

04

SISTEMA WEB

**PARA LA CREACIÓN DE HORARIOS DE CLASES EN LA
UNIVERSIDAD METROPOLITANA DEL ECUADOR**

SISTEMA WEB

PARA LA CREACIÓN DE HORARIOS DE CLASES EN LA UNIVERSIDAD METROPOLITANA DEL ECUADOR

WEB SYSTEM FOR THE CREATION OF CLASS SCHEDULES AT THE METROPOLITAN UNIVERSITY OF ECUADOR

Tony sé de la Rosa-Martín¹

E-mail: tdelarosa@umet.edu.ec

ORCID: <https://orcid.org/0000-0002-0881-6034>

Jorge Luis León-González²

E-mail: jleon@umet.edu.ec

ORCID: <https://orcid.org/0000-0003-2092-4924>

¹ Universidad Metropolitana del Ecuador. Ecuador.

² Investigador Independiente. Estados Unidos.

Cita sugerida (APA, séptima edición)

De la Rosa-Martín, T., León-González, J. L. (2024). Sistema Web para la creación de horarios de clases en la Universidad Metropolitana del Ecuador. *Revista Metropolitana de Ciencias Aplicadas*, 7(S1), 38-48.

RESUMEN

La investigación se enfocó en el desarrollo de un sistema web de creación de horarios de clases para la Universidad Metropolitana del Ecuador-UMET. Para ello, en primer lugar, se determinaron los requerimientos funcionales y no funcionales del sistema, para posteriormente proceder a su desarrollo y finalmente llevar a cabo la validación de calidad de este. La metodología se enmarcó en un enfoque cuantitativo, nivel descriptivo, diseño no experimental-transversal y tipo de investigación de campo. La población y muestra estuvo constituida por 13 directores de carrera y tres (3) expertos en el área de software. La técnica de recolección de datos empleada fue la encuesta, y como instrumentos dos cuestionarios que permitieron identificar en primer lugar los requerimientos para el funcionamiento del software y se constituyó por 16 preguntas y en segundo lugar la evaluación de la calidad del sistema web que se constituyó por 10 preguntas, ambas con escala de respuesta dicotómica (Si/No). La validación se realizó a partir del juicio de tres (3) expertos y la confiabilidad quedó determinada para ambos cuestionarios por el coeficiente Kuder Richardson (KR-20); arrojando valores de 0.974 y 0.741 respectivamente reflejando una buena confiabilidad. Se obtuvo 7 (siete) requerimientos funcionales y 9 (nueve) no funcionales lo que permitió el desarrollo del sistema web concatenado con las necesidades reales obteniendo un sistema eficiente y robusto el cual su calidad fue evaluada por expertos. El sistema obtenido permitió automatizar el proceso reduciendo la carga de trabajo y tiempo empleado para crear horarios de clases.

Palabras clave:

Sistema Web, horarios, requerimientos funcionales, requerimientos no funcionales.

ABSTRACT

The research focused on the development of a web system for creating class schedules for the Metropolitan University of Ecuador-UMET. To do this, first, the functional and non-functional requirements of the system will be determined, to subsequently proceed with its development and finally carry out its quality validation. The methodology was framed in a quantitative approach, descriptive level, non-experimental-cross-sectional design and type of field research. The population and sample consisted of 13 career directors and three (3) experts in the software area. The data collection technique used was the survey, and as instruments two questionnaires that made it possible to identify, firstly, the requirements for the operation of the software and consisted of 16 questions, and secondly, the evaluation of the quality of the web system that was constituted for 10 questions, both with a dichotomous response scale (Yes/No). The validation was carried out based on the judgment of three (3) experts and the reliability was determined for both questionnaires by the Kuder Richardson coefficient (KR-20), yielding values of 0.974 and 0.741 respectively, reflecting good reliability. 7 (seven) functional requirements and 9 (nine) non-functional requirements were obtained, which allowed the development of the web system concatenated with the real needs, obtaining an efficient and robust system whose quality was evaluated by experts. The system obtained made it possible to automate the process, reducing the workload and time used to create class schedules.

Keywords:

Web system, schedules, functional requirements, non-functional requirements.

INTRODUCCIÓN

En la actualidad, los sistemas web se han vuelto cada vez más populares y utilizados en todo el mundo. La facilidad que estos sistemas ofrecen a los usuarios para acceder a información y realizar tareas en línea ha llevado a su amplia aceptación en la sociedad. En particular, en el ámbito educativo, el uso de sistemas web se ha vuelto cada vez más relevante, sobre todo en las universidades, donde la cantidad de estudiantes y profesores que interactúan en línea es cada vez mayor. En el Ecuador, muchas universidades todavía utilizan métodos tradicionales como papel y lápiz para crear los horarios de clases. Además, algunas de estas instituciones utilizan herramientas ofimáticas que generan un gran costo debido al pago de licencias de software. Esta situación ha llevado a que muchos de estos sistemas sean ambiguos y propensos a errores, lo que puede generar confusiones y retrasos en el inicio de las clases. En este contexto, la Universidad Iberoamericana del Ecuador (UMET) se enfrenta a desafíos similares en la creación de horarios de clases. A pesar de los esfuerzos de los directores de carrera, el proceso manual utilizado en la creación de horarios puede ser propenso a errores humanos y a la asignación incorrecta de aulas y laboratorios.

Para solucionar estos problemas, la UMET está buscando desarrollar un sistema web que automatice el proceso de creación de horarios de clases.

