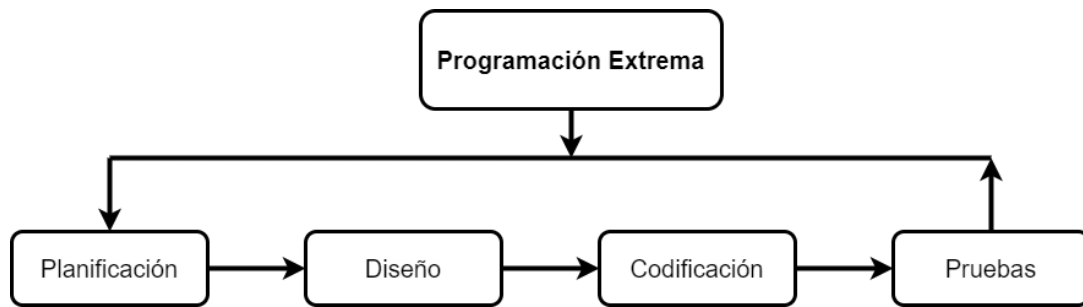


Figura 11 Programación Extrema

En esta etapa se realiza un análisis preliminar del problema a resolver y se conforma un grupo de trabajo que forma parte del proceso de desarrollo de la aplicación en todas las fases. Mediante el uso de la metodología XP, se puede realizar cambios durante el desarrollo para modificar las historias de usuario mediante el desarrollo de la aplicación.

Tabla 2

Participantes de la metodología.

Miembros	Grupos	Roles
Erik Morocho	Tesista	Programador, Tester.
Ing. Guillermo Pérez	Cliente	Owner
Ing. Cosme MacArthur	Consultor	Traker

2.1.1 Historias de usuario

Las Historias de Usuario han sido descritas en un lenguaje común, para que puedan ser entendidas por todos los miembros del grupo de trabajo, representando los requerimientos con los que debe cumplir el sistema como se representan en las siguientes tablas descritas a continuación.

Tabla 3

Historia de usuario diseño de la base de datos

Historia de usuario	
Numero: 1	Usuario: Desarrollador
Nombre Historia: Diseño de la base de datos	
Prioridad: Alto	Riesgo: Alto
Esfuerzo: Medio	Interacción: 1
Programador responsable: Erik Morocho	

Descripción:

Diseño y elaboración de la base de datos la cual se utilizará durante la construcción de aplicaciones web.

Observaciones:**Tabla 4**

Historia de usuario desarrollo del backend

Historia de usuario	
Numero: 2	Usuario: Desarrollador
Nombre Historia: Desarrollo del backend	
Prioridad: Alto	Riesgo: Alto
Esfuerzo: Medio	Interacción: 1
Programador responsable: Erik Morocho	
Descripción:	
Desarrollo y construcción de las siguientes Apis para poder realizar las siguientes funciones:	
<ul style="list-style-type: none"> • Autenticación de usuarios • CRUD de usuarios • CRUD de categorías • CRUD de cursos • CRUD de módulos • CRUD de contenido • CRUD de evaluaciones • CRUD de preguntas • CRUD de respuestas • Control y gestión del carrito de compras • CRUD de pagos 	
Observaciones: La palabra CRUD es una forma de abreviar 4 palabras con sus iniciales las cuales hacen referencia a: Create, Read, Update, Delete	

Tabla 5

Historia de usuario diseño y construcción de las vistas de inicio

Historia de usuario

Numero: 3

Usuario: Desarrollador

Nombre Historia: Diseño y construcción de las vistas de inicio.

Prioridad: Alto

Riesgo: Alto

Esfuerzo: Medio

Interacción: 1

Programador responsable: Erik Morocho

Descripción:

Diseño y construcción de la pantalla de inicio de la aplicación que deberá estar constituida por los siguientes componentes:

- Inicio
- ¿Quiénes somos?
- ¿Nuestros servicios?
- Contactarnos
- Cursos disponibles
- Iniciar sesión
- Registrarse

Observaciones:

Tabla 6

Historia de usuario diseño y construcción de la vista para autenticación de usuarios

Historia de usuario

Numero: 4

Usuario: Desarrollador

Nombre Historia: Diseño y construcción de la vista para autenticación de usuarios

Prioridad: Alto

Riesgo: Alto

Esfuerzo: Medio

Interacción: 1

Programador responsable: Erik Morocho

Descripción:

Diseño y construcción de las pantallas para autenticación de usuarios que deberá contener los componentes:

- Formulario para iniciar sección
- Formulario para registro de usuarios
- Recuperar contraseña

- Autenticación mediante Google o Facebook.

Observaciones:

Tabla 7

Historia de usuario diseño y construcción de la vista de inicio del usuario

Historia de usuario	
Numero: 5	Usuario: Desarrollador
Nombre Historia: Diseño y construcción de la vista de inicio del usuario	
Prioridad: Alto	Riesgo: Alto
Esfuerzo: Medio	Interacción: 1
Programador responsable: Erik Morocho	
Descripción:	

Diseño y construcción de la pantalla de inicio de un usuario. Los componentes que podrán visualizarse son:

- Inicio
- Categorías
- Cursos
- Mis cursos
- Historial de compras
- Carrito de compras
- Perfil de usuario
- Cerrar sección

Observaciones:

Tabla 8

Historia de usuario diseño y construcción de la vista del perfil de usuario

Historia de usuario	
Numero: 6	Usuario: Desarrollador
Nombre Historia: Diseño y construcción de la vista del perfil de usuario	
Prioridad: Alto	Riesgo: Alto

Esfuerzo: Medio

Interacción: 1

Programador responsable: Erik Morocho

Descripción:

Diseño y construcción de la pantalla del perfil de usuario para visualizar, actualizar información básica o cambiar su contraseña.

Observaciones:

Tabla 9

Historia de usuario diseño y construcción de un dashboard para visualizar un curso

Historia de usuario	
Numero: 7	Usuario: Desarrollador
Nombre Historia: Diseño y construcción del dashboard para visualizar un curso.	
Prioridad: Alto	Riesgo: Alto
Esfuerzo: Alto	Interacción: 1
Programador responsable: Erik Morocho	
Descripción:	
Diseño y construcción del dashboard para visualizar un curso el cual deberá constar de los siguientes componentes visuales:	
<ul style="list-style-type: none">• Reproducción de los videos del curso• Módulos del curso• Contenidos de los módulos del curso• Evaluaciones del curso• Certificado del curso• Forma de calificar el curso• Progreso del curso	
Observaciones:	

Tabla 10

Historia de usuario diseño y construcción del dashboard para el usuario instructor

Historia de usuario

Numero: 8

Usuario: Desarrollador

Nombre Historia: Diseño y construcción del dashboard para el usuario instructor

Prioridad: Alto

Riesgo: Alto

Esfuerzo: Alto

Interacción: 1

Programador responsable: Erik Morocho

Descripción:

Diseño y construcción de un dashboard para el usuario instructor el cual podrá visualizar los componentes de:

- Lista de los cursos asignados
- Edición de los cursos asignados
- Previsualización de los cursos asignados
- CRUD de los módulos del curso
- CRUD de los contenidos de los módulos
- CRUD de las evaluaciones del curso
- CRUD de las preguntas de las evaluaciones
- CRUD de las respuestas de las preguntas

Observaciones:

Tabla 11

Historia de usuario diseño y construcción del componente visual de los cursos

Historia de usuario

Numero: 9

Usuario: Desarrollador

Nombre Historia: Diseño y construcción del componente visual de cursos

Prioridad: Alto

Riesgo: Alto

Esfuerzo: Alto

Interacción: 1

Programador responsable: Erik Morocho

Descripción:

Diseño y construcción del componente visual de los cursos los cuales estarán constituido por los siguientes componentes:

- Información del curso
- Módulos del curso

- Contenidos del módulo de un curso

Observaciones:

Tabla 12

Historia de usuario diseño y construcción del dashboard del administrador

Historia de usuario	
Numero: 10	Usuario: Desarrollador
Nombre Historia: Diseño y construcción del dashboard del administrador	
Prioridad: Alto	Riesgo: Alto
Esfuerzo: Medio	Interacción: 1
Programador responsable: Erik Morocho	

Descripción:

Diseño y construcción de un dashboard para el usuario administrador el cual deberá contener los siguientes componentes visuales:

- Inicio con información relevante
- Lista de clientes de la aplicación web
- CRUD de las categorías para los cursos
- CRUD de cuentas para administradores e instructores
- CRUD de los cursos para asignar a los instructores
- Lista y forma de impresión de las ventas

Observaciones:

Tabla 13

Historia de usuario diseño y construcción del componente visual carrito de compras

Historia de usuario	
Numero: 11	Usuario: Desarrollador
Nombre Historia: Diseño y construcción del componente visual carrito de compras	
Prioridad: Alto	Riesgo: Alto
Esfuerzo: Medio	Interacción: 1

Programador responsable: Erik Morocho

Descripción:

Diseño y construcción del componente visual para el carrito de compras de un usuario el cual deberá constar con la siguiente información:

- Información de los productos a comprar
- Forma de eliminar un producto
- Sumatoria de los precios de los productos (Total a pagar)

Observaciones:

Tabla 14

Historia de usuario diseño y construcción del componente visual para realizar pagos

Historia de usuario	
Numero: 12	Usuario: Desarrollador
Nombre Historia: Diseño y construcción del componente visual para realizar pagos	
Prioridad: Alto	Riesgo: Alto
Esfuerzo: Medio	Interacción: 1
Programador responsable: Erik Morocho	
Descripción:	
Diseño y construcción del componente visual para realizar formas de pago este componente debe constar con:	
<ul style="list-style-type: none">• Información sobre los productos que se va a pagar• Forma de pago mediante PayPal	
<hr/> Observaciones: <hr/>	

2.2 Diseño

Mediante diagramas de flujo se representará el desarrollo de la construcción del software.

2.2.1 Arquitectura de la aplicación web

A continuación, se detalla el diseño, estructura y componentes tecnológicos que formaran parte de la aplicación web.

En la figura 12 se puede visualizar la estructura de la aplicación web la cual esta conformada por el servidor realizado con node.js, express, MongoDB y Typescript, un cliente conformado por react y redux, además de una forma de despliegue con Docker y Nginx en un servidor en la nube con sistema operativo Ubuntu, utilizando GitHub como repositorio de archivos de código fuente.

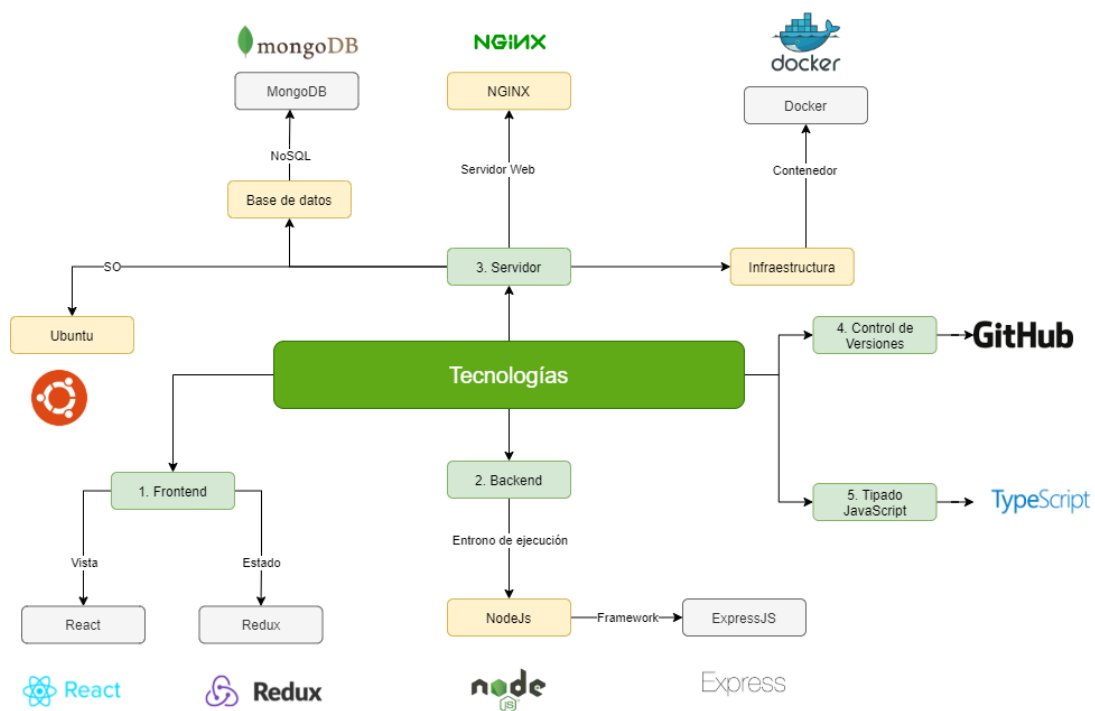


Figura 12 Arquitectura de la aplicación web

2.2.2 Diagrama de procesos

A continuación, se presentan los diagramas de subprocessos y procesos que serán utilizados en los módulos de E-learning y evaluaciones que serán automatizados en la aplicación web.

2.2.2.1 Diagramas de subprocessos

Diagrama del subprocesso de autenticación: Aquí se representa una serie de pasos que conlleva un correcto inicio de sesión.

En la figura 13 se muestra como es la validación del inicio de sesión de un usuario el cual debe proveer sus credenciales de acceso y dependiendo si es usuario esta registrado tendrá acceso al sistema, de lo contrario deberá crear una cuenta.

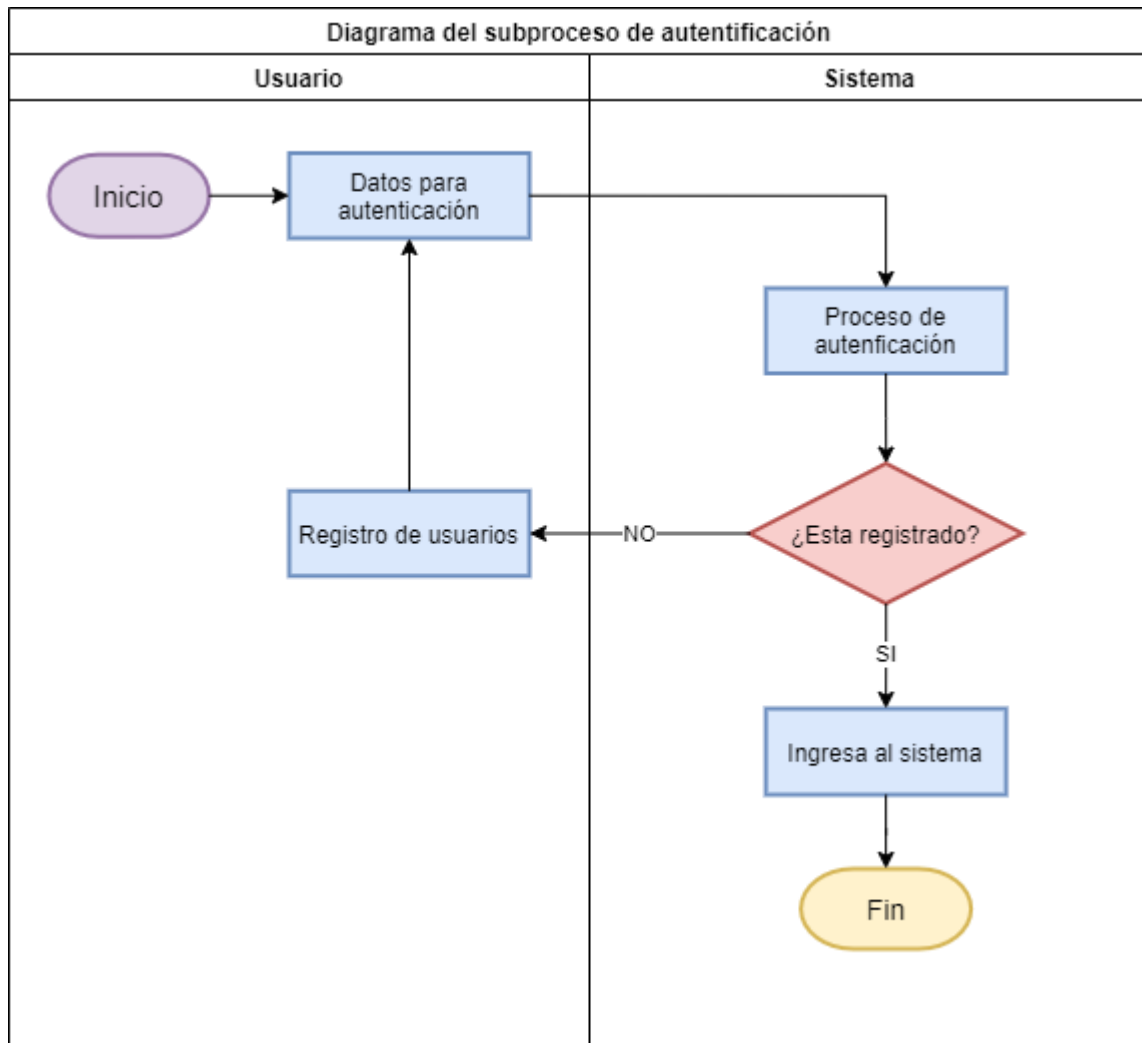


Figura 13 Diagrama del subproceso de inicio de sesión

Diagrama del subproceso de pago de cursos: Aquí se detalla cómo debe realizarse un subproceso de pago para la compra de un producto.

En la figura 14 se muestra cómo se debe realizar el proceso de pago el cual debe iniciar mediante la selección del método de pago, para la verificación de los datos de pago y una vez que se validen que son correctos, se realiza el pago y se puede visualizar la compra realizada, pero en caso de fallar el pago el usuario deberá escoger otro método de pago ya que sus datos de pago son incorrectos.

