

PROTOTIPO DE GESTOR DE CONTENIDOS PARA PUBLICACIÓN DE TESIS DE GRADO EN TECNOLÓGICO NACIONAL CAMPUS CIUDAD JUÁREZ MEDIANTE PATRON MVC

CONTENT MANAGER PROTOTYPE FOR PUBLICATION OF GRADUATE THESIS AT TECNOLÓGICO NACIONAL CAMPUS CIUDAD JUAREZ USING MVC PATTERN

Valenzuela Leyva Víctor Manuel¹, Sánchez Leal María Eugenia², Anglés Barrios Claudia³, Gallegos Borunda Alma Patricia⁴, Farías Veloz Verónica⁵

¹Ingeniero en Sistemas Computacionales, Tecnológico Nacional del México / Instituto Tecnológico de Ciudad Juárez, vict_ley@hotmail.com, 656 6882500, Ave. Tecnológico 1340 Fracc. El Crucero, CP 32500, Cd Juárez, Chih. México

²Maestra en Sistemas de Información. Tecnológico Nacional del México / Instituto Tecnológico de Ciudad Juárez, Departamento de Sistemas y Computación. esanchez@itcj.edu.mx, 656 6882500, Ave. Tecnológico 1340 Fracc. El Crucero, CP 32500, Cd. Juárez, Chih. México

³Maestra en Software Libre. Tecnológico Nacional del México / Instituto Tecnológico de Ciudad Juárez. Departamento de Sistemas y Computación. cangles@itcj.edu.mx, 656 6882500, Ave. Tecnológico 1340 Fracc. El Crucero, CP 32500, Cd. Juárez, Chih. México

⁴Maestra en Gestión de Tecnologías de la Información. Tecnológico Nacional del México / Instituto Tecnológico de Ciudad Juárez. Departamento de Sistemas y Computación. pgallegos@itcj.edu.mx, 656 6882500, Ave. Tecnológico 1340 Fracc. El Crucero, CP 32500, Cd. Juárez, Chih. México

⁵Maestra Software Libre. Tecnológico Nacional del México / Instituto Tecnológico de Ciudad Juárez. Departamento de Sistemas y Computación. vfarias@itcj.edu.mx, 656 6882500, Ave. Tecnológico 1340 Fracc. El Crucero, CP 32500, Cd. Juárez, Chih. México

Resumen - Prototipo de CMS (Content Management System, Sistema Gestor de Contenidos) desarrollado para el departamento de CI (Centro de Información) del Tecnológico de México campus Ciudad Juárez.

Este sistema sirve para registrar y almacenar documentos de grado, postgrado e informes técnicos presentados por exalumnos del plantel para obtener su título profesional.

Consiste en un prototipo de sitio web con diseño sencillo, intuitivo y fácil de usar por cualquier tipo de usuario, desarrollado en PHP 7 empleado el framework Laravel y con vistas elaboradas con HTML 5 y Bootstrap 4.6.

El principal propósito de este sistema es dar inicio al desarrollo de un sistema funcional final más robusto, eficiente y atractivo.

El prototipo desarrollado e implementado a manera de prueba, demuestra que a corto y mediano plazo tendrá un impacto positivo en la administración o gestión institucional de contenidos de documentos de tesis, proporcionando una base de datos de consulta y resguardo de tan importantes proyectos desarrollados por egresados en todas las carreras de licenciatura, maestría y doctorado.

Palabras Clave: Gestor de Contenidos, prototipo, almacenamiento, indexación

Abstract - Prototype of CMS (Content Management System) developed for the IC department (Information Center) of Tecnológico de México, Ciudad Juárez campus.

This system is used to record and store undergraduate and postgraduate documents and technical reports that former

students of the campus have submitted to obtain their professional title.

The prototype is a website with a design that is simple, intuitive, and easy to use for any type of user, developed in PHP 7 using the Laravel framework, and views made with HTML 5 and Bootstrap 4.6.

The main purpose of this system is to start the development of a more robust, efficient and attractive functional final system.

The prototype developed and implemented as a test, shows that in the short and medium terms it will have a positive impact on the institutional administration or content management of thesis documents, providing a database for consultation and protection of such important developed projects by graduates of all bachelor's, master's, and PhD's degrees.

Key words – Content manager, prototype, storage, indexing.

INTRODUCCIÓN

El propósito de este proyecto es, diseñar un prototipo de repositorio digital, es decir, un Sistema Gestor de Contenidos (CMS) que facilite el acceso a los documentos de grado a la comunidad tecnológica del campus.

El Centro de Información (CI) es la unidad que se encarga del control de los materiales bibliográficos y de brindar los servicios bibliotecarios necesarios para la comunidad del Tecnológico Nacional Campus Ciudad Juárez. Así mismo, el CI es el lugar donde se puede encontrar el repositorio de todos los documentos de grado y postgrado

de ex alumnos del Instituto que obtuvieron su título profesional.