Este sistema web permitiría una mayor eficiencia en la creación de horarios, lo que podría reducir los errores humanos y evitar la asignación incorrecta de aulas y laboratorios.

El objetivo del desarrollo de este sistema web es mejorar la eficiencia y creación de los horarios de en la UMET. Además, dicho sistema web también podría mejorar la calidad de vida estudiantil y del personal académico.

Para llevar a cabo el desarrollo, se realizó una investigación que utilizó un enfoque cuantitativo y la herramienta para recopilar datos será la encuesta a expertos en el área de software de la UMET.

La tecnología web ha tenido un impacto revolucionario la sociedad, cambiando por completo la forma en que las personas se relacionan, trabajan y desenvuelven. En el ámbito educativo, se ha observado su poder transformador. La constante evolución de las herramientas y plataformas web ha facilitado el acceso a la información, la comunicación instantánea y la optimización de procesos, generando un impacto profundo en la vida diaria.

Por lo tanto, un sistema web se define como aquel que tiene servicios, tanto en contenido como en funcionalidad, que se ofrecerán mediante la web. El diseño de componentes se centra en los objetos de esta y en cómo se puede mostrar la interfaz al usuario final. Además, el diseño funcional del sistema web se enfoca en la manipulación

de los objetos, realización de cálculos y consulta en base de datos. Asimismo, el sistema web tiene un diseño a nivel de componentes. Por esta razón, se debe tener en cuenta que los sistemas y aplicaciones basados en la web residen en una web e interoperan con diferentes tipos de navegadores y plataformas de hardware (Pressman & Maxim, 2019).

También, los sistemas web son aquellos que emplean la infraestructura de Internet para brindar servicios y funcionalidades a los usuarios. Se crean especialmente para funcionar a través de la web, utilizando tecnologías como el protocolo HTTP, los navegadores web y los servidores. Estos sistemas proporcionan una variedad extensa de servicios y funcionalidades, desde páginas web estáticas simples hasta aplicaciones complejas y dinámicas que permiten interacciones avanzadas con los usuarios (Sommerville, 2019)

De acuerdo con lo antes mencionado, el sistema web se puede definir como una aplicación que permite la interacción fácil para realizar una tarea en específico con la conexión de un servidor web, permitiendo la disponibilidad de la información en cualquier plataforma, en la que se tenga un navegador y así obtener la información o contenido que se solicita en tiempo real.

Según la Real Academia Española (2021), un horario se define como *“un cuadro indicador de las horas en que deben ejecutarse determinadas actividades”*.

Por su parte, un horario de clase es la planificación de las actividades educativas que se realizan en diferentes días a la semana para llevarse a cabo durante la jornada académica. Es importante que, el horario de clases tenga una visión de la planificación que permita a los estudiantes variar entre diferentes tipos de actividades y descansos que se ajusten a sus necesidades (Gobierno de España, 2022)

En este sentido, puede decirse que toda institución educativa debe tener procedimientos establecidos para la elaboración de horarios de manera rápida y oportuna, siendo lo más recomendado contar con programas automatizados. Sin embargo, a nivel nacional, en el ámbito educativo se encuentra una falta de implementación tecnológica para la planificación y estructuración de los horarios de clases, estos procesos se realizan de manera manual, lo que conlleva a errores y retrasos en las labores académicas (Baquerizo, 2021).

De esta situación no escapa la Universidad Iberoamericana del Ecuador, ya que en ella los horarios de clase se realizan de forma manual en Excel. Esta metodología incide en la gran cantidad de tiempo que deben emplear los directores de carrera y la directora Académica para realizar los horarios de esta manera, lo cual puede ocasionar diversas fallas al crear los referidos horarios de clases que pueden llevar a colisiones entre horarios y aulas, así como retrasos en el inicio del semestre.

Además, esta forma manual de gestionar los horarios presenta otros problemas que afectan ampliamente a la universidad. Por un lado, la dependencia de hojas de cálculo como Excel implica una mayor posibilidad de errores humanos al ingresar y manipular la información, generando situaciones de sobreocupación de aulas o superposición de clases.

Por otro lado, la falta de flexibilidad es una limitación significativa en este proceso. Los cambios de última hora, como la necesidad de agregar una nueva asignatura, modificar el horario de una clase o ajustar la disponibilidad de un aula, se vuelven complicados y requieren un esfuerzo considerable debido a la naturaleza estática de los documentos en Excel. Esto puede ocasionar retrasos en la comunicación de cambios, lo que afecta la planificación y organización de todos los involucrados.

En cuanto al proceso de creación de horarios de clase, se utiliza la malla curricular híbrida-presencial y la malla curricular semipresencial nocturna, las cuales abarcan desde el primer hasta el octavo nivel. Cada carrera cuenta con un tronco común que une las materias, y es responsabilidad de los directores de carrera crear los horarios de clase asignando las asignaturas correspondientes a cada una. Durante este proceso, es importante garantizar una asignación adecuada de espacios físicos y virtuales para el correcto desarrollo de las clases.

Con respecto al resumen de profesores, se generan las horas y lugares donde deben dictar sus clases. Una vez finalizados los documentos, los directores de carrera envían el horario y los resúmenes de profesores a la Dirección Académica para su validación.

El objetivo del estudio fue implementar un sistema web para la creación de horarios de clases en la Universidad Metropolitana del Ecuador (UMET).