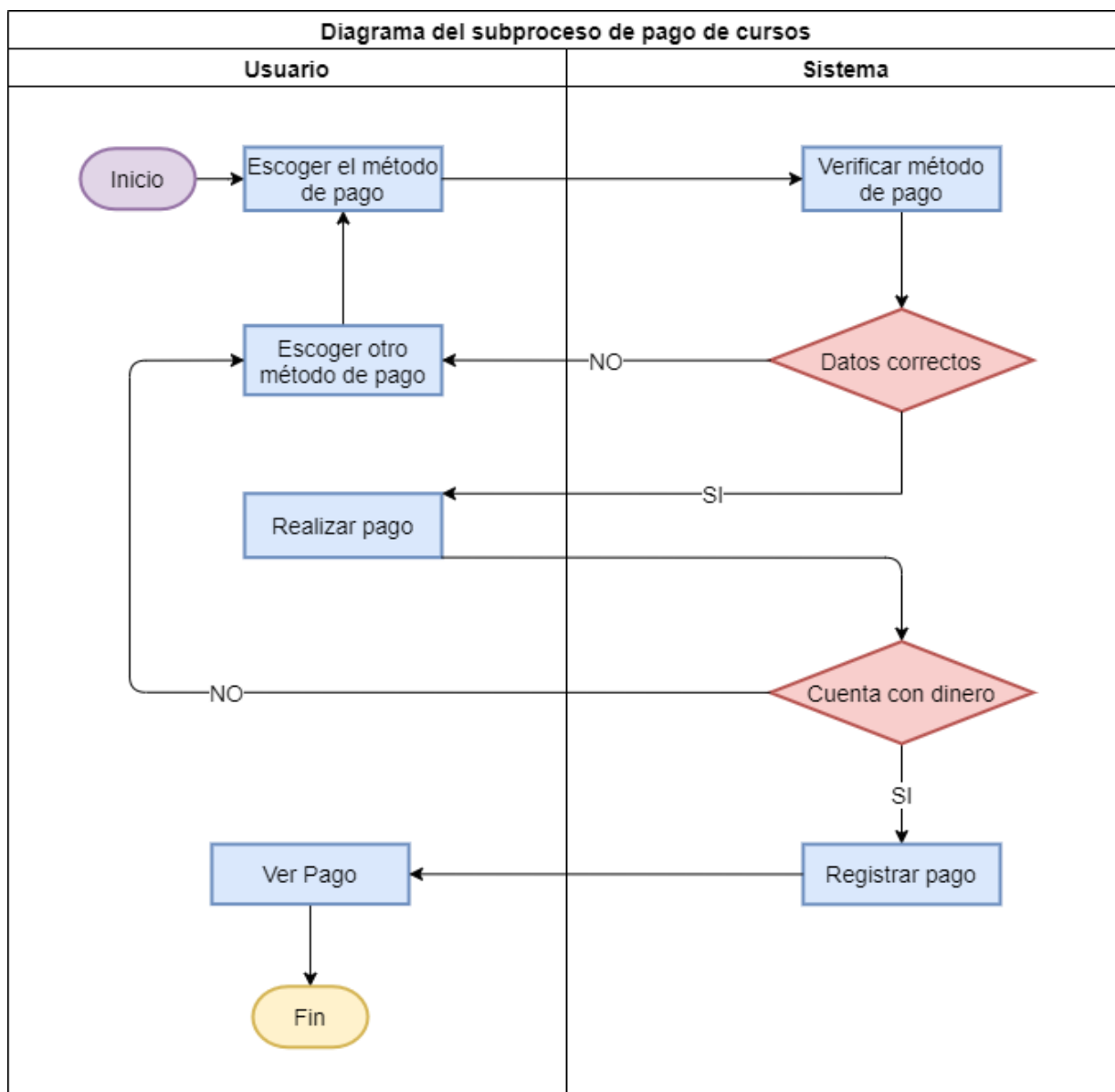


Figura 14 Diagrama del subproceso de pago de cursos

Diagrama del subproceso de categorías: Se realiza una serie de pasos para poder realizar un CRUD de categorías.

En la figura 15 se representa los métodos que se podrían realizar en la sección de categorías los cuales son: crear, actualizar, eliminar y visualizar datos, cada uno de estos métodos tiene sus fases secuenciales, pero la de eliminación debe ser controlada con mayor prioridad, ya que se tiene que preguntar al usuario si desea eliminar una categoría, de lo contrario no se eliminara ninguna categoría.

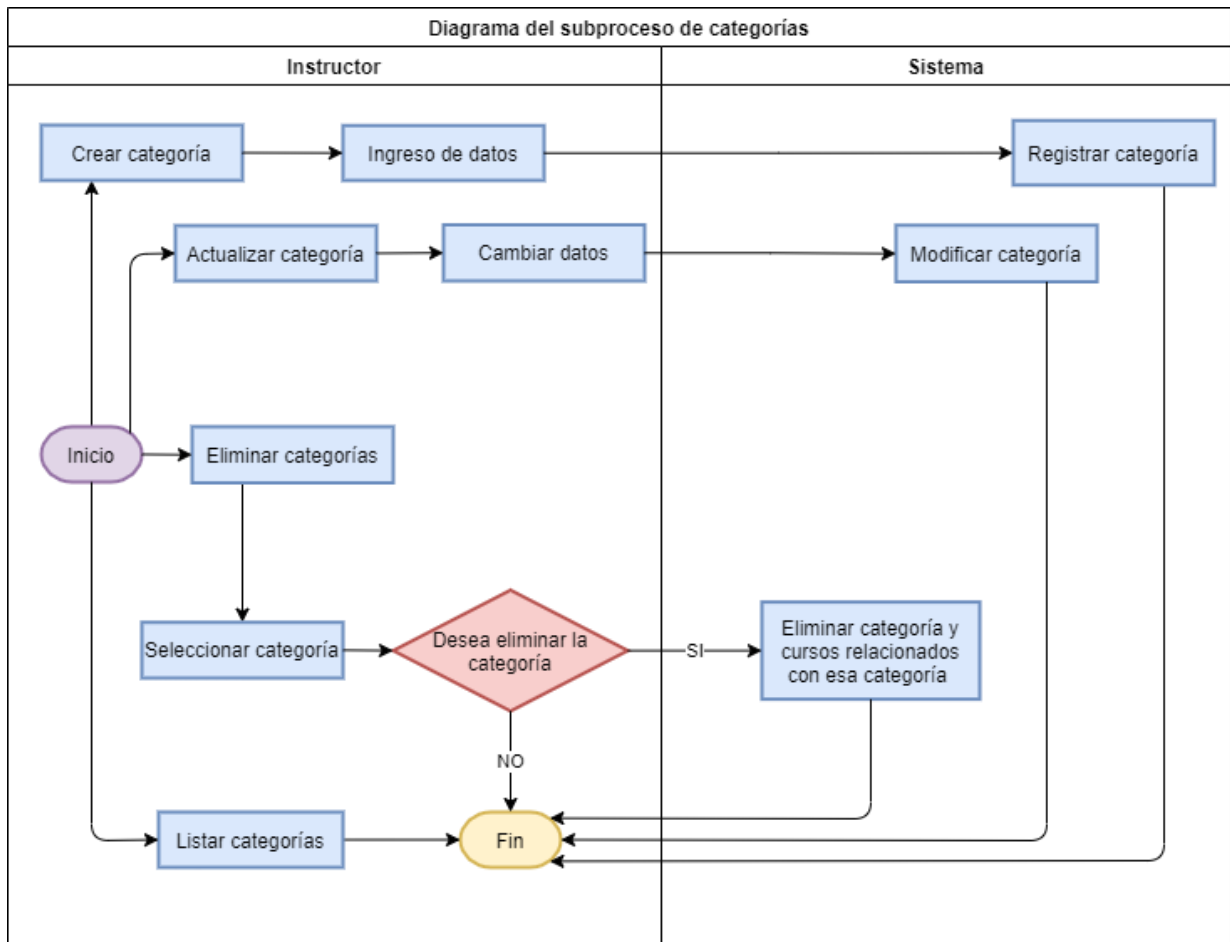


Figura 15 Diagrama del subproceso de categorías

Diagrama del subproceso de cursos: En la siguiente figura 16 se puede observar los pasos o secuencias que se tiene que tomar en cuenta al realizar un CRUD de cursos con sus módulos y contenidos de cada módulo, pero se tiene que tomar en cuenta que para el método de eliminación se debe preguntar al usuario si desea eliminar un curso de lo contrario no se debe eliminar ninguna información.

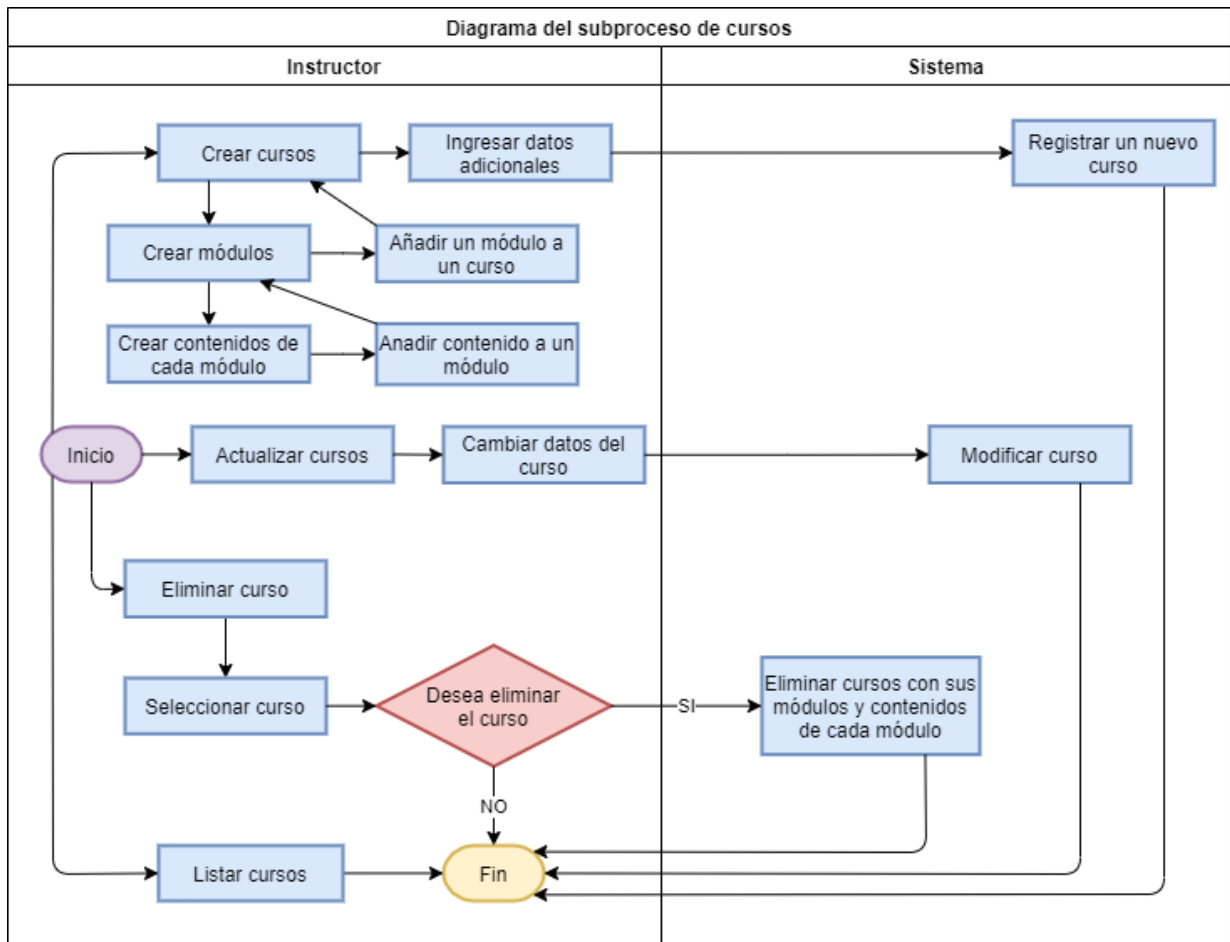


Figura 16 Diagrama del subproceso de cursos

Diagrama del subproceso de evaluaciones: En la siguiente figura 17 se puede visualizar una serie de pasos a tomar en cuenta para poder realizar un CRUD de las evaluaciones con sus preguntas y respuestas, pero se debe tomar en cuenta que para la eliminación de una evaluación se debe preguntar si el usuario desea eliminar una evaluación, de lo contrario no se debe eliminar ninguna información.

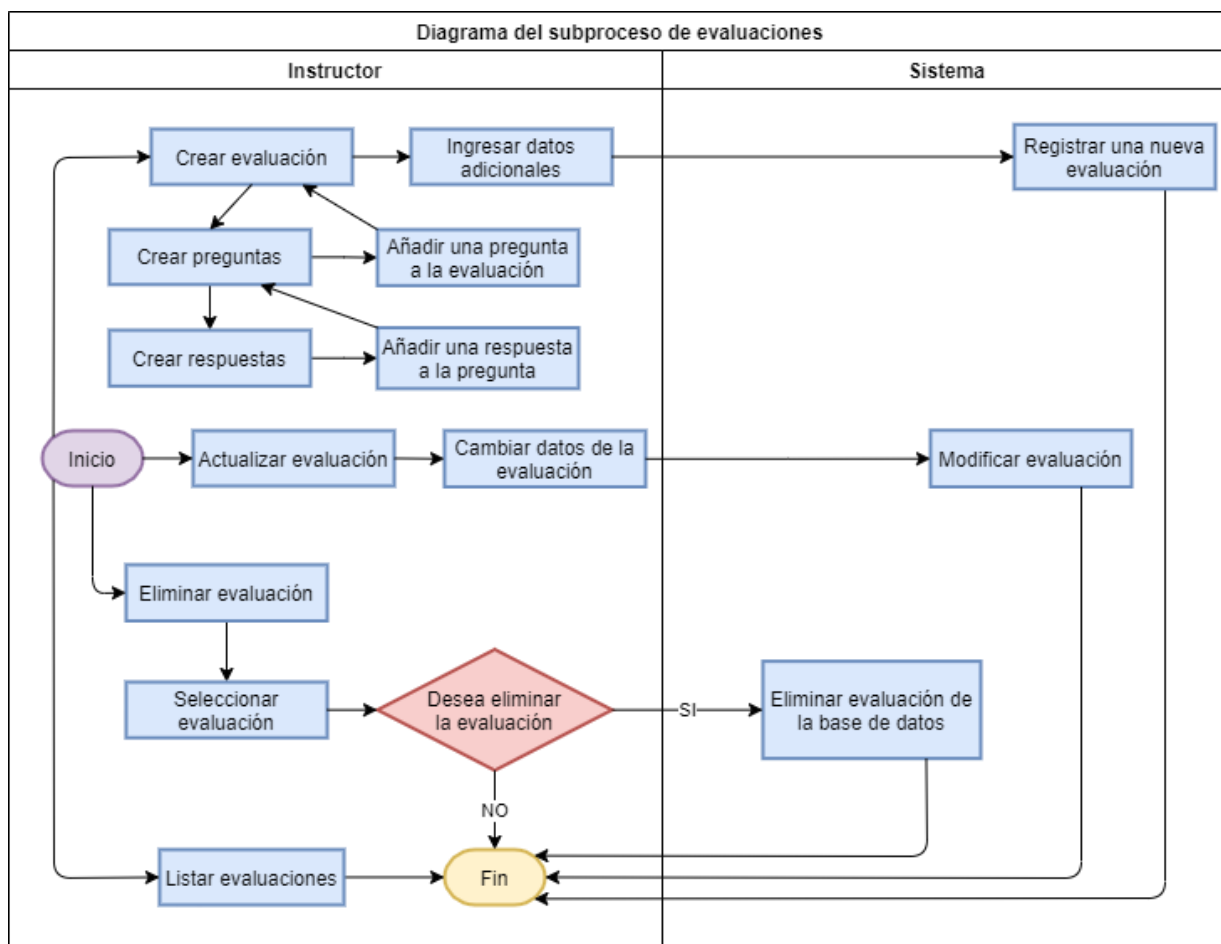


Figura 17 Diagrama del subproceso de evaluaciones

2.2.2.2 Diagrama de procesos del módulo de E-learning y evaluaciones

Estos procesos muestran lo que se debe automatizar en el módulo de E-learning y evaluaciones.

Adicional se debe tomar en cuenta que en todas las figuras que se muestran a continuación existen figuras de forma triangular de color rojo las cuales ejercen la función de validador o condicional de si o no para proseguir a un posterior paso, también existen rectángulos de color verde que representan los subprocesos realizados en la sección 2.2.2.1.

- Diagrama de procesos del usuario: A continuación, en la siguiente figura 18 se puede observar los pasos, validaciones que un usuario normal debe realizar para poder tener acceso a todas sus funcionalidades.

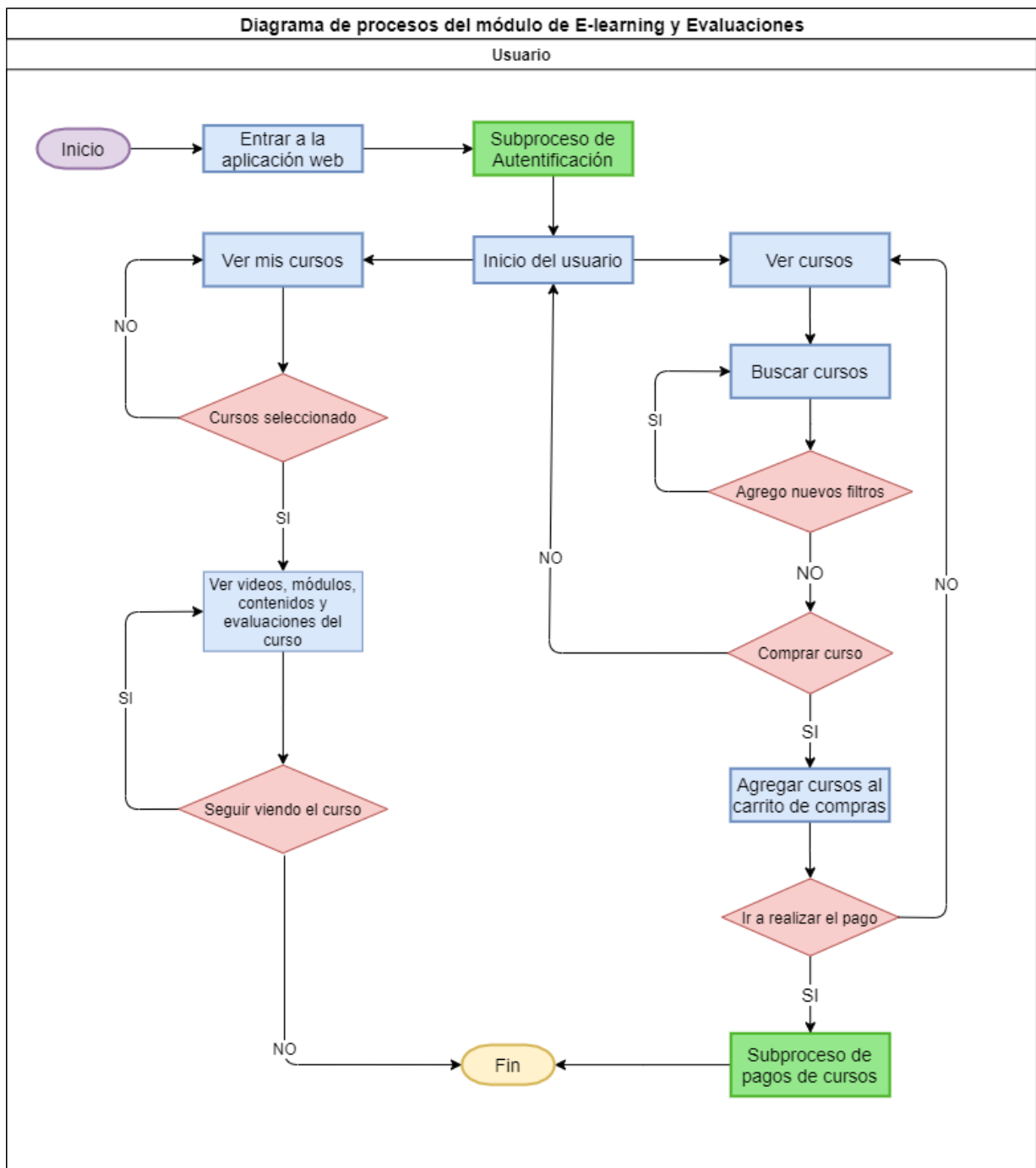


Figura 18 Diagrama de procesos del usuario en los módulos de E-learning y Evaluaciones

- Diagrama de procesos del instructor: A continuación, en la siguiente figura 19 se observa una secuencia de pasos, validaciones que un usuario instructor debe realizar para poder tener acceso a todas sus funcionalidades.