Debido a la limitante del software en el uso exclusivo dentro del centro de información y que únicamente brinda información de forma general de los documentos, ocasionalmente son consultados; aunado a esto, requiere la aprobación de la jefatura del Centro de Información para solicitar el préstamo de alguno de ellos.

Por estos motivos, esos documentos no se emplean como fuentes de consultas para trabajos futuros, lo que provoca que pierdan valor académico como tesis de grado.

Desarrollar un prototipo de un sistema gestor de contenidos para la publicación de tesis de grado mediante el patrón MVC (Modelo-Vista-Controlador), es el inicio de un sistema final que no solo permitirá llevar una administración de los documentos de forma digital, si no que al ser un CMS, estará disponible en línea para que se pueda acceder de una forma sencilla a la información desde cualquier punto, permitiendo realizar búsquedas de documentos de tesis de grado y así mismo obtener una copia digital de lo consultado. Además, este sistema ayudará a que estos documentos formen parte de las fuentes de investigación de manera más recurrente para alumnos activos, agregándoles valor académico.

DESARROLLO

Metodología

Según la definición planteada por Maida y Pacienza en 2015 [6], una metodología para el desarrollo de software es un marco de referencia que se utiliza para estructurar, planificar y dar seguimiento en el proceso de desarrollo de sistemas de información y no tiene que ser necesariamente adecuada para usarla en todos los proyectos.

En la realización de este proyecto, se utilizó la metodología de desarrollo de software ágil SCRUM (figura 1).

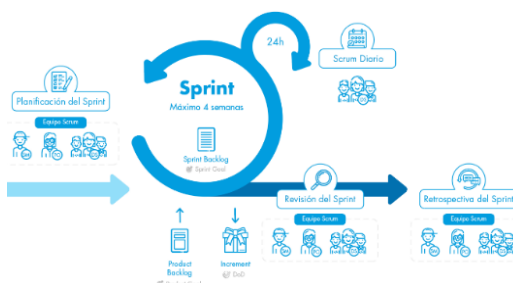


Figura 1. Flujo del ciclo SCRUM [21].

El patrón MVC comprende una arquitectura de software que separa los datos de una aplicación, la interfaz de usuario y la lógica del sistema en 3 componentes distintos como se observan en la figura 2: el modelo, la vista y el controlador; lo que permite mantener la arquitectura de la aplicación en distintas capas que se encargan de realizar diferentes tareas, ofreciendo diversos beneficios en el desarrollo de la aplicación [3].

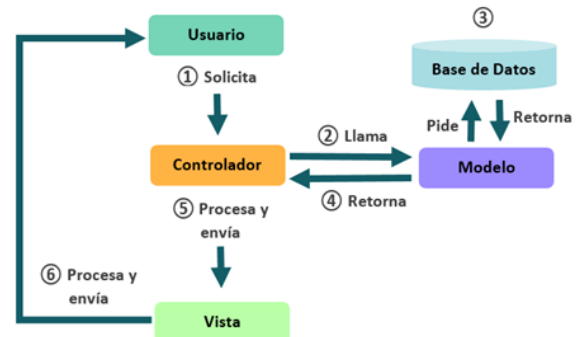


Figura 2. Flujo del proceso del patrón MVC [8].

Para el uso de SCRUM, se establecen 3 roles: Product Owner, SCRUM Master y el Development team [2], sin embargo, al ser este un equipo pequeño se decidió prescindir del SCRUM Master

- Product Owner: LA Jessika Elvira Martínez Urías.
- Development team: Victor Manuel Valenzuela Leyva.

Se llevo a cabo una reunión con el Product Owner Jessika Martínez el día 25 de noviembre del 2020, proponiendo el prototipo de sistema y explicando el propósito de este. En esta misma reunión se realizó el análisis inicial de las necesidades por cubrir y se establecieron los requerimientos funcionales y no funcionales que comprendería el sistema (Tabla 1 y Tabla 2).

También, se estableció que, a partir de la fecha de inicio del desarrollo del proyecto, el término de cada iteración seria en lapsos de 4 semanas (cada mes) hasta terminar el proyecto.

Tabla 1. Requerimientos no funcionales.

Requerimiento	Descripción
RNF1	La interfaz gráfica debe contener el logo de la secretaria de Educación Pública, Tecnológico Nacional de México y del Tecnológico Campus Ciudad Juárez.
RNF2	El diseño de la interfaz debe ser simple y fácil de entender.
RNF3	La navegación por el sitio debe ser simple y sin tantos pasos.

Tabla 2. Requerimientos funcionales.