MATERIALES Y MÉTODOS

Para llevar a cabo un marco de trabajo y organizar los flujos de manera ágil, se debe tener en cuenta que *“las metodologías ágiles se centran en el factor humano y el producto software, es decir, ellas le dan mayor valor al individuo, a la colaboración del cliente y al desarrollo incremental del software con iteraciones muy cortas”* (Orjuela & Rojas, 2008, p. 159), lo cual demuestra que, en un proyecto da la opción de permitir adaptarse y resolver las etapas en poco tiempo.

En este sentido, Scrum es una técnica de la metodología ágil en la cual se trabaja de forma donde los equipos evalúan el progreso como lo hicieron y el modo que lo han hecho, además se organizan de la manera en que trabajan, proporcionando herramientas y mejorando rápidamente tanto la velocidad como la calidad del trabajo (Sutherland, 2014).

Scrum establece tres roles, el Scrum Master, el Product owner o dueño del producto, y el equipo de desarrollo o

DevelopmentTeam, el Scrum master desempeña el papel de liderar al equipo y garantizar el cumplimiento de las reglas y procesos de la metodología. El Product owner actúa como el representante de los accionistas y clientes que utilizan el software. Por último, el equipo de desarrollo está compuesto por un grupo de profesionales cuya responsabilidad es transformar la lista de requerimientos en funcionalidades concretas del software (Molina, Vite, & Dávila, 2018).

Artefactos SCRUM

De acuerdo con Nazareno et al. (2013) es *“un artefacto es la descripción de un producto de trabajo. Los artefactos pueden estar compuestos por otros artefactos y son productos de trabajos concretos consumidos, producidos o modificados por los distintos eventos del proceso. Se representan los conceptos vinculados a los artefactos del proceso”*. (p. 924)

Lo que demuestra que un artefacto es la descripción de un producto de trabajo en el cual se muestran los conceptos de los artefactos.

Para los artefactos SCRUM es importante tener en cuenta que *“la generación del producto principal está guiada por un conjunto ordenado de requerimientos denominado ProductBacklog. El product backlog contiene cada 17 requerimiento (ProductBacklogItem) que podría ser tratado en el desarrollo de un producto (Product), como así también en cada incremento del producto (ProductIncrement). Habitualmente la lista de ProductBacklogItem se encuentra ordenada por valor, riesgo, prioridad y necesidad”*. (Nazareno et al., 2013)

Por lo cual, los artefactos SCRUM están constituidos por el ProductBacklog, el ProductIncrement que se refiere al incremento del producto y por último el ProductBacklogItem.

Arquitectura del sistema

Para el desarrollo del sistema web, se implementará el patrón MVC (Modelo, Vista, Controlador) que según Huanca (2020), lo define como *“el MVC es un patrón de diseño arquitectónico de software, que sirve para clasificar la información, la lógica del sistema y la interfaz que se le presenta al usuario. En este tipo de arquitectura existe un sistema central o controlador que gestiona las entradas y la salida del sistema, uno o varios modelos que se encargan de buscar los datos e información necesaria y una interfaz que muestra los resultados al usuario final. Es muy usado en el desarrollo web porque al tener que interactuar varios lenguajes para crear un sitio es muy fácil generar confusión entre cada componente si estos no son separados de la forma adecuada. Este patrón permite modificar cada uno de sus componentes si necesidad de afectar a los demás”*. (p. 22)

Por lo tanto, el patrón MVC es una valiosa herramienta de desarrollo de software al proporcionar una arquitectura

bien definida debido a la adecuada separación de los componentes y asegurando una estructura eficiente, organizada y escalable.

Los componentes que conforman MVC son los siguientes:

El modelo es responsable de la manipulación, gestión y actualización de los datos, especialmente cuando se utiliza una base de datos. En este componente, se llevan a cabo las consultas, búsquedas, filtros y actualizaciones de información (Huanca, 2020).

La vista es el componente encargado de presentar al usuario final las pantallas, ventanas, páginas y formularios que resultan de una solicitud. Desde la perspectiva del programador, este componente se encarga del front-end, es decir, de programar la interfaz de usuario en caso de una aplicación de escritorio, o de la visualización de las páginas web (Huanca, 2020).

El controlador desempeña la función de administrar las instrucciones recibidas, atenderlas y procesarlas. Es el componente encargado de facilitar la comunicación entre el modelo y la vista. Su tarea principal implica solicitar los datos necesarios, manipularlos para obtener los resultados deseados y luego enviarlos a la vista para su visualización (Huanca, 2020).

Por consiguiente, al implementar el patrón MVC en el sistema web de creación de horarios, se obtiene una estructura clara y organizada.

Tecnologías de desarrollo web

Para el desarrollo de este sistema web, es crucial tener en cuenta el back-end definido por Mestres (2018), como *“aquello que se encuentra del lado del servidor y se encarga de interactuar con bases de datos, verificar manio- bras de sesiones de usuarios, montar la página en un ser- vidor y servir todas las vistas creadas por el desarrollador frontend”*.

Por lo tanto, es la parte del sistema web que se encarga de la lógica, datos y la funcionalidad.

Además, el Backend se complementa con el front-end, que según Mestres (2018) lo define como *“aquellas tec- nologías de desarrollo web del lado del cliente, es decir, las que corren en el navegador del usuario y que son básicamente tres: HTML, CSS y JavaScript”* (p. 5). Esto implica que el front-end se encarga de la interfaz gráfica del sistema web, facilitando la interacción con el usuario.