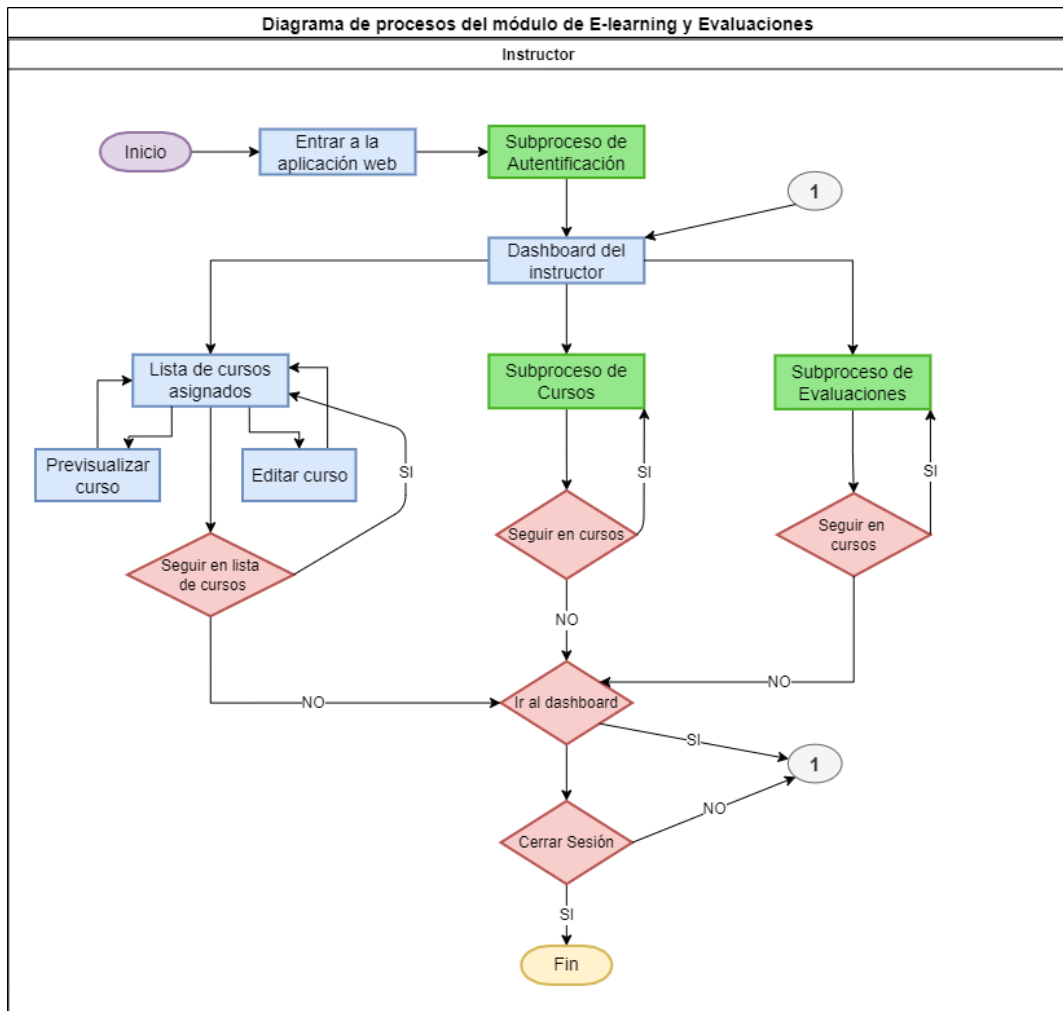


Figura 19 Diagrama de procesos del instructor en los módulos de E-learning y Evaluaciones

- Diagrama de procesos del administrador: A continuación, en la siguiente figura 20 se podrá observar los pasos, validaciones que un usuario administrador debe realizar para poder tener acceso a todas sus funcionalidades.

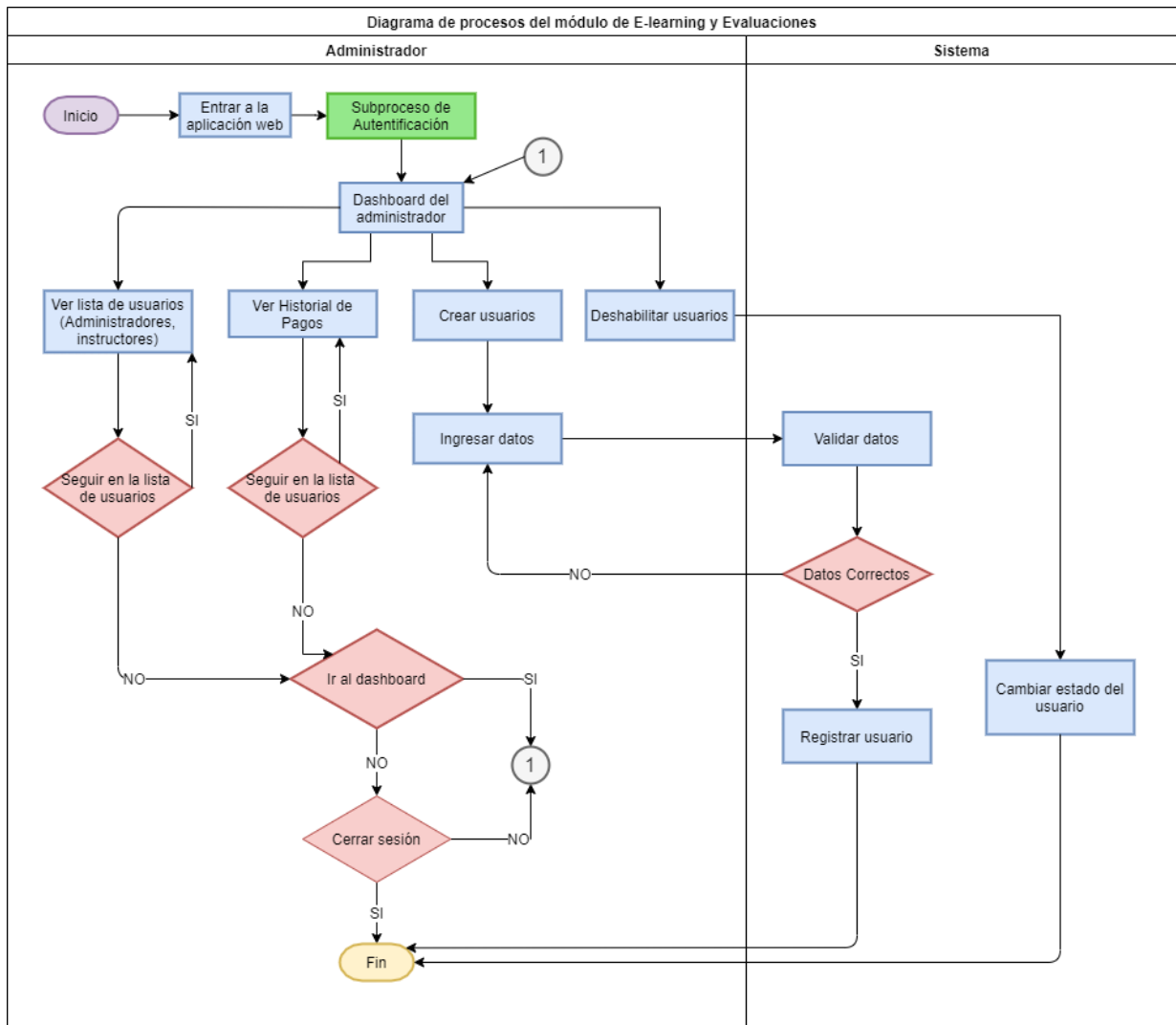


Figura 20 Diagrama de procesos del administrador en los módulos de E-learning y Evaluaciones

2.2.3 Modelo de la base de datos

Aquí se detalla el diseño de la base de datos que se utilizara en la aplicación web para almacenar la información de usuarios, categorías, cursos, evaluaciones, carrito de compras e historial de pagos.

El siguiente modelo de base de datos estará enfocado a la base de datos MongoDB (base de datos no relacional), por ende, se debe tomar en cuenta que “las relaciones” no afectan ningún proceso importante de la base de datos como pueden ser duplicidad de información, integridad de datos o accesibilidad de la información.

2.2.4 Casos de uso

Los siguientes casos de uso ayudaran a ilustrar las funciones y roles que cumple cada uno de los usuarios (usuario, instructor y administrador) en la aplicación permitiendo definir el alcance de la aplicación.

A continuación, el siguiente caso de uso representado con la figura 21 y la tabla 16 las funcionalidades del usuario administrador.

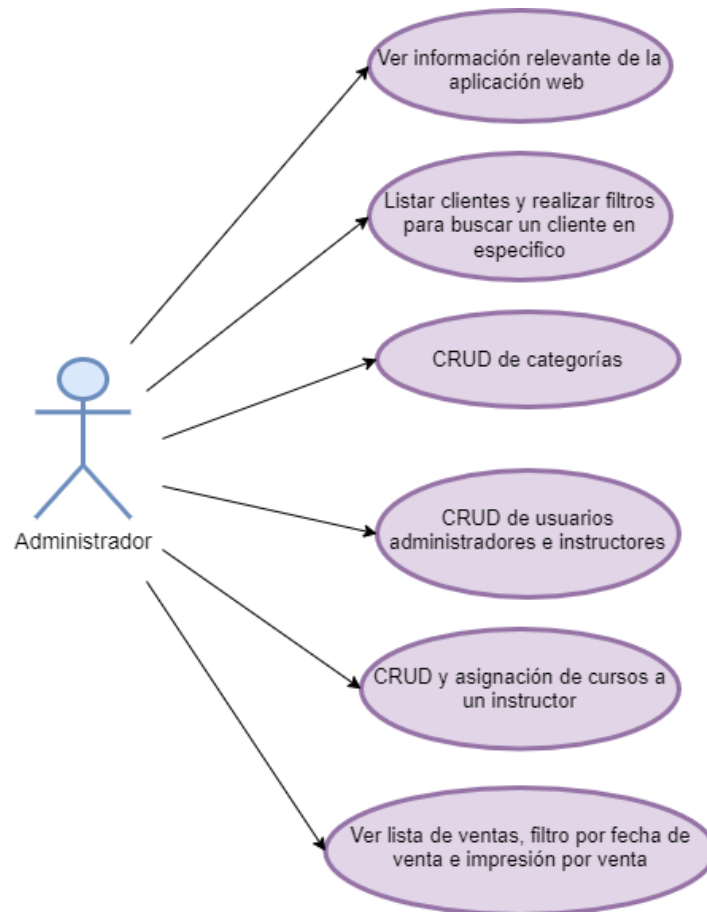


Figura 21 Caso de uso del usuario administrador

Tabla 15

Caso de uso del usuario administrador

Identificador del caso de uso	CU_UADMINISTRADOR
Nombre caso de uso	USUARIO ADMINISTRADOR
Actores	ADMINISTRADOR

Propósito	<p>Acceso al control de usuarios instructores y administradores, ver clientes, CRUD de cursos, CRUD de categorías y visualización del historial de ventas.</p> <p>El administrador podrá gestionar, visualizar y controlar la creación, eliminación y activación o desactivación de un usuario administrador o bien sea instructor.</p>
Visión general	<p>Ver historial de ventas, clientes e información general de la aplicación web.</p> <p>CRUD de cursos y categorías</p> <p>Además, el administrador podrá visualizar todas las compras realizadas por los usuarios.</p>
Tipo	Primario, Esencial
Curso de eventos	Secuencial
Acciones del actor	Respuesta del sistema
Administrador inicia sesión	El administrador debe ingresar sus credenciales para poder tener accesos a sus funcionalidades.
Administrador ingresa a la sección clientes	El administrador podrá visualizar todos los clientes y realizar un filtro para realizar una búsqueda más eficiente.
Administrador ingresa a la sección de usuarios	El administrador podrá visualizar, crear, eliminar y activar o desactivar un usuario administrador o instructor.
Administrador ingrese a la sección de historial de ventas	El administrador podrá visualizar los registros de compras realizadas por los usuarios en donde podrá observar cual usuario realizo la comprar el costo de la comprar y los detalles del curso que compro.
Administrador ingrese a la sección de categorías	El administrador podrá crear, listar, actualizar y eliminar categorías que no tengan cursos referentes.

Administrador ingrese a la sección cursos

El administrador crear, listar, actualizar y eliminar cursos, los cursos solo podrán ser eliminados si no existe ningún estudiante que haya comprado el curso aún.

En el siguiente caso de uso representado en la figura 22 y en la tabla 17 se puede observar las actividades y funciones que podrá realizar el usuario instructor después de ser creado por el usuario administrador.

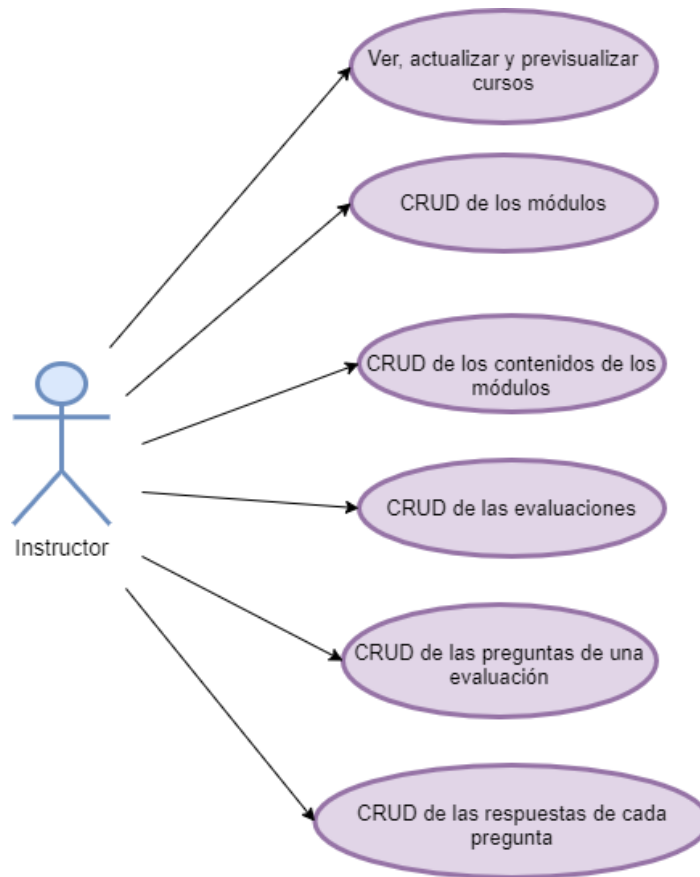


Figura 22 Caso de uso del usuario instructor

Tabla 16

Caso de uso del usuario instructor

Identificador del caso de uso	CU_UINSTRUCTOR
Nombre caso de uso	USUARIO INSTRUCTOR
Actores	INSTRUCTOR

	Ver los cursos asignados, actualizarlos y previsualizarlos como quedaran a la hora de publicarlos.
Propósito	Tener acceso al CRUD de módulos, contenidos, evaluaciones, preguntas y respuestas que estarán plasmados en un curso.
Visión general	El instructor podrá crear, visualizar, actualizar y eliminar todo el contenido para un curso asignado, además de poder visualizarlo antes de publicarlo y tener una retroalimentación de cómo quedará de cara al usuario final.
Tipo	Primario, Esencial
Curso de eventos	Secuencial
Acciones del actor	Respuesta del sistema
Instructor inicia sesión	El instructor debe ingresar sus credenciales para poder tener accesos a sus funcionalidades.
Instructor ingresa a la sección cursos asignados	El instructor podrá observar los cursos asignados, actualizarlos y previsualizarlos antes de publicarlos.
Instructor ingresa a la sección de módulos	El instructor podrá visualizar, crear, actualizar y eliminar módulos del curso.
Instructor ingrese a la sección de contenidos	El instructor podrá visualizar, crear, actualizar y eliminar contenidos de cada modulo
Instructor ingresa a la sección de evaluaciones	El instructor podrá visualizar, crear, actualizar y eliminar evaluaciones de un curso.
Instructor ingresa a la sección de preguntas	El instructor podrá visualizar, crear, actualizar y eliminar preguntas de una evaluación.
Instructor ingresa a la sección de respuestas	El instructor podrá visualizar, crear, actualizar y eliminar respuestas de una pregunta.

En el siguiente caso de uso representado por la figura 23 y la tabla 18 se puede observar las actividades y funciones que podrá realizar el usuario común después de haberse registrado.

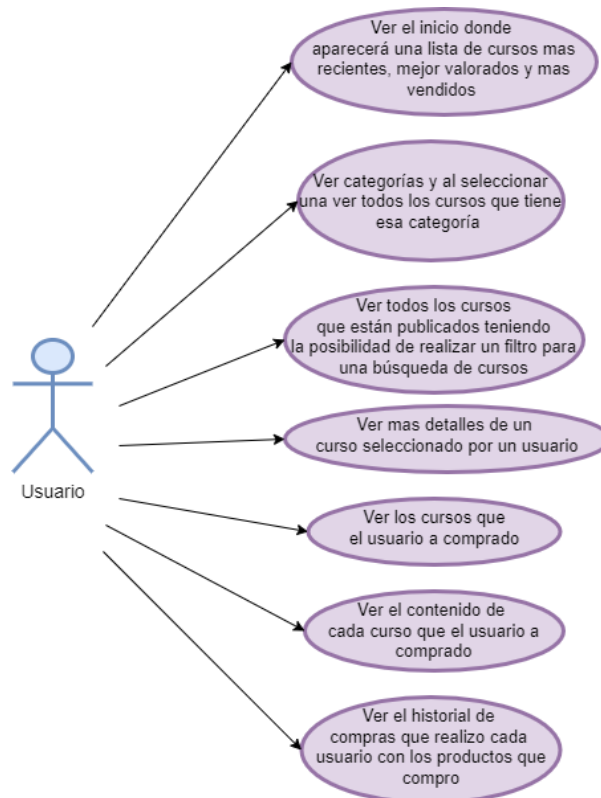


Figura 23 Caso de uso del usuario

Tabla 17

Caso de uso del usuario

Identificador del caso de uso	CU_USUARIO
Nombre caso de uso	USUARIO
Actores	USUARIO
Propósito	Visualización de categorías, cursos, mis cursos, carrito de compras, historial de compras, además de la posibilidad de modificación del carrito de compras y el perfil de usuario.
Visión general	El usuario podrá visualizar las categorías, cursos, sus cursos comprados, historial de compras realizadas, ítems

en el carrito de compras, datos o información necesaria sobre el usuario, además de contar la posibilidad de eliminar o añadir nuevos ítems al carrito de compras y la opción de poder modificar sus datos personales en el perfil de usuario.

El usuario podrá realizar pagos de ítems que se encuentren agregados en el carrito de compras mediante una tarjeta de crédito o débito utilizando el método de pago PayPal.

Tipo	Primario, Esencial
Curso de eventos	Secuencial
Acciones del actor	Respuesta del sistema
Usuario inicia sesión	El usuario debe ingresar sus credenciales para poder tener accesos a las funcionalidades.
Usuario ingresa a la sección de inicio del usuario	En esta sección se presentan los cursos mejor calificados, los cursos más vendidos y los cursos más nuevos.
Usuario ingresa a la sección de categorías	Aquí el usuario podrá seleccionar la categoría para que pueda ver los cursos de esa categoría.
Usuario ingresa a la sección de cursos	Aquí el usuario podrá ver todos los cursos que existen en la plataforma, teniendo la opción de realizar filtros para realizar una búsqueda mucho más eficiente.
Usuario ingresa a la sección de historial de compras	El usuario podrá visualizar todos los cursos que ha comprado con sus detalles, costo y fecha de cuando realizo la compra.
Usuario ingresa a la sección de carrito de compras	El usuario podrá visualizar los ítems añadidos para una posterior compra, además podrá eliminar o seguir añadiendo más ítems al carrito de compras.

Usuario ingresa a la sección de perfil de usuario	El usuario podrá visualizar su información personal, la cual podrá ser modificada con los datos requeridos por la aplicación.
Usuario ingresa a la sección de proceso de pago	El usuario podrá realizar pagos de los ítems añadidos al carrito de compras mediante tarjeta de crédito o débito mediante PayPal.

2.3 Codificación

En esta fase se procede a realizar la configuración de las herramientas que se utilizarán y de esa manera proseguir con el desarrollo de la aplicación web en base a los que se presentó en la base de diseño.

2.3.1 Configuración del entorno de desarrollo

El desarrollo de la aplicación web se realizó en dos sistemas operativos los cuales fueron Windows 10 home de 64 bits y Garuda Linux de 64 bits, además se utilizó las siguientes herramientas y software para la construcción del sistema.

- Visual Studio Code: es un editor de código abierto desarrollado por la empresa Microsoft.
- Hackolade: es un software que ayuda al modelado de bases de datos no SQL.
- Node.Js: es un entorno de ejecución de JavaScript para el desarrollo de aplicaciones.
- Express: es un framework de backend que ayuda a la creación de Apis.
- MongoDB: es una base de datos no SQL basa en documentos.
- Nginx: es un proxy que ayuda a la ejecución de aplicaciones en producción.
- Git: es un gestor de control de versiones para poder seguir y controlar el código realizado en una aplicación.
- GitHub: es un repositorio donde se pueden almacenar los archivos de nuestro código fuente para poder tener acceso a él desde cualquier dispositivo.
- TypeScript: es el lenguaje de tipado que funciona en base a JavaScript.
- React: Framework de frontend que ayuda a la creación de interfaces usuario.

- Redux: es un gestor del estado de react.

2.3.2 Creación y ejecución de la base de datos

En esta sección se crea la base de datos con su respectiva puesta en marcha en MongoDB Atlas utilizando la librería mongoose, la cual ayuda a generar un modelo de base de datos de manera automática declarando los modelos en archivos TypeScript, lo cual generando el siguiente modelo de base de datos representado en la figura 24.

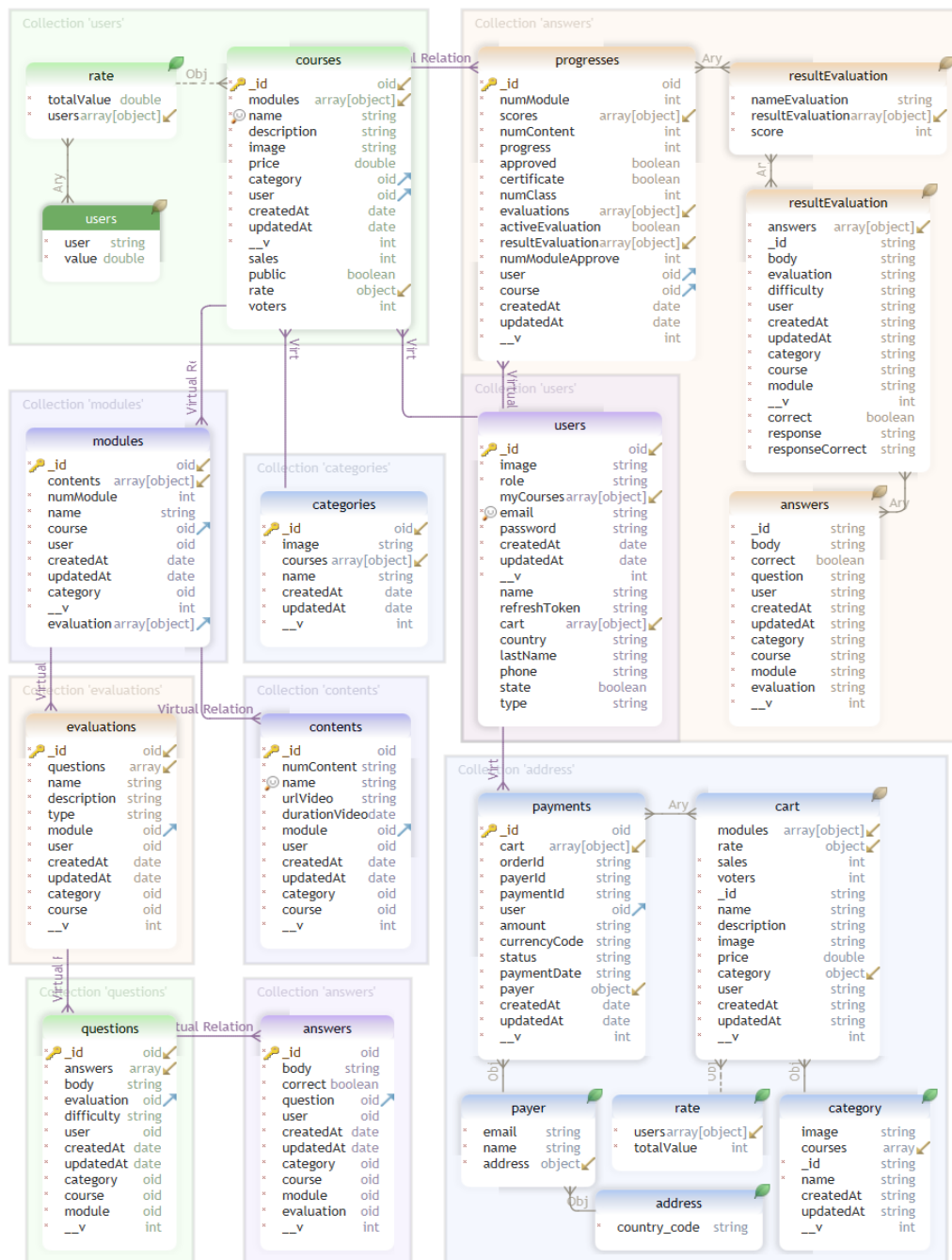


Figura 24 Modelo de la base de datos en MongoDB

2.3.3 Creación del backend de los módulos de E-learning y evaluaciones

En esta sección, se desarrolla la configuración del servidor web, la conexión y modelos de la base de datos, controladores y rutas para la consulta de información en la API REST.

A continuación, para la creación de una nueva aplicación con node.js se debe realizar el siguiente comando el cual es **npm init** el cual nos pedirá una serie de información de la aplicación de manera opcional como se muestra a continuación en la figura 25.

```
n/~/nueva app
11:52:17 > npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help init` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (nueva-app) nueva app
Sorry, name can only contain URL-friendly characters.
package name: (nueva-app)
version: (1.0.0)
description:
entry point: (index.js)
test command:
git repository:
keywords:
author:
license: (ISC)
About to write to C:\Users\Omar\Downloads\nueva app\package.json:

{
  "name": "nueva-app",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC"
}

Is this OK? (yes)
n/~/nueva app
11:52:35 > █
```

Figura 25 Creación de una nueva app con Node.Js

Una vez creada la nueva app, esta se creará con un solo archivo de configuración el cual almacena información de las dependencias (librerías), dependencias de desarrollo y scripts necesarios para el correcto funcionamiento de la aplicación. Este archivo se lo conoce como package.json como se muestra a continuación en la figura 26.

```
package.json X
package.json > ...
1 {
2   "name": "backend",
3   "version": "1.0.0",
4   "description": "",
5   "main": "index.js",
6   "scripts": {
7     "start": "node dist/index.js",
8     "dev": "ts-node-dev src/index.ts",
9     "build": "tsc",
10    "lint": "eslint '**/*.ts' --quiet --fix",
11    "prepare": "husky install"
12  },
13  "keywords": [],
14  "author": "",
15  "license": "ISC",
16  "dependencies": { ...
32  },
33  "devDependencies": { ...
54  },
55  "lint-staged": { ...
60  }
61 }
62 |
```

Figura 26 Archivo de configuración de una aplicación hecha con Node.Js y TypeScript

Una vez creado la app además del archivo de configuración deberemos crear toda la estructura de carpetas que el usuario vea necesario para su aplicación como se visualiza en la siguiente figura 27.

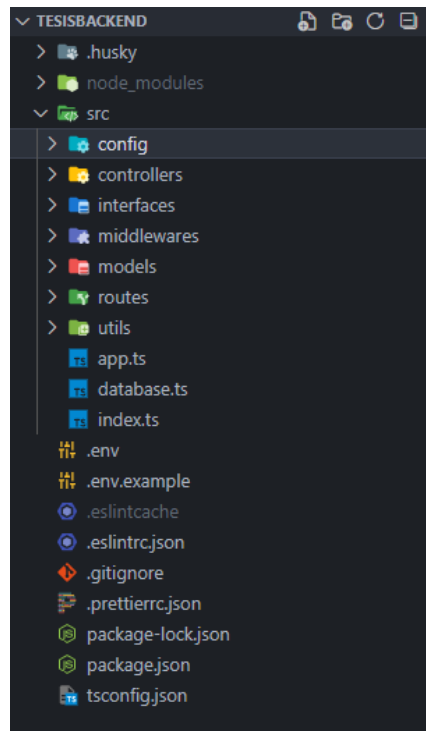


Figura 27 Estructura de carpetas de los módulos web

2.3.3.1 Creación de los modelos en el backend

Una vez creada la estructura del proyecto se deberá crear los modelos de la base de datos en archivos TypeScript sin necesidad de haber modelado la base de datos previamente.

La siguiente figura 28 muestra cómo se debe realizar estos modelos para creación de colecciones en la base de datos MongoDB.

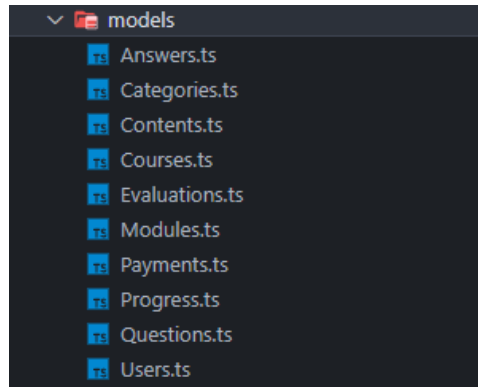


Figura 28 Archivos de los modelos de la base de datos

Cada uno de estos archivos va ligado a una interfaz la cual permite crear solo los atributos declarados en la interfaz previa.

En la siguiente figura 29 presentada a continuación se puede visualizar la estructura de cómo se crearon cada uno de los modelos.

```
Users.ts x
src > models > Users.ts > ...
1 import bcrypt from 'bcrypt';
2 import paginate from 'mongoose-paginate-v2';
3 import uniqueValidator from 'mongoose-unique-validator';
4 import { LeanDocument, model, PaginateModel, Schema, Types } from 'mongoose';
5 import { IUser, IUserModel } from '../interfaces/user.interface';
6 import { TypeSchema } from '../interfaces/mongoose.interface';
7
8 > const userSchema: TypeSchema<IUserModel, IUser> = new Schema(...
45   },
46   { timestamps: true },
47 );
48
49 userSchema.methods.matchPassword = async function matchPassword(password: string):
Promise<boolean> { ...
55 };
56
57 userSchema.methods.toJSON = function toJSON(): LeanDocument<IUserModel> { ...
62 };
63
64 userSchema.plugin(uniqueValidator as any, { message: 'Error, el correo electrónico ya existe' }
);
65
66 userSchema.plugin(paginate as any);
67 export default model<IUserModel>('User', userSchema) as PaginateModel<IUserModel>;
68 |
```

Figura 29 Estructura para la creación de modelos en MongoDB

2.3.3.2 Creación de los controladores del backend

Los controladores creados en la aplicación web de preferencia se los debe poner un nombre característico para diferenciarlos de los otros ya que a la hora de programar puede haber confusiones con nombres de otros archivos como pueden ser modelos o rutas.

Así que en la siguiente figura 30 se muestra cómo se detalla el nombre de todos los archivos que conforman los controladores de la aplicación.

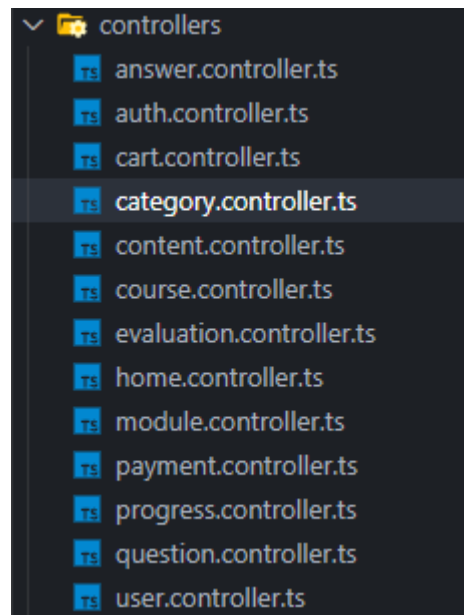


Figura 30 Archivos que conforman los controladores de la API REST

Estos controladores son métodos que ayudan a la creación, extracción, actualización y eliminación de información en la base de datos, mediante una previa petición realizada a la API REST.

La estructura de un controlador se lo podrá visualizar en la siguiente figura 31.

```
category.controller.ts X
src > controllers > category.controller.ts > ...
1 import { Request, Response } from 'express';
2 import { ICategory, ICategoryModel } from '../interfaces/category.interface';
3 import Categories from '../models/Categories';
4 import { queryDataBase } from '../utils/queryDataBase';
5 import responseError from '../utils/responseError';
6
7 > export const getCategories = async (req: Request, res: Response): Promise<Response> => { ...
14 };
15
16 > export const getAllCategories = async (req: Request, res: Response): Promise<Response> => { ...
23 };
24
25 > export const getCategory = async (req: Request, res: Response): Promise<Response> => { ...
32 };
33
34 > export const createCategory = async (req: Request, res: Response): Promise<Response> => { ...
55 };
56
57 > export const updateCategory = async (req: Request, res: Response): Promise<Response> => { ...
75 };
76
77 > export const deleteCategory = async (req: Request, res: Response): Promise<Response> => { ...
85 };
```

Figura 31 Estructura para la creación de controladores de una API REST

2.3.3.3 Creación de las rutas del backend

Las rutas de una API REST tienen que ser diferenciadas con un nombre o una terminación diferente a todos los archivos creados en el backend ya que puede ocasionar muchas confusiones para importar archivos de código.

Para evitar confusiones entre archivos de la aplicación se han creado rutas con los siguientes nombres y terminación específica como se muestra en la siguiente figura 32.

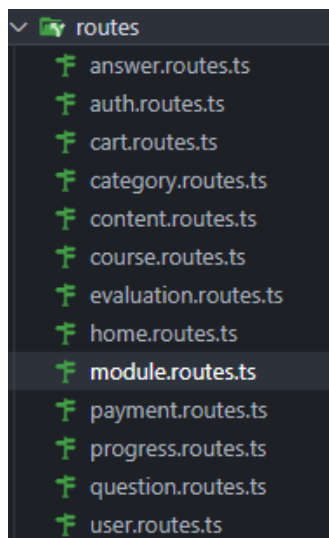


Figura 32 Archivos que conforman las rutas de la API REST

Las rutas de la API REST son URLs que ayudan a la petición y devolución de información mediante el protocolo http o https a una aplicación backend haciendo que la

información pueda ser consumida por cualquier aplicación desde cualquier sistema web frontend siguiendo reglas de consumo hacia la API REST.

En la siguiente figura 33 se muestra un ejemplo de cómo se estructuran este tipo de rutas para el consumo de una API REST.

```
import { Router } from 'express';
import {
  createCategory,
  deleteCategory,
  getAllCategories,
  getCategories,
  getCategory,
  updateCategory,
} from '../controllers/category.controller';
import { auth, authAdmin } from '../middlewares/authentication';

const router = Router();

router.route('/').get(getCategories).all(auth).post(authAdmin, createCategory);
router.route('/all').get(getAllCategories);
router.route('/:id').get(getCategory).all(auth, authAdmin).put(updateCategory).delete(deleteCategory);

export default router;
```

Figura 33 Estructura para la creación de rutas de una API REST

2.3.4 Configuración y preparación para el desarrollo frontend

En esta sección se debe empezar a seleccionar todas las librerías y herramientas que se utilizarán para la creación y de las interfaces de usuario mediante las siguientes librerías React, Redux, Sass, Ant Design.

Para empezar un nuevo proyecto de React con TypeScript necesitamos realizar el siguiente comando en la consola de nuestro equipo: **npx create-react-app nombre-app --template typescript**, este comando nos creará la siguiente estructura de carpetas que se representan en la siguiente figura 34.