Asimismo, esta tecnología se relaciona con el uso de un frameworks que es un modelo de un dominio determina- do o algún aspecto importante del mismo. Los framework- s modelan cualquier dominio, puede ser un dominio técnico como la distribución de basura, o un dominio de aplicación como la banca o los seguros, Adicional nos

proporciona un diseño e implementaciones reutilizables a los clientes (Riehle, 2000). Se puede afirmar que un fra- mework actúa como el esqueleto del sistema web y sim- plifica el diseño, lo que a su vez permite reducir el tiempo de diseño e implementación mediante la reutilización del diseño y código.

Adicionalmente, es importante destacar que los framework- s presentan algunas características relevantes como la abstracción de URLs y sesiones que se encarga los fra- meworks, la no manipulamos directamente, el Acceso de datos que incluyen las herramientas para la integración de los datos, los controladores que nos facilita para la gestión de eventos, como él envió de datos mediante el formulario o el acceso a una página, siendo fáciles de adaptar al sistema y finalmente tenemos la autenticación y control que nos incluye varios mecanismos para la iden- tificación de usuarios (Gutiérrez, 2006).

Para el desarrollo del sistema web se utilizará el fra- mework Angular que es un marco de diseño de aplica- ción y una plataforma para el desarrollo de sistemas web escalables, eficientes y sofisticadas, además contiene una amplia colección de bibliotecas integradas que in- cluyen el enrutamiento, la administración de formulario, la comunicación de cliente/servidor y más (Angular, 2022).

Herramientas de modelado

En el proceso de desarrollo del software, se emplean he- rramientas de modelado entre las cuales se destaca el Lenguaje Unificado de Modelado (UML), el cual se define como *“una de las herramientas más emocionantes en el mundo actual del desarrollo de sistemas. Esto se debe a que permite a los creadores de sistemas generar diseños que capturen sus ideas en una forma convencional y fá- cil de comprender para comunicarlas a otras personas... UML se puede usar para modelar distintos tipos de siste- mas: sistemas de software, sistemas de hardware y orga- nizaciones del mundo real”* (Auz, 2016, p. 27)

Lo que demuestra que esta herramienta es muy útil en la modelación del sistema destacando su capacidad para ser fácilmente comprendida por los usuarios y clientes sin conocimientos en el campo.

Base de datos

Para la gestión de los datos del sistema web se nece- sita una base de datos que según Marqués (2011), *“es un conjunto de datos almacenados en memoria externa que están organizados mediante una estructura de datos. Cada base de datos ha sido diseñada para satisfacer los requisitos de información de una empresa u otro tipo de organización”* (p. 2). Por lo cual se convierte en un ele- mento esencial para gestionar los datos eficientemente dentro del presente sistema web.

Es importante implementar una base de datos al diseñar el sistema web, y para ello se empleará MongoDB que *“es una base de datos NoSQL ampliamente utilizada, de propósito general, orientada a documentos, con funciones que incluyen replicación de alta disponibilidad y fragmentación automática para escala horizontal”* (Morejón, 2018, p. 59). Además, la elección de MongoDB como base de datos permite aprovechar sus ventajas en cuanto a escalabilidad.

Entorno de desarrollo

En la fase de desarrollo del sistema web, también se tiene en cuenta la incorporación de un entorno de desarrollo integrado conocido como “IDE”. En este sentido, se ha seleccionado Visual Studio Code debido a que *“nos brinda la reutilización de código para crear nuestros propios constructores, funciones, clases etc., integrando la implementación y diseño de formularios de Windows utilizado en la construcción de software de gestión y automatización de procesos debido a que resulta fácil encontrar fuentes, información y documentación para construir proyectos”*. (Serrano, 2019, p. 55)

En resumen, la utilización de un IDE como Visual Studio Code permite un desarrollo más efectivo del sistema web, gracias a los beneficios que ofrece.

RESULTADOS Y DISCUSIÓN

Modelo de procesos

Se elaboró el artefacto diagrama de casos de uso para establecer el orden de las acciones del flujo del sistema tal como se muestra en la Figura 1, 2 y 3.

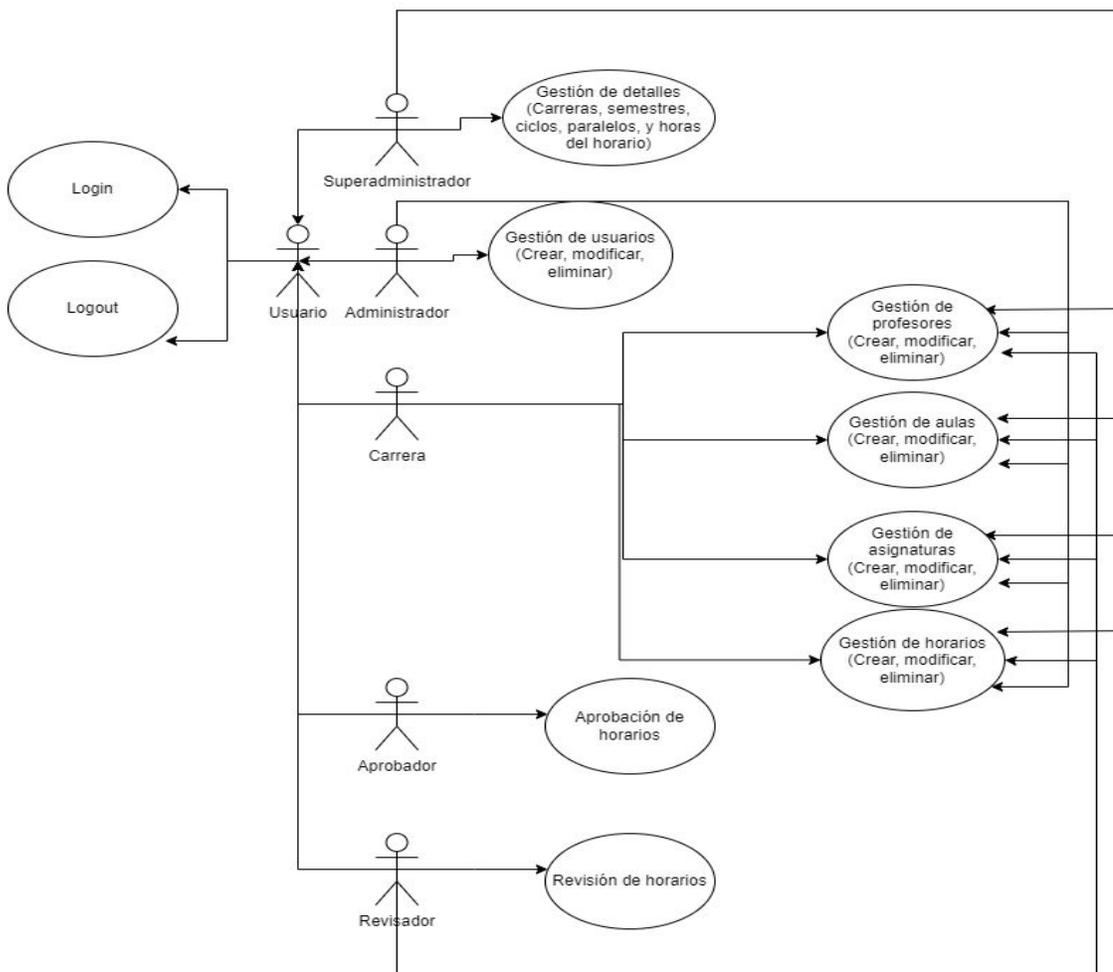


Figura 1. Diagrama de Casos de Uso.

Diagrama base de datos no relacional NoSQL Entidad-relación

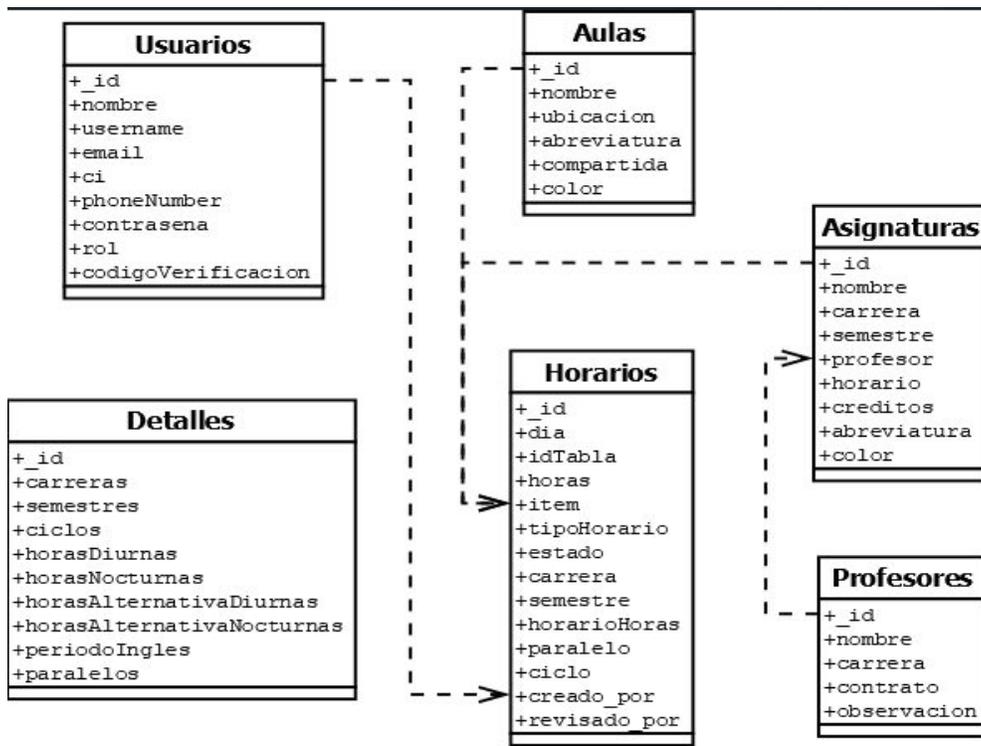


Figura 2. Modelo de bases de datos no relacionales.