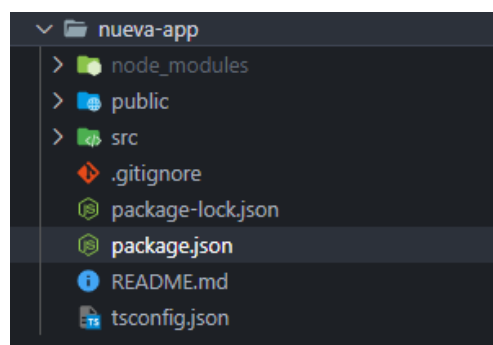


Figura 34 Estructura de carpetas al crear una nueva app de React con TypeScript

Una vez terminada la creación de la nueva app de manera opcional se puede configurar ciertas reglas de desarrollo mediante Eslint, Prettier, y husky los cuales ayudan a tener un mejor control y parametrización del código fuente que se realiza en la aplicación.

Creando y configurando una serie de archivos extra a la aplicación inicial los cuales se los puede visualizar en la siguiente figura 35.

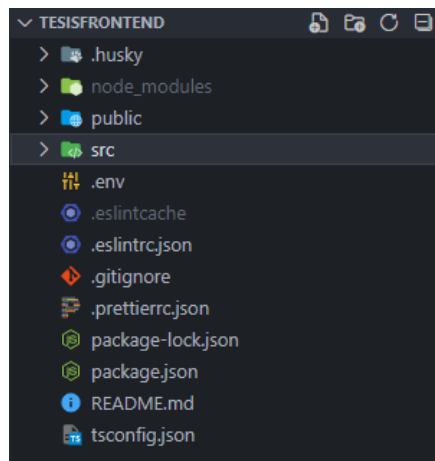


Figura 35 Archivos de configuración para establecer reglas y estandarización de código limpio

2.3.4.1 Estructura Diseño atómico (Atomic Design)

El diseño atómico es una metodología de construcción de sistemas de diseño que encaja con React debido a la forma en la que se crean las interfaces basadas en componentes, y más que una forma completa de estructurar nuestro proyecto es una forma de organizar sus componentes.

Y la idea principal es dividir nuestros componentes en cinco tipos de elementos los cuales son: components, contexts, hooks, pages, routes, utils, services, App.jsx, index.js.

En nuestro caso debemos tomar en cuenta se utiliza Redux que es una librería que ayuda a la gestión del estado de React, por lo cual la carpeta services y routes deben ser sustituidas por una carpeta redux que deberá contener: actions, reducers, store,ts que son uno de los conceptos que maneja esta librería como se puede mostrar en la siguiente figura 36.

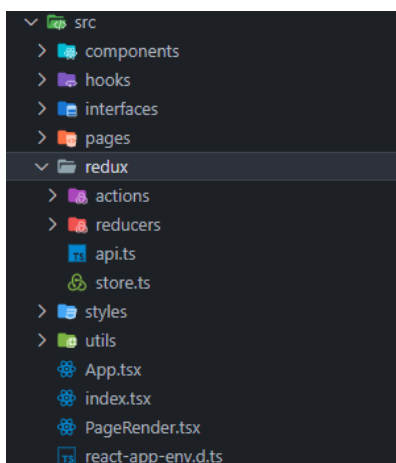


Figura 36 Estructura diseño atómico en React, Redux y TypeScript

2.3.4.2 Hooks en el frontend

Los hooks de React son una gran herramienta de desarrollo que ayudaron a disminuir la complejidad y errores que existía al intentar enviar y recibir datos de consultas a una API REST o condiciones de activar o desactivar componentes visuales.

Además, React permite la creación de hooks customizados para la reutilización y separación del código fuente con el código visual de la aplicación en un componente, como se representa a continuación en la siguiente figura 37.

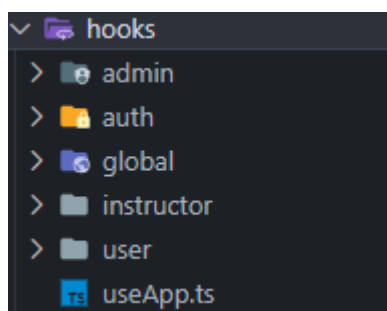


Figura 37 Hooks customizados en React

2.3.4.3 Pages en el frontend

En esta sección se realiza el diseño principal de una página ejemplo: Sign In la cual puede estar estructurada como un lego de varios componentes pequeños reutilizables en toda la aplicación ejemplo botones o inputs.

Por lo que para intentar tener un correcto orden se crean carpetas y archivos con nombres descriptivos para evitar confusiones y errores en importaciones de componentes como se puede visualizar en la siguiente figura 38 de la capeta pages.

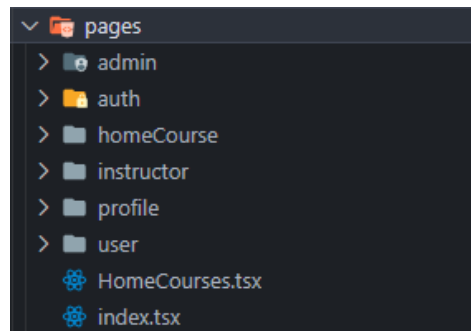


Figura 38 Estructura de la carpeta Pages en el frontend

2.3.4.4 Componentes en el frontend

Los componentes en React permiten la reutilización de código fuente debido a que se puede separar en pequeñas parcelas de código y ser implementadas en cualquier componente principal o secundario de la aplicación.

Los componentes por lo general siempre deben llevar un nombre descriptivo el cual los identifique, como se muestra a continuación en la figura 39.

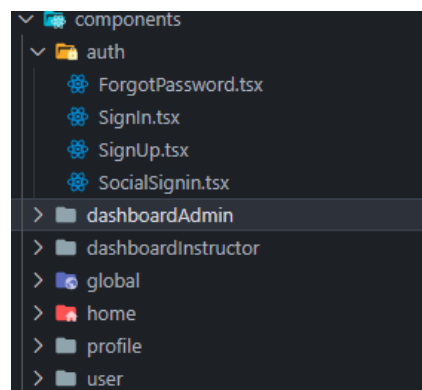


Figura 39 Componentes visuales del frontend

2.3.4.5 Redux para el control del estado de React

Redux ayuda al control del estado de React, que básicamente es una forma de almacenamiento temporal de la información que se extrae de una API REST pero que solo dura cuando la aplicación no tiene ningún cambio en la dirección URL o refresco del navegador.

Esta librería tiene 3 conceptos principales tipos, acciones y reductores, además de su configuración de un store que es donde se declaran las variables que existirán en el estado de React que es como una caja que guarda información accesible desde cualquier vista de la aplicación.

Los archivos de esta librería de Redux de preferencia deben ser declarados bien sea con la terminación action o reducer para reconocer que hace cada archivo como se presenta a continuación en la figura 40.

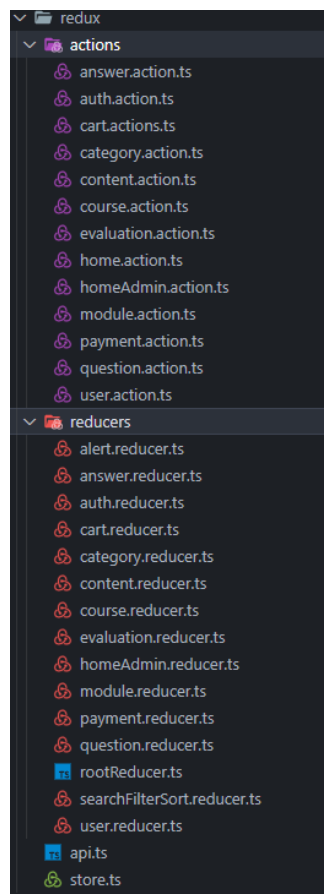


Figura 40 Redux como control del stado de React

2.4 Pruebas

Para la evaluación del funcionamiento adecuado del backend y frontend de la aplicación se utilizó herramientas que ayudan a emular y visualizar la información que se envía y se entrega mediante una API REST.

Entre una de las herramientas que se tiene para el testeo de rutas de una API REST podemos encontrar Postman que es un software que permite él envío y devolución de

información mediante una ruta hacia una API REST, permitiendo verificar que dicha información es la que nosotros esperamos que devuelva el backend.

Otra de las herramientas útiles para el desarrollo de la aplicación web por parte del frontend podemos encontrar React Developer Tools, Redux DevTools y DevTools del navegador que permiten controlar las interfaces de usuario, el estado de la información devuelta por el backend y propiedades que puede tener un componente en ese momento de ejecución o visualización en el navegador.

2.4.1 Pruebas con Postman en el backend

Postman es una aplicación que nos permite realizar pruebas de API REST, ya que es un cliente HTTP que nos da la posibilidad de inspeccionar solicitudes HTTP a través de una interfaz gráfica de usuario, la cual recibe diferentes tipos de respuestas que luego tendremos que autenticar.

2.4.1.1 Métodos HTTP

Postman nos ofrece muchos métodos para interactuar con APIs REST como por ejemplo los más usados:

- GET: Obtener información
- POST: Agregar información
- PUT: Reemplazar la información
- PATCH: Actualizar información
- DELETE: Borrar información

2.4.1.2 Códigos de respuesta

Al testar una API REST con Postman, podremos obtener diversos códigos de respuesta, pero los más comunes suelen ser:

- Serie 100 > Respuestas Temporales, por ejemplo: '102 Processing'.
- Serie 200 > Respuestas donde el cliente acepta el request, siendo procesado exitosamente en el servidor, por ejemplo: '200 Ok'.

- Serie 300 > Respuestas relacionadas a redireccionamiento URL, por ejemplo: '301 Moved Permanently'.
- Serie 400 > Respuestas de error del lado del cliente, por ejemplo: '400 Bad Request'.
- Serie 500 > Respuestas de error del lado del servidor, por ejemplo: '500 Internal Server Error'.

2.4.1.3 Validación con Postman de la API REST

En esta sección se realiza la comprobación de respuesta de cada una de las rutas que se creó en el backend para validar que las rutas devuelvan lo esperado sin obtener un código de error como por ejemplo error 400, error 401, error 500 y entre otros errores que suelen ocurrir en una API REST.

La manera en que se suelen realizar dichas pruebas es mediante la inserción de una ruta HTTP con su método de respuesta, sus parámetros necesarios que usualmente pueden ser las cabeceras y o cuerpo de la consulta HTTP como se puede representar a continuación en la siguiente figura 41.

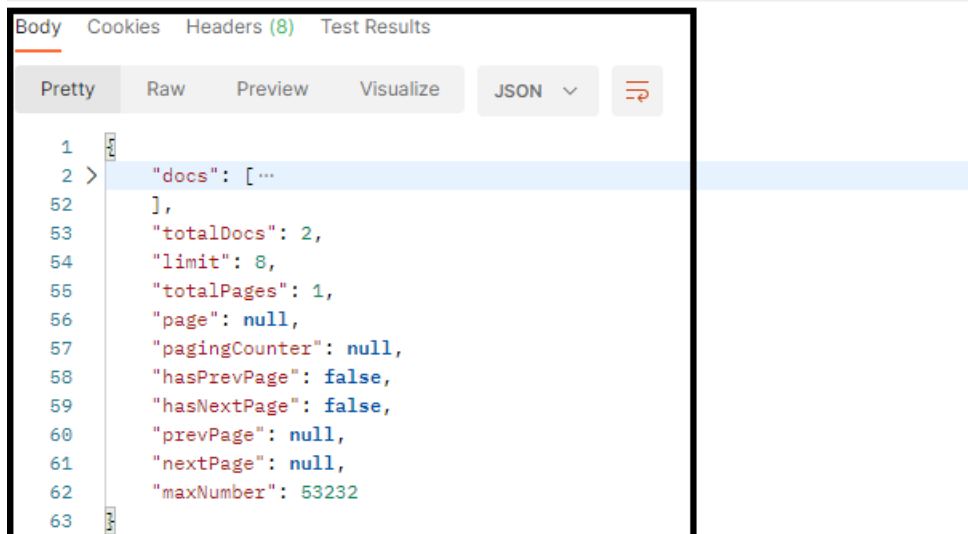
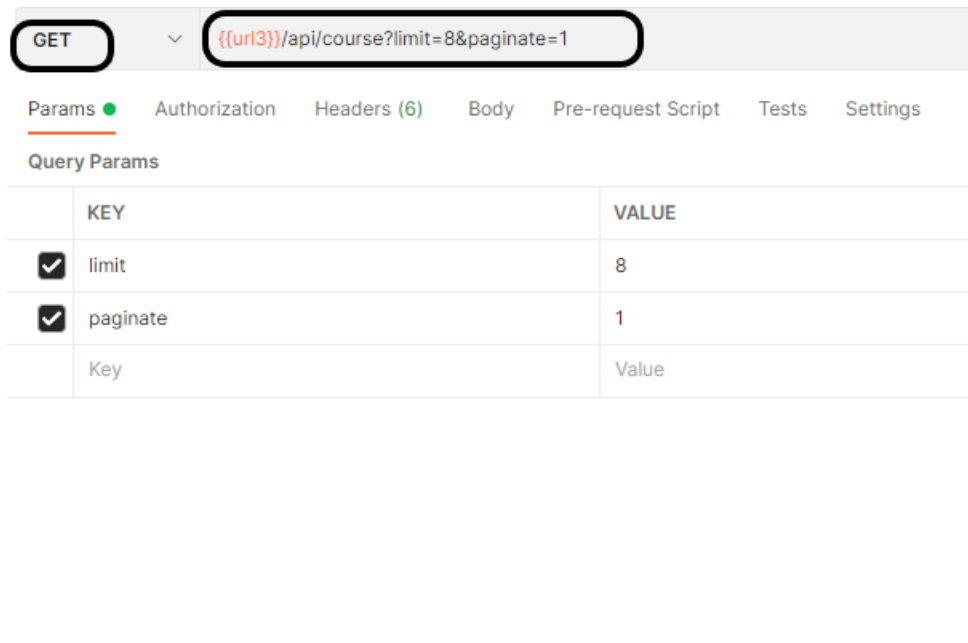


Figura 41 Validación de una API con Postman

2.4.2 Pruebas con React Developer Tools en el frontend

React Developer Tools es una extensión del navegador que ayuda a validar, verificar y visualizar información necesaria de un componente, que permite solucionar errores visuales y de comportamiento de un componente como por ejemplo el renderizado de información de una componente hacia otro, ocultar o desactivar componentes en un espacio de la pantalla, entre otros.

Esta extensión de una forma fácil y sencilla de validar que un componente contiene una secuencia de información y visualización de datos que a la hora de realizar las interfaces de usuario son muy útiles en la detección de errores de un componente.

Para realizar estas pruebas deberemos tener componentes que se visualicen en una página donde nosotros inspeccionaremos el componente necesario y procederemos a verificar que el componente contiene lo que nosotros esperamos que aparezca en la pantalla principal sino deberemos corregir el componente como se presenta a continuación en la siguiente figura 42.

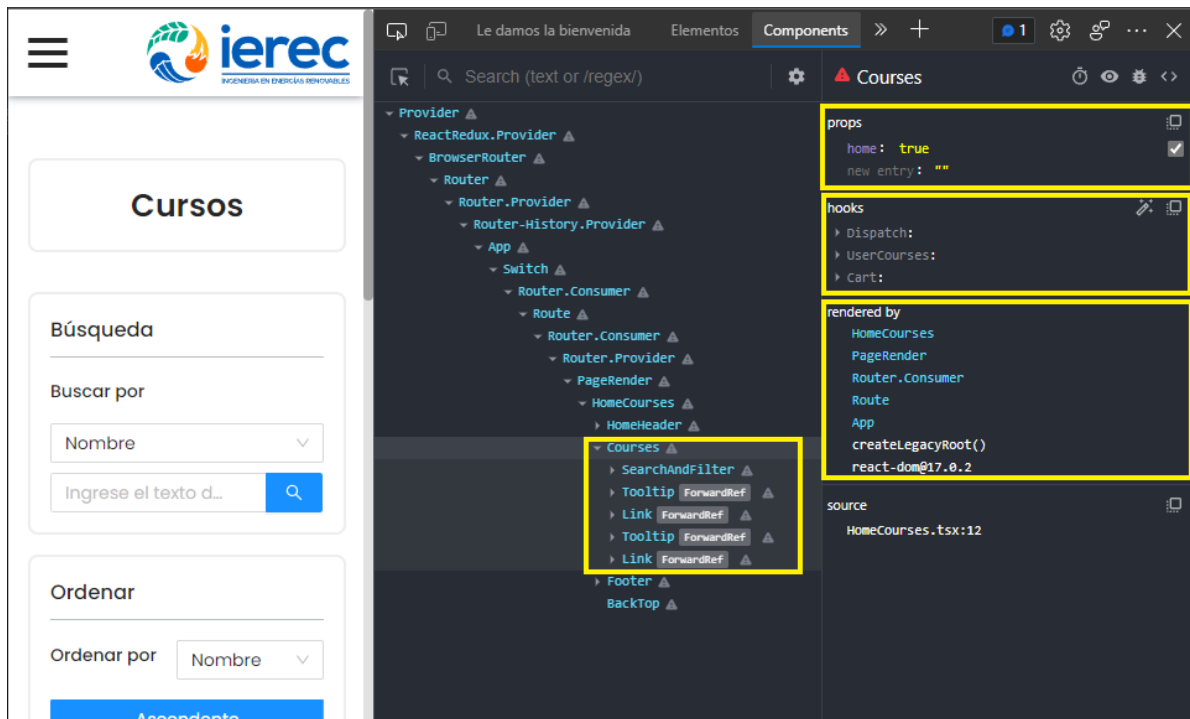


Figura 42 Herramienta para validación de componentes en React

2.4.3 Pruebas con Redux DevTools en el frontend

Redux DevTool es una herramienta que es de mucha ayuda para conocer y validar el proceso de petición, espera y devolución de información de una API REST la cual consta de una forma de reproducir todo el proceso de petición y devolución de datos.

Además, con esta herramienta útil podemos visualizar todos los datos almacenados en el estado de React que es como un estado de memoria que almacena información de manera temporal mientras un usuario no refresque el navegador o cambie de dirección URL de manera manual.

Para el proceso de validación se debe cargar una página desde cero y empezar a visualizar si la información de los eventos y datos que devuelve cada evento para ver si las peticiones al backend están bien realizadas y sino proceder a su debida corrección, a continuación, se presenta en la figura 43 como es esta herramienta.