Diagrama BPMN

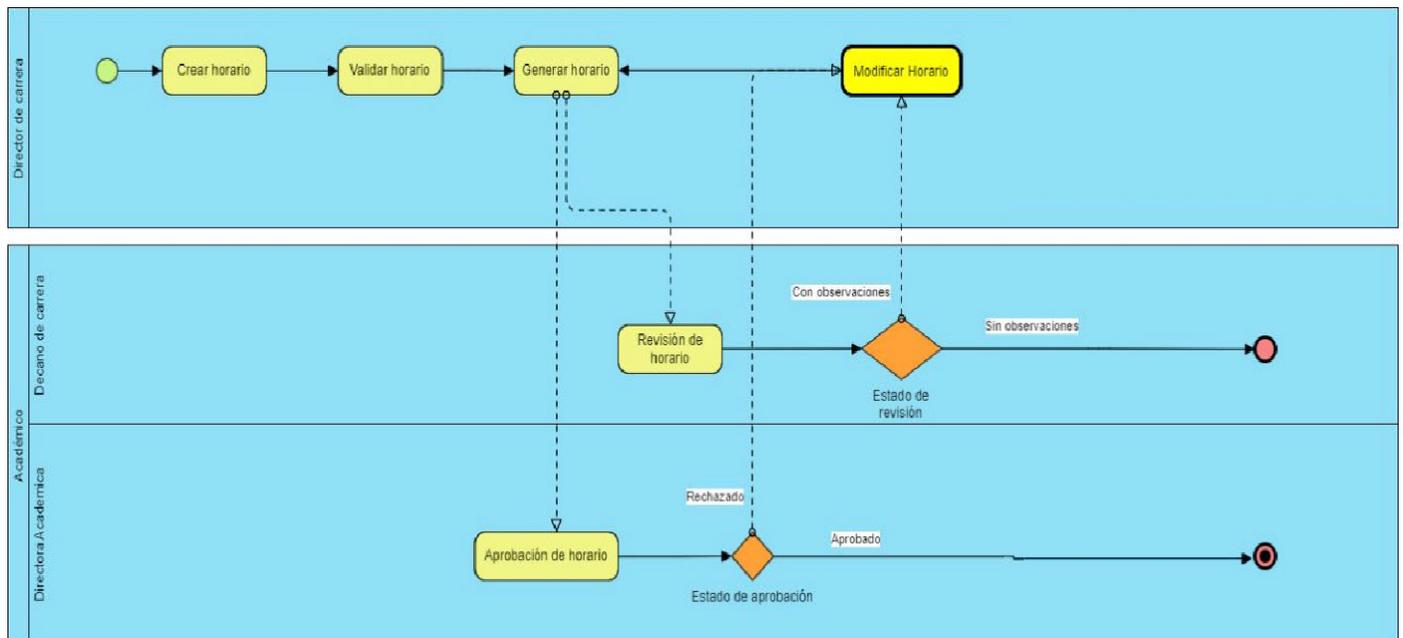


Figura 3. Modelo BPMN.

Diseño de interfaz de usuario

Para el desarrollo del sistema web es importante diseñar una interfaz de usuario debido a que es importante planificar la creación de pantallas necesarias y roles que tendrá el sistema web para que el usuario final pueda utilizarlo de manera fácil y sencilla, para el diseño la interfaz de usuario.

Los roles que existen en el diseño de la interfaz son los siguientes:

Superadministrador tiene acceso a todas las funciones del sistema.

Administrador tiene acceso a todas las funciones del sistema, excepto la modificación de los detalles del horario (horas, carreras, ciclos y paralelos).

Usuario Aprobador tiene acceso a todas las funcionalidades, excepto la modificación de los detalles del horario y la modificación del usuario.

Usuario Revisor tiene acceso a todas las funcionalidades, excepto la modificación de los detalles del horario, la modificación del usuario y la funcionalidad de aprobar.

Usuario por Carrera tiene acceso a todas las funcionalidades, excepto la modificación de los detalles del horario, la modificación del usuario y la funcionalidad de aprobar.

Se utilizó la herramienta online Lucid Chart para diagramar la navegación general dentro de la aplicación la cual limita el acceso a ciertas funciones según el rol. Tal como se muestra en la Figura 4.

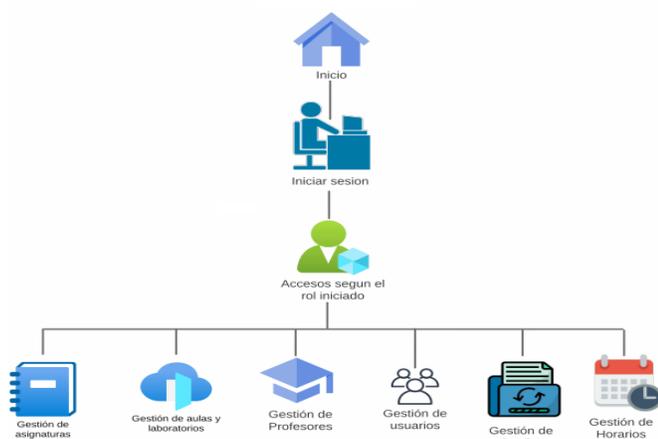


Figura 4. Diseño de la interfaz de usuario.

Arquitectura del desarrollo

Se debe elegir un patrón de arquitectura adecuada para el desarrollo del sistema web de creación de horarios de clase se utilizará el patrón de arquitectura MVC el cual divide la aplicación en componentes lógicos, el modelo define la estructura de datos y las consultas a la base de datos, la vista determina que datos debe mostrar y devuelve una respuesta y el controlador gestiona la interacción.

Es importante definir lo que es el modelo cliente servidor según el artículo de la web oficial de IBM que *“un servidor es un sistema que contiene datos o proporciona recursos a los que deben acceder otros sistemas de la red. Un cliente es un sistema que solicita servicios o datos de un servidor”*. (IBM, 2021)

Estos conceptos se relacionan debido a que al utilizarse juntos el cliente (navegador web) actúa como la vista en el patrón MVC porque es el responsable de mostrar la interfaz de usuario y recibir acciones, mientras que el servidor se encarga del modelo que tiene acceso al controlador y a los datos en el modelo, luego, el servidor envía la respuesta al cliente el cual se mostrara en la vista, Tal como se muestra en la Figura 5.

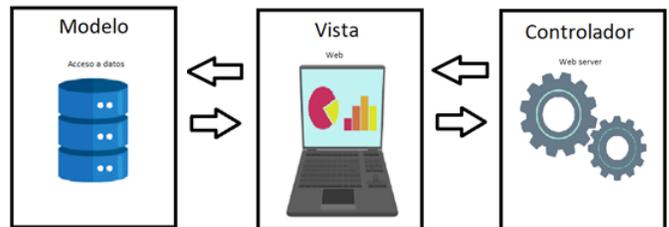


Figura 5. Modelo de arquitectura MVC.

A continuación, en las siguientes figuras se podrá visualizar la implementación la arquitectura MVC en el desarrollo el sistema web de creación de horarios de clase (Figura 6, 7 y 8).

Vista MVC del inicio de sesión



Sistema web de creación de horarios

Formulario de inicio de sesión con los siguientes campos:

- Usuario*
- Contraseña*
- Botón: Iniciar sesión

Figura 6. Vista MVC Inicio de sesión.

Modelo MVC del inicio de sesión

```
Generate tests for the below class
export class Usuario {
  constructor({
    public _id: string,
    public nombre: string,
    public username: string,
    public ci: string,
    public email: string,
    public contrasena: string,
    public rol: string,
    public phoneNumber?: string,
  }) {}
}
```

Figura 7. Modelo MVC usuario.

Controlador MVC del Inicio de sesión

```
loginjs M x
Backend > controllers > loginjs > controller > sendCode
25 |
26 | var controller = {}
27 |
28 | sendCode: async (req, res) => {
29 |   let { username, password } = req.body;
30 |   try {
31 |     user = await User.findOne({ $or: [{ username: username }, { ci: username }] });
32 |     if (!user) {
33 |       return res.status(200).json({ message: 'Usuario no encontrado', status: "error" });
34 |     }
35 |
36 |     let isPasswordMatch = await bcrypt.compare(password, user.contrasena);
37 |
38 |     if (!isPasswordMatch) {
39 |       return res.status(200).json({ message: 'Contraseña incorrecta', status: "error" });
40 |     }
41 |
42 |     // Generar el código de verificación
43 |     codigoVerificacion = generateVerificationCode();
44 |
45 |     // Guardar el código de verificación en el usuario
46 |     user.codigoVerificacion = codigoVerificacion;
47 |     await user.save();
48 |
49 |
50 |     const myAccessToken = await oAuth2Client.getAccessToken()
51 |
52 |
53 |     // Configurar el transporte de correo electrónico con nodemailer
54 |     let transporter = nodemailer.createTransport({
55 |       service: 'gmail',
56 |       auth: {
57 |         type: 'OAuth2',
58 |         user: 'sistemaunibehorarios@gmail.com',
59 |         clientId: '106822865384-1v7tmg8aeqststtaip63ecrki9epq1.apps.googleusercontent.com',
```

Figura 8. Controlador MVC de inicio de sesión.

Evaluación de la calidad del sistema web para la creación de horarios de clases en la UNIB.E, a través una encuesta dirigida a expertos en el área

La evaluación de la calidad del sistema web para la creación de horarios de clases permito saber que el 100% de los expertos en área de software consideran que el sistema permite crear horarios de clase de manera precisa y sin errores, tal como se observa en la Figura 9.

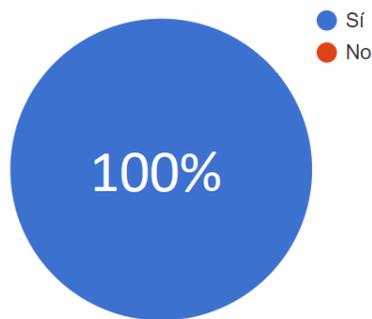


Figura 9. Creación de horarios de clase de manera precisa y sin errores.

De igual manera, se obtuvo que el 100% de los expertos en el área de software consideran que el sistema web es capaz de gestionar y asignar correctamente las asignaturas, docentes y aulas, tal como se observa en la Figura 10.

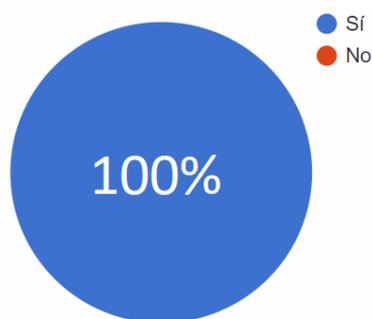


Figura 10. Gestión y asignación correcta de asignaturas, docentes y aulas.

Del mismo modo, se descubrió que el 100% de los expertos en el área de software consideran que el sistema web de creación de horarios puede acomodar cambios de último momento en los horarios de clase sin dificultad, tal como se observa en la Figura 11.

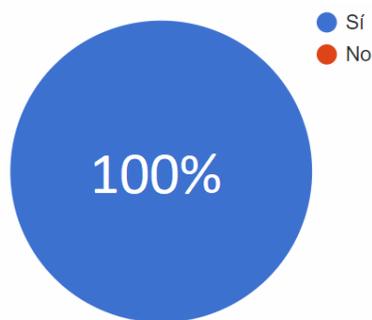


Figura 11. Acomodar cambios de último momento.