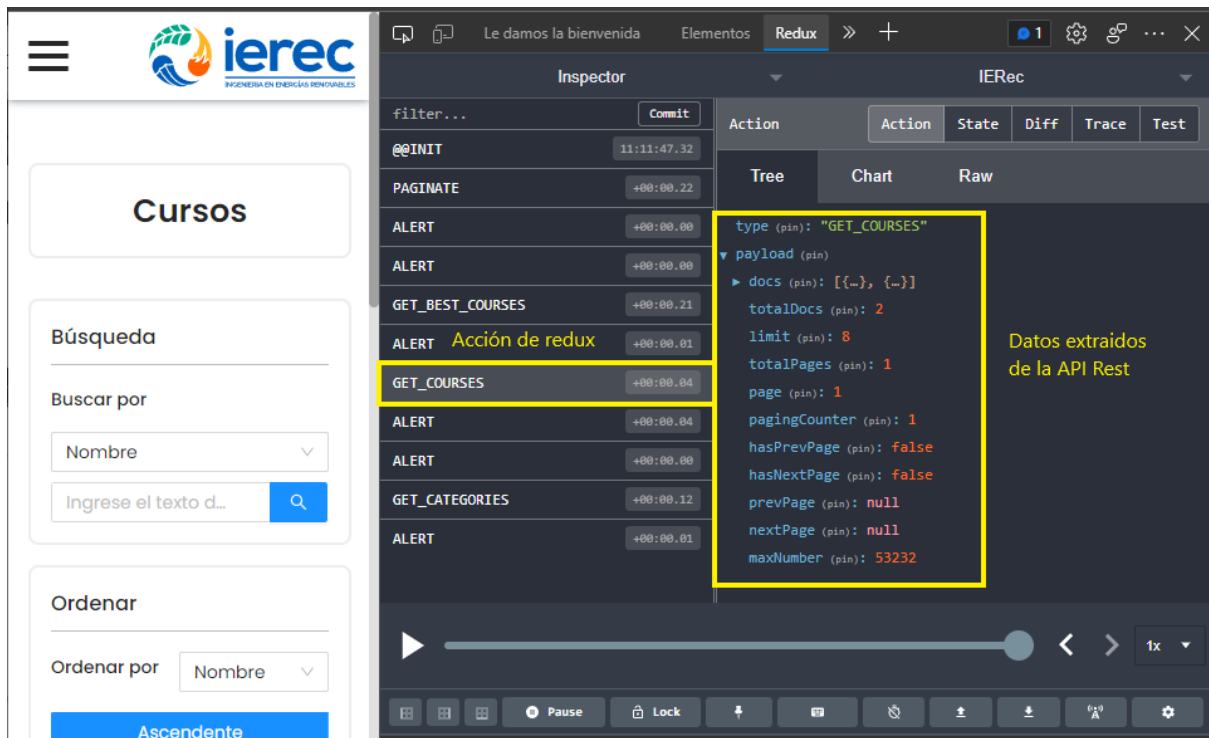


Figura 43 Herramienta para la validación de estado de React con Redux

2.4.4 Pruebas con DevTools del navegador web en el frontend

En esta etapa se realizaban pruebas de carga de contenido en una red inestable o con poca señal de internet para comprobar que la pagina tenga un rendimiento optimo y que no demore el proceso de carga del sitio web.

Para eso se debió utilizar la sección de Red de las herramientas de desarrollo del navegador y empezar a ir por cada componente refrescando la página y viendo que el contenido en una red 3G con baja carga no demore más de 20s en cargar todo el contenido, porque eso sería un problema en el performance de una página o sitio web.

A continuación, se presentará en la figura 44 como se debe utilizar la sección de red para validar la carga de contenidos en una red inestable o con poca señal de internet.

The image shows a web browser interface with a course page on the left and the developer tools network tab on the right. The course page has a header with the 'ierec' logo and a sidebar with sections for 'Cursos', 'Búsqueda', and 'Ordenar'. The developer tools network tab is open, showing a list of XHR requests. A yellow box highlights the '3G lento' network condition and the 'Tiempo y tamaño de carga de la aplicación' summary at the bottom.

Nombre	Estado	Tipo	Iniciador	Tamaño	Hora	Cascada
best	200	xhr	xhr.js:187	211 B	2.17 s	
course?page=1&limit=8	200	xhr	xhr.js:187	210 B	2.16 s	
category?page=1&limit=8	200	xhr	xhr.js:187	210 B	2.00 s	

3 / 25 solicitudes 631 B/495 kB transferido 10.1 kB /13.7 MB recursos Finalizar: 9.19 s DOMContentLoaded: 4.74 s Cargar: 4.95 s

Figura 44 Validación de carga de contenido con herramientas del navegador

CAPITULO III

VALIDACIÓN DE RESULTADOS

3.1 Metodología

La usabilidad se define como la medida de un producto o sistema puede ser evaluado por usuarios específicos para lograr objetivos concretos de manera eficiente, eficaz y satisfactoria en un contexto determinado (Mex-Alvarez et al., 2019).

La evaluación de la usabilidad se la realiza de forma heurística o empírica. Por lo cual se ha tomado la decisión de realizar una evaluación empírica debido a que este método incluye técnicas y herramientas que requieren la participación del usuario (Mex-Alvarez et al., 2019).

3.1.1 Método de evaluación

El cuestionario de Escala de Usabilidad del Sistema, o System Usability Scale (SUS) original consta de cinco preguntas positivas y cinco negativas, mediante una escala de 5 niveles que van de “Totalmente en desacuerdo” a “Totalmente de acuerdo” (Hedlefs Aguila & Garza Villegas, 2016).

En base a esto, se realizará una encuesta sobre los módulos de E-learning y evaluaciones como parte de un modelo de calidad de producto mediante Microsoft Forms que consta con un cuestionario de diez preguntas en una adaptación de este cuestionario al español de (Hedlefs Aguila & Garza Villegas, 2016) en su versión positiva, tal como recomiendan los autores de esta adaptación.

El cuestionario consta de las siguientes diez preguntas:

1. ¿Cree que me gustaría utilizar frecuentemente este sitio web?
2. ¿Encontró el sitio web sencillo?
3. ¿Piensa que el sitio web es fácil de usar?
4. ¿Piensa que podría utilizar este sitio web sin el apoyo de un técnico?
5. ¿Encontró que varias de las funciones en el sitio web estaba bien integradas?
6. ¿Piensa que había demasiada consistencia en el sitio web?

7. ¿Imagina que la mayoría de las personas podrían aprender a usar este sitio web muy rápido?
8. ¿Encontró el sitio web muy intuitivo?
9. ¿Se sintió muy confiado (seguro) al utilizar el sitio web?
10. ¿Puede utilizar el sitio web sin tener que aprender algo nuevo?

Para evaluar la encuesta se ha tomado en cuenta la escala de Likert de 5 puntos siendo 5 totalmente de acuerdo y 1 totalmente en desacuerdo (Sauro & Lewis, 2016).

1. Totalmente en desacuerdo
2. En desacuerdo
3. Ni de acuerdo ni en desacuerdo
4. De acuerdo
5. Totalmente de acuerdo

Los resultados del cuestionario se los representa en la tabla 18 de la siguiente manera, en la primera columna las opciones o posibles respuestas y de ahí hacia la derecha cada columna enumerada del 1 al 10 haciendo referencia a cada una de las preguntas descritas elaboradas en el cuestionario.

Tabla 18

Resultados de la encuesta en la plataforma Microsoft Forms

Respuesta (opciones)	1	2	3	4	5	6	7	8	9	10
Totalmente en desacuerdo	0	0	0	0	0	0	0	0	0	0
En desacuerdo	3	2	3	3	1	2	1	1	1	1
Ni de acuerdo ni en desacuerdo	7	6	6	6	6	7	7	7	7	7
De acuerdo	20	21	18	19	21	23	22	22	20	20

Totalmente de acuerdo	28	29	31	30	30	26	28	28	30	30
Total	58	58	58	58	58	58	58	58	58	58

3.2 Análisis de resultados

A continuación, se detallan los resultados obtenidos de cada pregunta de la encuesta SUS aplicada a 58 estudiantes de la carrera de Software.

Pregunta 1: ¿Cree que me gustaría utilizar frecuentemente este sitio web?

La figura 45 corresponde a la pregunta 1 y en su análisis se puede observar que la opción más votada fue “Totalmente de acuerdo” con un 48% de los votos, mientras que la segunda opción fue “De acuerdo” con un 35% de los votos. Esto demuestra que si bien existe una tendencia a que los usuarios reutilicen el sitio web, todavía hay margen de mejora para lograr mejores calificaciones.

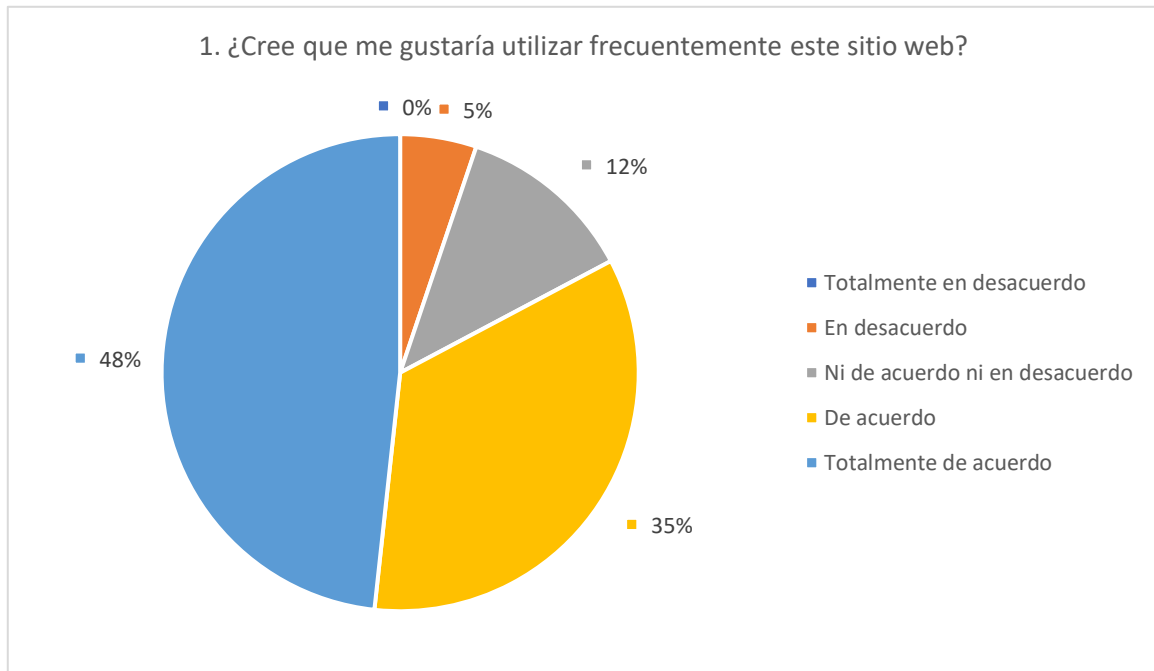


Figura 45 Resultados de la pregunta 1 de la encuesta SUS

Pregunta 2: ¿Encontró el sitio web sencillo?

La figura 46 corresponde a la pregunta 1 y en su análisis se puede observar que la opción más votada fue “Totalmente de acuerdo” con un 50% de los votos, mientras que la

segunda opción fue “De acuerdo” con un 36% de los votos. Indica que el usuario no encuentra la interfaz de usuario compleja, al punto de ser inconfundible con sus tareas.

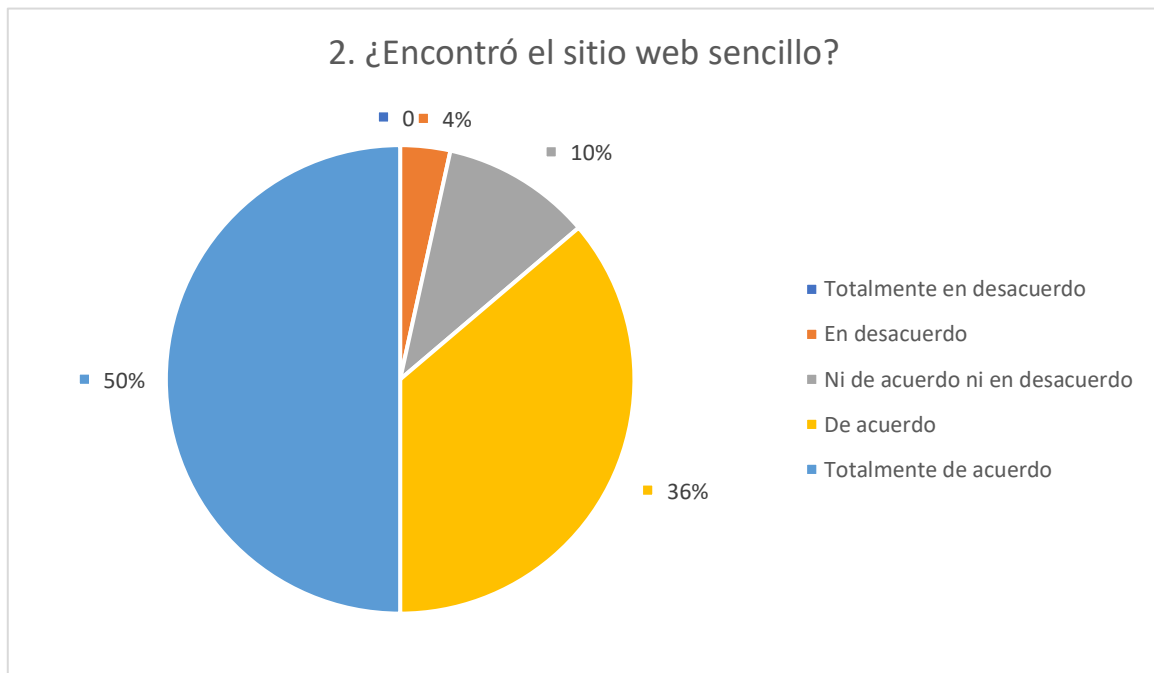


Figura 46 Resultados de la pregunta 2 de la encuesta SUS

Pregunta 3: ¿Piensa que el sitio web es fácil de usar?

La figura 47 corresponde a la pregunta 3 y en su análisis se puede observar que la opción más votada fue “Totalmente de acuerdo” con un 54% de los votos, mientras que la segunda opción fue “De acuerdo” con un 31% de los votos. Esta pregunta confirma que el modelo de interacción es utilizable incluso para los no expertos mediante la interacción con las interfaces de la aplicación.

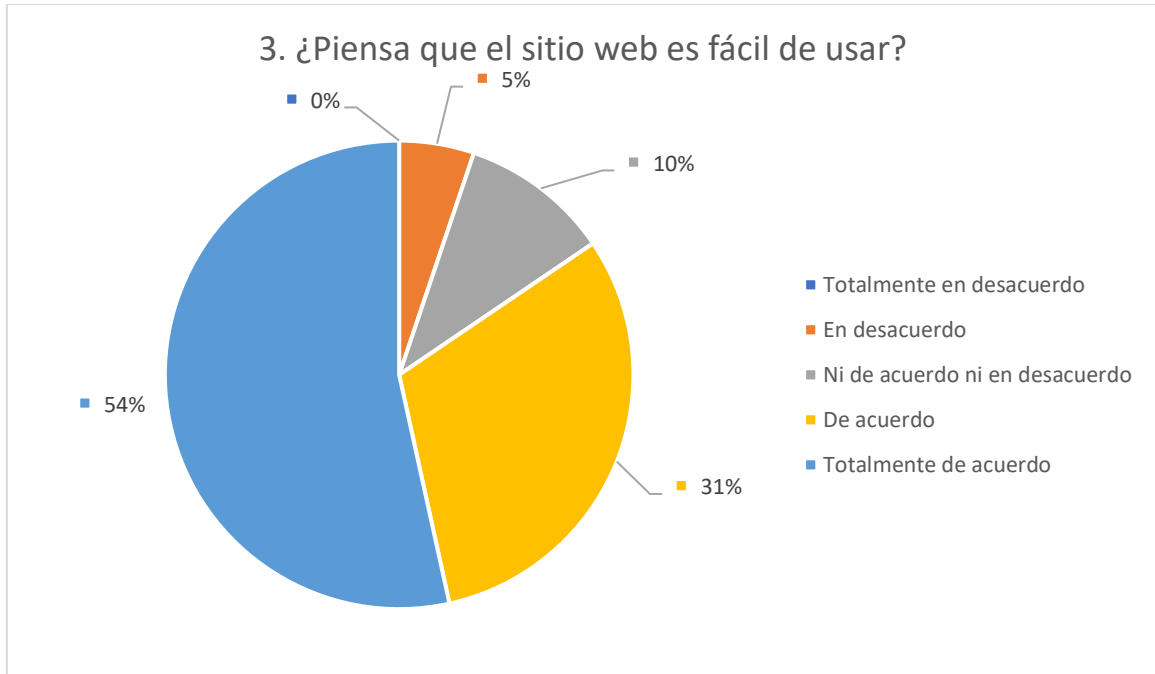


Figura 47 Resultados de la pregunta 3 de la encuesta SUS

Pregunta 4: ¿Piensa que podría utilizar este sitio web sin el apoyo de un técnico?

La figura 48 corresponde a la pregunta 4 y en su análisis se puede observar que la opción más votada fue “Totalmente de acuerdo” con un 52% de los votos, mientras que la segunda opción fue “De acuerdo” con un 33% de los votos. Los resultados muestran que los usuarios necesitan poca o ninguna ayuda para usar las interfaces de los sitios web. Una buena señal, teniendo en cuenta que se trata de usuarios no expertos.

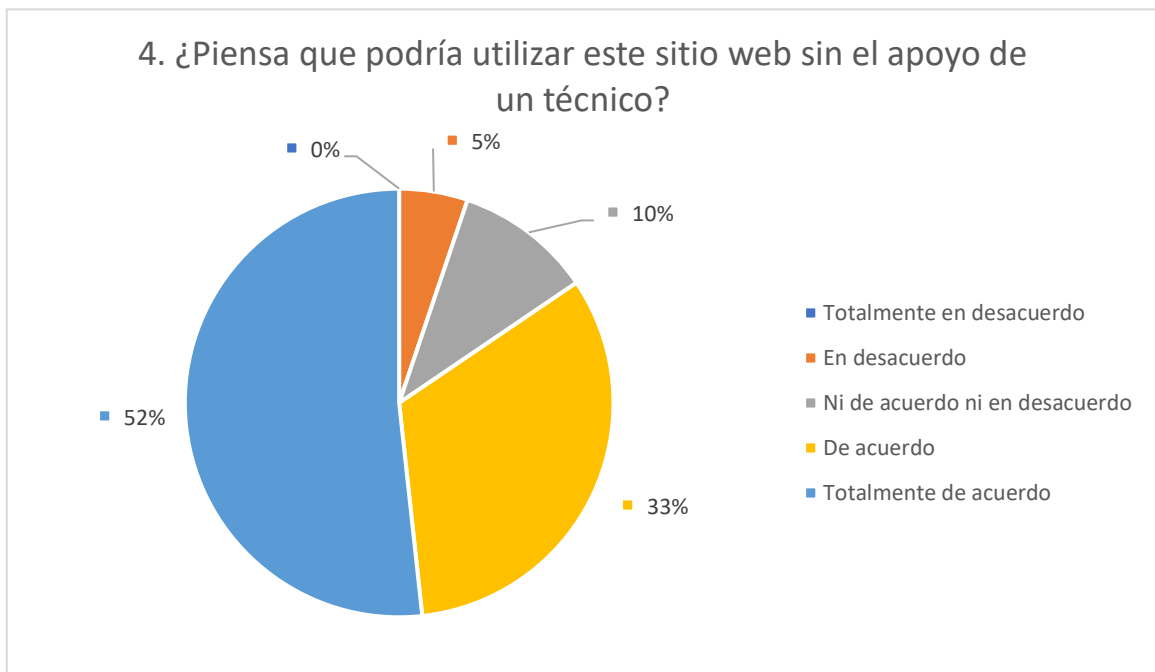


Figura 48 Resultados de la pregunta 4 de la encuesta SUS

Pregunta 5: ¿Encontró que varias de las funciones en el sitio web estaba bien integradas?