Asimismo, se evidenció que el 66,7% de los expertos en el área de software consideran que el sistema web de creación de horarios protege adecuadamente la información de los horarios de clase, evitando accesos no

autorizados o manipulando datos, mientras que el 33,3% no lo considera de esta manera tal como se observa en la Figura 12.

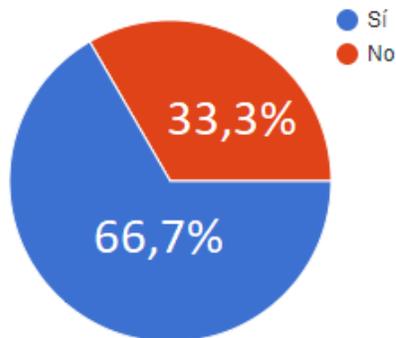


Figura 12. Protección de la información de los horarios de clase.

De igual forma, se evidenció que el 100% de los expertos en el área de software consideran que el sistema web es fácil de aprender y utilizar para los usuarios sin una formación extensa, tal como se observa en la figura 13.

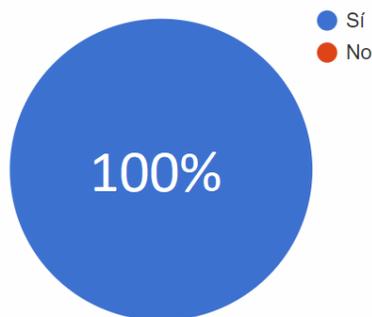


Figura 13. Fácil de usar y utilizar.

Por otro lado, se evidenció que el 66,7% de los expertos en el área de software consideran que el sistema web proporciona una interfaz intuitiva y clara para crear y gestionar horarios de clase, mientras que el 33,3% no lo considera de esta manera, tal como se observa en la figura 14.

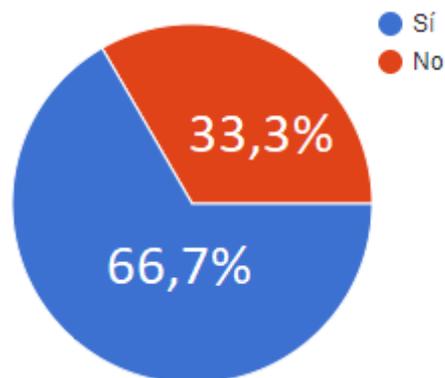


Figura 14. Interfaz intuitiva y clara.

CONCLUSIONES

Los requerimientos funcionales identificados, permitieron una asignación eficiente de profesores, gestión de aulas, recursos, y flexibilidad en la programación de horarios lo cual facilitó el proceso de automatización de creación de horarios de la UMET.

Los requerimientos no funcionales identificados garantizaron la eficacia y satisfacción de los usuarios al brindar usabilidad, estabilidad y confiabilidad en el sistema web de creación de horarios de clases.

El sistema web para la creación de horarios de clases en la UMET, fue desarrollado aplicando las mejores prácticas y metodologías adecuadas para garantizar la calidad del sistema web, logrando así un producto final robusto y funcional.

La evaluación de la calidad del sistema web ha demostrado que cumple con los estándares esperados, ya que proporciona a los usuarios una experiencia satisfactoria o eficiente por su facilidad de uso, generación precisa de horarios y la efectividad de las funcionalidades implementadas, tal como han señalado los expertos, asegurando su utilidad y valor para la comunidad académica.

REFERENCIAS BIBLIOGRÁFICAS

- Angular. (2022). The web development framework for building the future. <https://angular.io/>
- Baquerizo, J. (2021). Aplicación web para la automatización y gestión de horarios de clases mediante algoritmos basados en el modelo de programación lineal en la escuela 'Lic. Angélica Villón Lindao. (Trabajo de titulación). Universidad Estatal Península de Santa Elena.
- Gobierno de España. (2022). Real Decreto 95/2022 por el que se establece la ordenación y las enseñanzas mínimas de la Educación Infantil. <https://www.boe.es/buscar/pdf/2022/BOE-A-2022-1654-consolidado.pdf>
- Huanca, J. (2020). Sistema de información con arquitectura MVC para la recaudación tributaria por obras públicas de pistas y veredas de la municipalidad provincial de Puno. (Tesis de licenciatura). Universidad Nacional del Altiplano
- IBM. (2021). IBM. Obtenido de www.ibm.com
- Marqués, M. (2011). Bases de datos. Publicacions de la Universitat Jaume I.
- Nazareno, R., Leone, H., & Gonnet, S. (2013). Trazabilidad de procesos ágiles: un modelo para la trazabilidad de procesos Scrum. (Poencia). XVIII Congreso Argentino de Ciencias de la Computación. Universidad Nacional de La Plata, Argentina.
- Pressman, R., & Maxim, B. (2019). Software Engineering: A Practitioner's Approach. Mc-Graw Hill.

Real Academia Española. (2021). Diccionario de la Real Academia Española. <https://www.rae.es/>

Riehle, D. (2007). Framework Design: A Role Modeling Approach. (Tesis doctoral). SWISS Federal Institute of Technology Zurich.

Sommerville, I. (2019). Engineering Software Products: An Introduction to Modern Software Engineering. Pearson.