La figura 49 corresponde a la pregunta 5 y en su análisis se puede observar que la opción más votada fue “Totalmente de acuerdo” con un 52% de los votos, mientras que la segunda opción fue “De acuerdo” con un 36% de los votos. Esto demuestra que las funciones del modelo de interacción son funcionales y no hay problemas entre ellas.

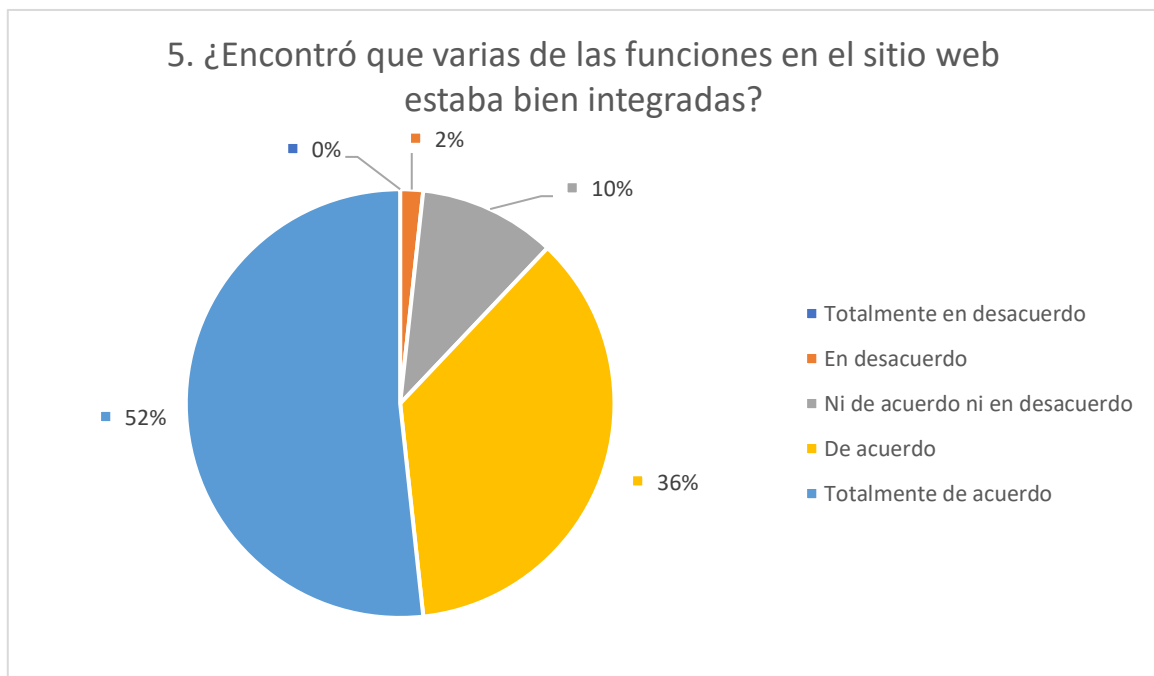


Figura 49 Resultados de la pregunta 5 de la encuesta SUS

Pregunta 6: ¿Encontró que varias de las funciones en el sitio web estaba bien integradas?

La figura 50 corresponde a la pregunta 6 y en su análisis se puede observar que la opción más votada fue “Totalmente de acuerdo” con un 45% de los votos, mientras que la segunda opción fue “De acuerdo” con un 40% de los votos. Los resultados muestran que la consistencia entre las funcionalidades e interfaces del sitio web tienen congruencia.

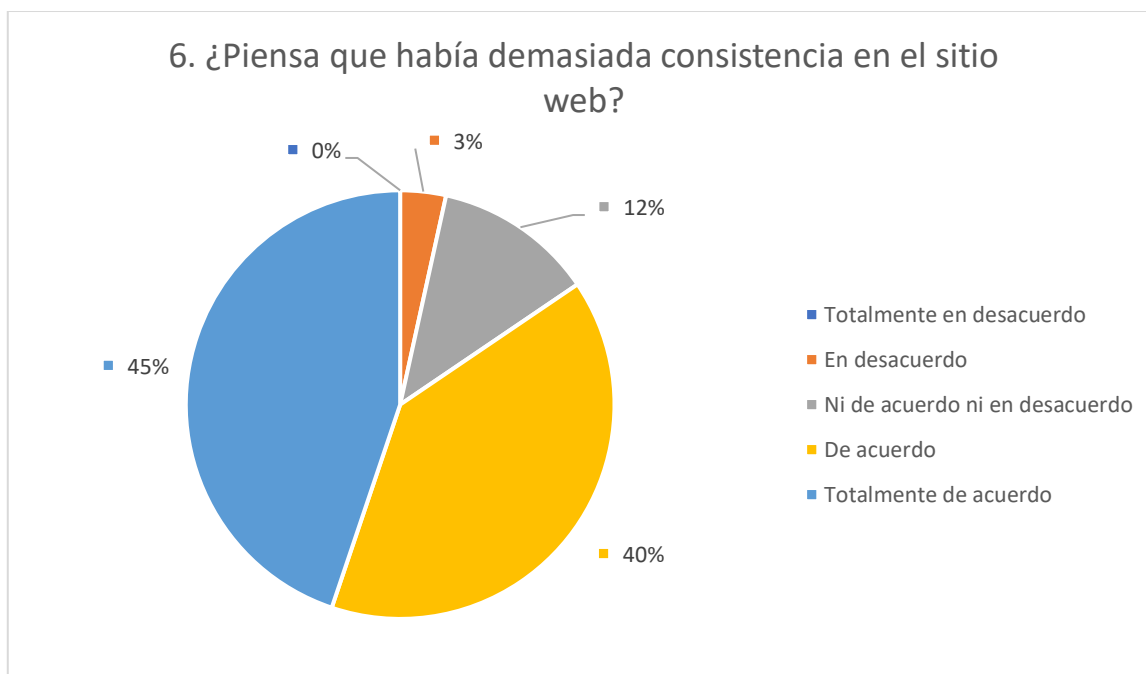


Figura 50 Resultados de la pregunta 6 de la encuesta SUS

Pregunta 7: ¿Imagina que la mayoría de las personas podrían aprender a usar este sitio web muy rápido?

La figura 51 corresponde a la pregunta 7 y en su análisis se puede observar que la opción más votada fue “Totalmente de acuerdo” con un 48% de los votos, mientras que la segunda opción fue “De acuerdo” con un 38% de los votos. Estos resultados muestran que el sitio web es ideal para aquellos que buscan cursos relacionados con la empresa IERec, ya que no necesitan un alto nivel de aprendizaje para seguir y ver todo el contenido que ofrece cada curso.

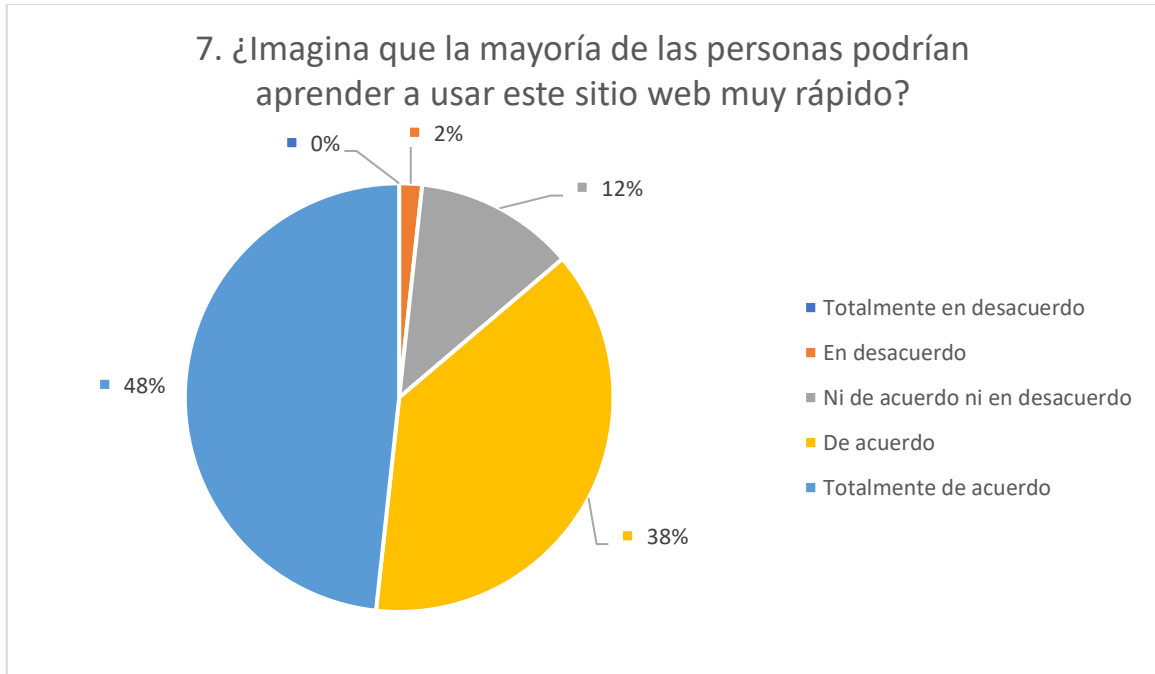


Figura 51 Resultados de la pregunta 7 de la encuesta SUS

Pregunta 8: ¿Encontró el sitio web muy intuitivo?

La figura 52 corresponde a la pregunta 8 y en su análisis se puede observar que la opción más votada fue “Totalmente de acuerdo” con un 48% de los votos, mientras que la segunda opción fue “De acuerdo” con un 38% de los votos. Según los datos presentados, el usuario muestra una percepción clara de lo que tiene que hacer simplemente manipulando el modelo de interacción, sin necesidad de un conocimiento profundo del tema.

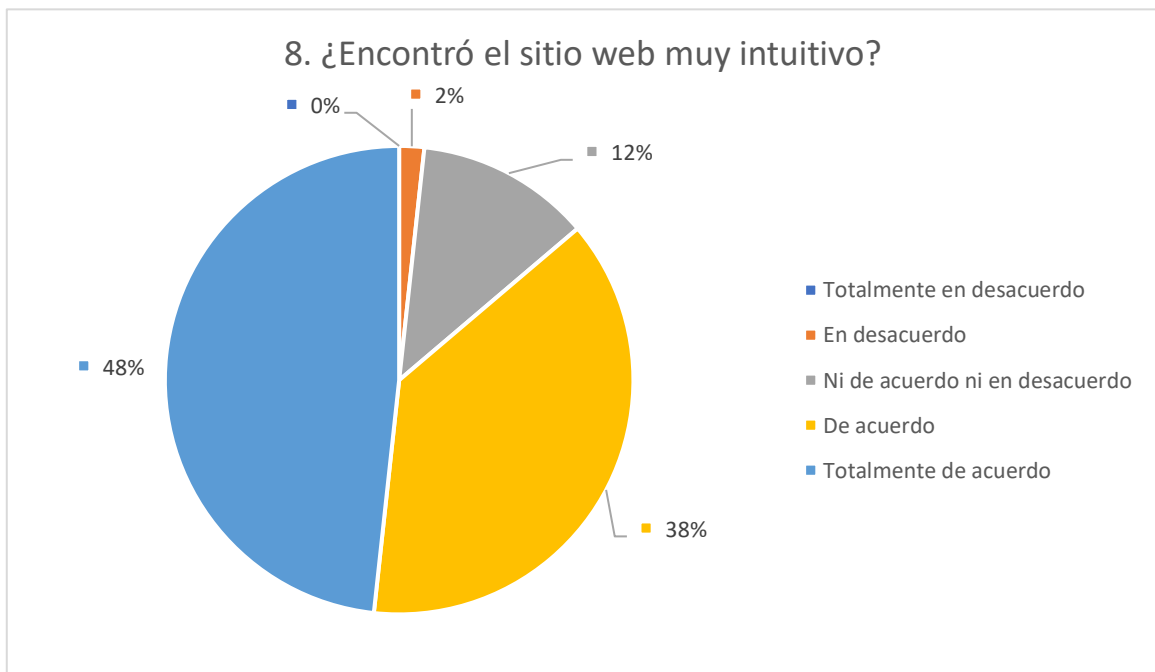


Figura 52 Resultados de la pregunta 8 de la encuesta SUS

Pregunta 9: ¿Se sintió muy confiado (seguro) al utilizar el sitio web?

La figura 53 corresponde a la pregunta 9 y en su análisis se puede observar que la opción más votada fue “Totalmente de acuerdo” con un 52% de los votos, mientras que la segunda opción fue “De acuerdo” con un 34% de los votos. Los resultados muestran que una gran cantidad de personas siente que el sitio es confiable y seguro mientras lo usan.

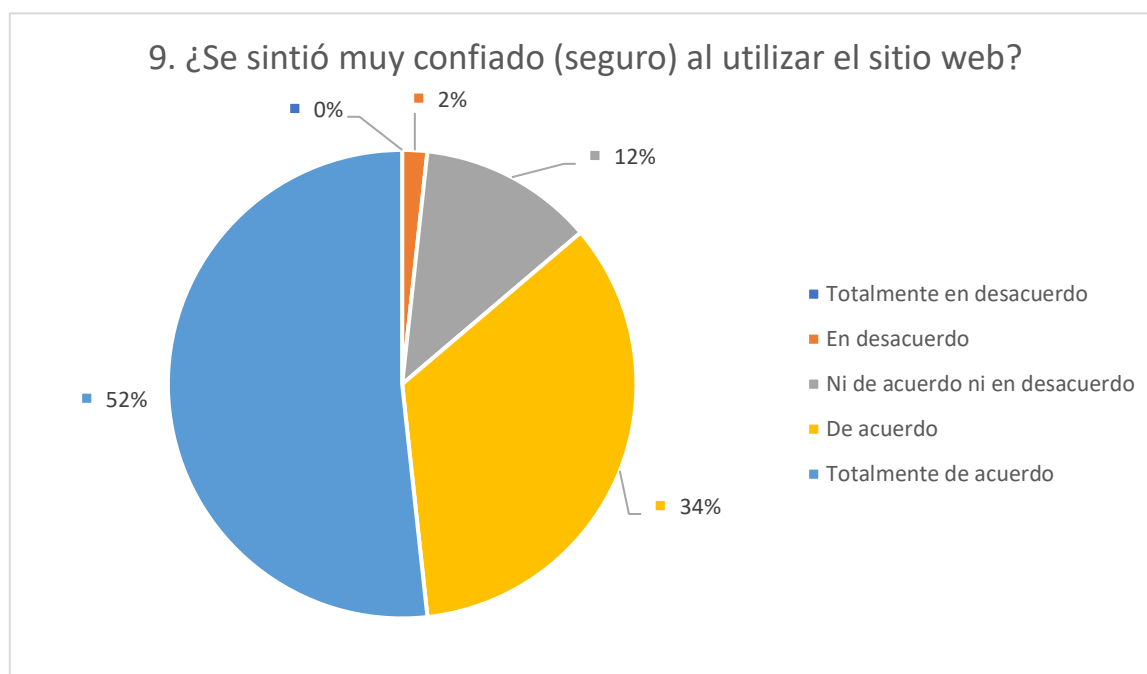


Figura 53 Resultados de la pregunta 9 de la encuesta SUS

Pregunta 10: ¿Puede utilizar el sitio web sin tener que aprender algo nuevo?

La figura 54 corresponde a la pregunta 10 y en su análisis se puede observar que la opción más votada fue “Totalmente de acuerdo” con un 52% de los votos, mientras que la segunda opción fue “De acuerdo” con un 34% de los votos. Como resultado de la encuesta, se puede deducir que el proceso de comprar y toma de cursos es fácil de seguir sin requerir mucho conocimiento, sin embargo, aún se puede mejorar para una comprensión mucho más básica de los usuarios.

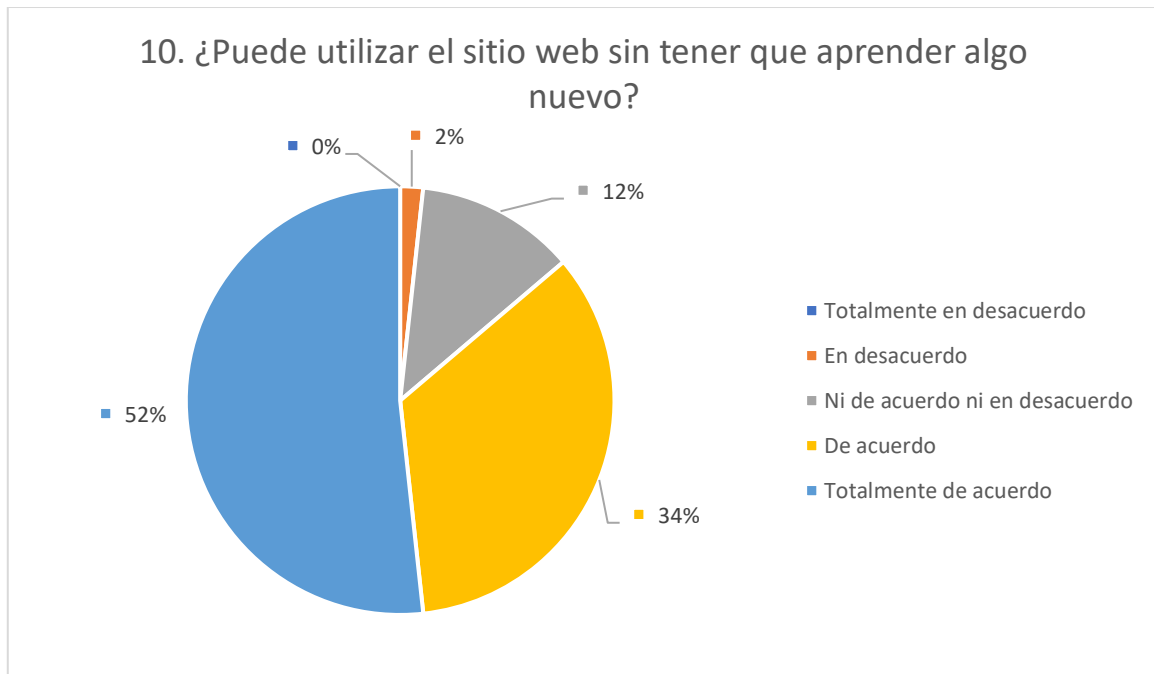


Figura 54 Resultados de la pregunta 10 de la encuesta SUS

Finalmente, en la figura 55, se muestra un resumen de los resultados de la encuesta SUS utilizada para evaluar la usabilidad del módulo de E-learning y Evaluaciones. De esta manera, se puede ver que la aceptación general de las interfaces de usuario que han cumplido su propósito de ayudar tanto a los no expertos como a los expertos a explorar de forma fácil los módulos mencionados.

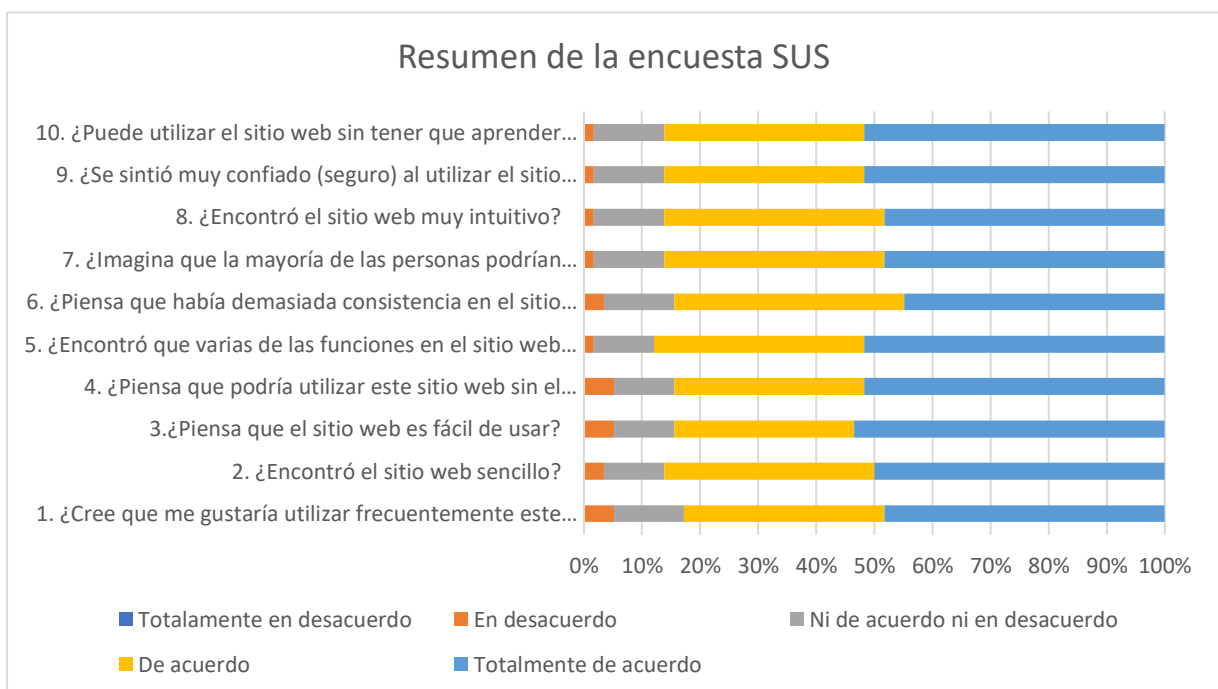


Figura 55 Resumen de la encuesta SUS aplicada a los módulos de E-learning y evaluaciones.

3.3 Interpretación de resultados

Para obtener resultados precisos del cuestionario de SUS, se debe realizar un análisis de todas las respuestas de cada uno de los encuestados. Este análisis se realiza en base a la adaptación de (Sauro & Lewis, 2016) entre las versiones positiva y negativa de la encuesta.

El método consiste en valorar cada una de las opciones del 1 al 5 y dependiendo si la pregunta es impar va del 1 al 5 y si la pregunta es par va del 5 al 1 como se presenta en la siguiente tabla 19:

Tabla 19

Valoración de cada una de las opciones de una encuesta SUS

Opciones	Preguntas Impar	Preguntas Par
Totalmente en desacuerdo	1	5
En desacuerdo	2	4
Ni de acuerdo ni en desacuerdo	3	3
De acuerdo	4	2
Totalmente de acuerdo	5	1

Una vez colocados los valores de las opciones por cada pregunta, procedemos a realizar una sumatoria de los valores pares (Y) e impares (X) de cada pregunta por serado, cómo se presenta en la siguiente tabla 20 aplicando las siguientes formulas $X = P1+P3+P5+P7+P9$ y $Y=P2+P4+P6+P8+P10$.

Tabla 20

Sumatoria de los valores de cada pregunta para 58 encuestados

P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	X	Y
----	----	----	----	----	----	----	----	----	-----	---	---

Persona 1	4	2	5	1	5	3	4	1	4	1	22	8
Persona 2	4	1	3	1	5	1	4	3	4	2	20	8
Persona 3	4	1	4	1	5	2	4	2	5	1	22	7
-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-	-
Persona 56	3	3	3	3	3	3	3	3	3	3	15	15
Persona 57	3	3	3	3	3	3	3	3	3	3	15	15
Persona 58	5	1	4	1	4	1	4	1	5	1	22	5

Ahora con la sumatoria de los números pares (Y) e impares (X) procedemos a realizar las siguientes formulas $X0 = X - 5$ y $Y0 = 25 - Y$, que ayudaran a obtener el grado SUS aplicando la siguiente formula $SUS = (X0 + Y0) * 2,5$. Una vez realizando este proceso para las respuestas de cada persona por pregunta se tiene que sacar un promedio de todos los grados SUS obtenidos como se presenta en la tabla 21.

Tabla 21

Resultados de X0, Y0, y grado SUS por persona para 58 encuestas

	X	Y	X0	Y0	SUS
Persona 1	22	8	17	17	85
Persona 2	20	8	15	17	80
Persona 3	22	7	17	18	87,5
-	-	-	-	-	-

	-	-	-	-	-	-
Persona 56	15	15	10	10	50	
Persona 57	15	15	10	10	50	
Persona 58	22	5	17	20	92,5	
Promedio					82,67	

El promedio oficial en base a las 58 encuestas realizadas a personas no expertas en el tema fue de 82,67 que según la escala de (Sauro & Lewis, 2016) representada en la figura 57, los módulos de E-learning y Evaluaciones tienen un grado A que significa que dichos módulos tienen un rendimiento superior a la media de aplicaciones web y un Adjetivo Excelente, que representa la experiencia del usuario con opciones de mejora en un futuro.

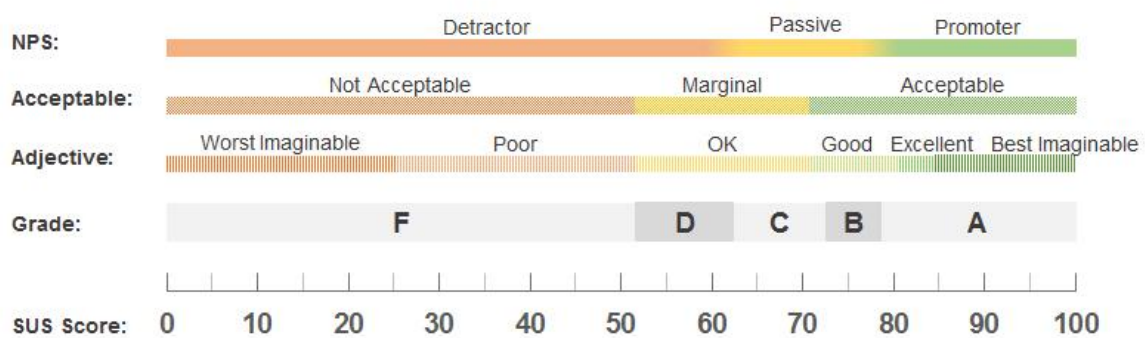


Figura 56 Grado de calificación SUS Fuente: (Sauro & Lewis, 2016)

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

El proceso de recolección de información permitió obtener sólidos conocimientos y establecer la base conceptual para definir los módulos de e-learning y evaluaciones, adaptado a las necesidades de la empresa IERec para el desarrollo y construcción de una aplicación web que mejore la oferta académica con la que cuenta la empresa.

JavaScript es un lenguaje fluido y dinámico cuando se ejecuta en el cliente (en el navegador) y Node.js no es una excepción, ya que nos permite cumplir con muchas solicitudes de usuarios en nuestro servidor. Node.js carga todos sus recursos a la vez, lo que reduce la cantidad de clics necesarios para que el usuario ejecute una funcionalidad.

Al crear un proyecto con Node.js en un servidor con altas características de hardware, los usuarios tendrán una buena experiencia y además los motores de búsqueda lo clasificarán rápidamente por la velocidad de carga del proyecto en el navegador.

Al buscar un lenguaje de programación que nos permita usarlo tanto en el cliente como en el servidor, Node.js es una de las mejores opciones porque está construido sobre el lenguaje de programación JavaScript y permite usarlo tanto en el servidor como en el cliente, haciendo que la integración entre los dos entornos sea más rápida y flexible.

Para llegar a un análisis preciso de los resultados, se debe tener en cuenta metodologías de validación de software de varios autores y escoger una que se adapte al modelo de trabajo, para evitar generar errores o falsos resultados.

Recomendaciones

Además de JavaScript y TypeScript, se sigue investigando otros posibles lenguajes que ayuden al desarrollo backend como pueden ser Python, Java, PHP, C#, GO o Rust que son alternativas que permiten la construcción de APIs REST.

React es la librería por excelencia del desarrollo frontend, pero se sugieren otras posibles alternativas conocidas como pueden ser Angular, Vue.js, Backbone.js, Preact, Svelte que ayudan a la creación de aplicación SPA y PWA como por ejemplo Facebook, Instagram o Whatsapp.

Se sigue tomando en cuenta el uso de un framework CSS, que permita acelerar el desarrollo de las páginas web, facilitando el trabajo en equipo y garantizando que todo lo que se desarrolle sea compatible con los navegadores.

Digital Ocean es una excelente plataforma que brinda servicios en la nube para el despliegue a producción de aplicaciones web, pero se recomienda tomar en cuenta otra serie de plataformas como pueden ser Amazon Web Service, Azure, Google Cloud, Heroku, Vercel, entre otros como posibles medios de alojamiento en la nube.

BIBLIOGRAFÍA

- Cerebriti. (2020). *Juegos de inteligencia, educativos y culturales*. Cerebriti.
<https://www.cerebriti.com/>
- Demartini, C., & Benussi, L. (2017). Do Web 4.0 and Industry 4.0 Imply Education X.0? In *IT Professional* (Vol. 19, Issue 3, pp. 4–7). IEEE Computer Society.
<https://doi.org/10.1109/MITP.2017.47>
- Digital Ocean. (2020). *DigitalOcean – The developer cloud*. <https://www.digitalocean.com/>
- Direccionamiento de Express. (2018). <https://expressjs.com/es/guide/routing.html>
- ExpressJS. (2020). *Express - Infraestructura de aplicaciones web Node.js*.
<https://expressjs.com/es/>
- git. (2018). *Git - Acerca del control de versiones*. Git. <https://git-scm.com/book/es/v2/Inicio---Sobre-el-Control-de-Versiones-Acerca-del-Control-de-Versiones>
- Git Inc. (2020). *Git - What is Git?* <https://git-scm.com/book/en/v2/Getting-Started-What-is-Git%3F>
- GitHub. (2020). *GitHub*. <https://github.com/>
- Guzmán Valle Alma Máter del Magisterio Nacional FACULTAD CIENCIAS, E. de, & Pérez, M. (2019). *UNIVERSIDAD NACIONAL DE EDUCACIÓN E-LEARNING. Conceptos, Importancia de la enseñanza E-learning, Ventajas y desventajas, principales herramientas de la enseñanza E-learning, principales aplicaciones*.
- Hedlefs Aguila, & Garza Villegas. (2016). *Adaptación al español del Cuestionario de Usabilidad de Sistemas Informáticos CSUQ*.
<https://www.reci.org.mx/index.php/reci/article/view/35/116>
- How DevOps Improves the 4 Phases of an Application Lifecycle | Inova Solutions*. (2020).
https://www.inovacorporation.com/blog/adoption-and-training/how-devops-improves-the-4-phases-of-an-application-lifecycle?gclid=CjwKCAjw7diEBhB-EiwAskVi14YPC7f6aupPLldP-eLDXmuGSxMMwR4hfYoBqOxozlpFmDrGXO6ISxoCOTgQAvD_BwE

- Hykes, S. (2016). *Empowering App Development for Developers | Docker*. Docker, Inc.
<https://www.docker.com/>
- ISO 25010. (n.d.). Retrieved July 28, 2021, from <https://iso25000.com/index.php/normas-iso-25000/iso-25010>
- JS Foundation and other contributors. (2019). *Getting Started with ESLint - ESLint - Pluggable JavaScript linter*. <https://eslint.org/docs/user-guide/getting-started>
- Kahoot. (2020). *Kahoot! Learning Games & Make Learning Awesome!*
https://kahoot.com/?utm_name=controller_app&utm_source=controller&utm_campaign=controller_app&utm_medium=link
- Meléndez Valladares, S. M., & Pérez Reyes, Neldin Noel Elizabeth Gaitan, M. (2016). *Metodología ágil de desarrollo de software programación extrema*. 146.
<http://repositorio.unan.edu.ni/1365/1/62161.pdf>
- Mex-Alvarez, D. C., Hernández-Cruz, L. M., Uc-Rios, C. E., & Cab-Chan, J. R. (2019). Análisis de usabilidad web a través de métricas estandarizadas y su aplicación práctica en la plataforma SAEFI. *Revista de Tecnologías Computacionales*, 15–24.
<https://doi.org/10.35429/joct.2019.9.3.15.24>
- Microsoft. (2016). What is cloud computing? A beginner's guide| Microsoft Azure. In *Cloud computing* (p. 1). <https://azure.microsoft.com/en-us/overview/what-is-cloud-computing/>
- Moodle.org. (2018). *Moodle - Open-source learning platform | Moodle.org*. Moodle.Org.
<https://moodle.org/?lang=es>
- Node.js. (2020). *Documentación | Node.js*. <https://nodejs.org/es/docs/>
- ONU. (2015). *Infraestructura – Desarrollo Sostenible*. 9. Industria, Innovación e Infraestructuras. <https://www.un.org/sustainabledevelopment/es/infrastructure/>
- OpenJs Foundation. (2020). *Acerca | Node.js*. <https://nodejs.org/es/about/>
- Plickers. (2019). <https://get.plickers.com/>
- ¿Qué es MongoDB? | MongoDB. (2019). ¿Qué Es MongoDB? | MongoDB.
<https://www.mongodb.com/es/what-is-mongodb>

- Quizizz. (2020). *Quizizz — The world's most engaging learning platform*. <https://quizizz.com/>
- React. (2019). *React – Una biblioteca de JavaScript para construir interfaces de usuario*. <https://es.reactjs.org/>
- React. (2020). *Presentando Hooks – React*. <https://es.reactjs.org/docs/hooks-intro.html>
- React – Una biblioteca de JavaScript para construir interfaces de usuario*. (2017). <https://es.reactjs.org/>
- Read Me · Redux en Español*. (2020). <https://es.redux.js.org/>
- RedHat. (2019). *¿Qué es una API de REST?* RedHat. <https://www.redhat.com/es/topics/api/what-is-a-rest-api>
- Sanmarti Puig, N. (2020). *Evaluar y aprender: un unico proceso*. Ediciones Octaedro, S.L. <https://elibro.net/es/lc/utnorte/titulos/163527>
- Sauro, J., & Lewis, J. R. (2016). Standardized usability questionnaires. In *Quantifying the User Experience* (pp. 185–248). Elsevier. <https://doi.org/10.1016/b978-0-12-802308-2.00008-4>
- Srinivasa, K. G., & Muppalla, A. K. (2015). Getting Started with Hadoop. In *Redux - Introduction* (Issue June, pp. 33–72). https://doi.org/10.1007/978-3-319-13497-0_2
- Trivinet.com - Juego de trivial online educativo*. (2020). <https://www.trivinet.com/es/trivial-online/version-web>
- Typescript: The starting point for learning Typescript*. (2019). Microsoft. <https://www.typescriptlang.org/docs/>
- TypeScript: Typed JavaScript at Any Scale*. (2020). <https://www.typescriptlang.org/>
- UNESCO. (2019). *Liderar el ODS 4 - Educación 2030*.
- Vu, H. D. (2016). COMPONENT-ORIENTED PROGRAMMING: OBJECTED-ORIENTED AND BEYOND. *International Journal of Scientific Research and Innovative Technology*, 3(11), 2313–3759.

Web Desing and Development. (2020). https://uietkuk.ac.in/training_placement_cell/wp-content/uploads/2018/10/Report-1.pdf

Welcome to the MongoDB Documentation — MongoDB Documentation. (2019). <https://docs.mongodb.com/>

What is Prettier? · Prettier. (2019). <https://prettier.io/docs/en/index.html>

Yasvi, M. (2019). *Review On Extreme Programming-XP*.

Замыкания - JavaScript | MDN. (2020). <https://developer.mozilla.org/es/docs/Web/JavaScript>

ANEXOS

Anexo A: Encuesta de usabilidad SUS

A continuación, se presenta la encuesta para la valoración de resultados de los módulos de E-learning y evaluaciones.

Encuesta de usabilidad para el módulo de E-learning y evaluaciones

El propósito es medir la interacción del módulo de E-learning y evaluaciones entre los usuarios y la aplicación web, verificando que los usuarios consigan sus objetivos de forma sencilla, intuitiva, agradable y segura.

Para ello se deberá evaluar mediante preguntas los siguientes factores principales de la usabilidad como son:

- **eficiencia:** el usuario satisface la necesidad de su búsqueda;
- **eficacia:** el usuario logra rápidamente alcanzar su cometido;
- **satisfacción:** el usuario se sintió agradable al navegar por el sitio web.

	Totalmente en desacuerdo	En desacuerdo	Ni de acuerdo ni en desacuerdo	De acuerdo	Totalmente de acuerdo
Cree que me gustaría utilizar frecuentemente este sitio web.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Encontró el sitio web sencillo.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Piensa que el sitio web es fácil de usar.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Piensa que podrá utilizar este sitio web sin el apoyo de personal técnico.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Encontró que varias de las funciones en el sitio web estaban bien integradas.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Piensa que había demasiada consistencia en el sitio web.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Imagina que la mayoría de las personas podrían aprender a usar este sitio web muy rápido	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Encontró el sitio web muy intuitivo.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Se sintió muy confiado (seguro) al utilizar el sitio web.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Puede utilizar el sitio web sin tener que aprender algo nuevo.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>