Requerimiento	Descripción
RF1	El sistema debe permitir el almacenamiento en el servidor de documentos digitales.
RF2	El sistema debe permitir realizar búsquedas de documentos mediante palabras clave.
RF3	El sistema debe mostrar una lista con los resultados de la búsqueda realizada.
RF4	El sistema debe permitir la visualización de solo la información concerniente al usuario del registro del documento seleccionado por el mismo.
RF5	El sistema debe permitir la visualización del documento desde el servidor (sin descargar).
RF6	EL sistema debe permitir la descarga del documento digital.
RF7	El sistema debe contar con un sistema de sesiones para administradores.
RF8	EL sistema debe permitir a los administradores el registro de la información de nuevos documentos en una base de datos.
RF9	El sistema debe permitir a los administradores la carga de documentos nuevos al servidor desde la interfaz gráfica.
RF10	EL sistema debe permitir a los administradores editar la información de los registros de documentos.
RF11	El sistema debe permitir a los administradores eliminar registros de documentos.
RF12	El sistema debe permitir a los administradores eliminar documentos almacenados en el servidor.

Product Backlog

En el Product Backlog del Sprint se tomaron en cuenta todas las actividades necesarias las cuales fueron clasificadas por niveles de prioridad determinados en base al impacto que tiene cada actividad para el desarrollo del sistema.

El nivel de prioridad de cada una de las actividades se determinó tomando en cuenta la posibilidad de verse afectadas de diversas formas. Por ejemplo, una actividad puede alterarse al modificar uno de los requisitos ya sea funcional o no funcional o de errores que se encuentren durante el desarrollo o en las pruebas del sistema.

En la tabla 3 se observa la lista de actividades planeadas, y la prioridad que se establece para su cumplimiento.

Tabla 3. Product Backlog.

Actividades	Prioridad
Análisis de requerimientos.	ALTA
Selección de tecnologías.	ALTA
Diseño de la arquitectura.	ALTA
Diseño de la base de datos.	MEDIA
Diseño de las interfaces.	MEDIA
Desarrollo Front-end.	MEDIA
Desarrollo Back-end.	ALTA

Pruebas.	ALTA
Entregas y evaluaciones.	ALTA
Correcciones y modificaciones.	MEDIA
Documentación.	MEDIA

Burndown Chart.

En la Figura 3 se presenta de forma visual el tiempo en días utilizado en cada uno de los 4 Sprints totales. Dicho estimado fue una guía para determinar en qué puntos del desarrollo del proyecto se requirió de más tiempo.

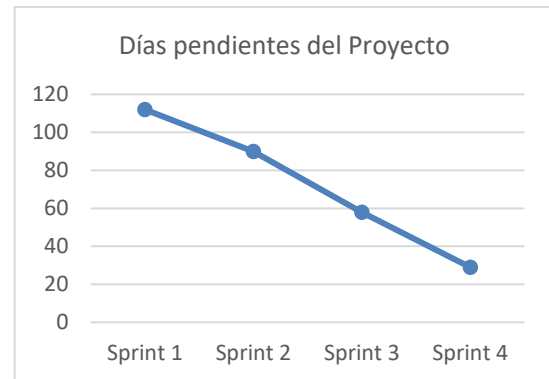


Figura 3. Burndown Chart del proyecto.

Cronología de Sprints.

Siguiendo la metodología SCRUM el proyecto se desarrolló a través de Sprints, los cuales son variantes respecto a tiempo y actividades a cumplir, para ello se decide cuales tareas se agregarán dentro de un sprint en específico y su propósito general como se muestra en la tabla 4.

Tabla 1. Cronología de los Sprints.

Sprint	Actividades	Prioridad
1	Análisis de requerimientos.	ALTA
	Selección de tecnologías.	ALTA
	Diseño de arquitectura.	ALTA
	Diseño de base de datos.	MEDIA
	Diseño de interfaces.	MEDIA
	Entregas y evaluaciones del Sprint 1.	ALTA
2	Análisis de requerimientos.	ALTA
	Desarrollo Front-end.	MEDIA
	Desarrollo Back-end.	ALTA
	Correcciones y modificaciones.	MEDIA
	Entregas y evaluaciones del Sprint 2.	ALTA
3	Análisis de requerimientos.	ALTA
	Desarrollo Front-end.	MEDIA
	Desarrollo Back-end.	ALTA
	Correcciones y modificaciones.	MEDIA
4	Entregas y evaluaciones del Sprint 3.	ALTA
	Análisis de requerimientos.	ALTA
	Desarrollo Back-end.	ALTA
	Pruebas.	ALTA
	Correcciones y modificaciones.	MEDIA
	Documentación.	MEDIA
	Entregas y evaluaciones del Sprint 4.	ALTA

En la figura 4 se muestra el diagrama general de casos de uso del proyecto.

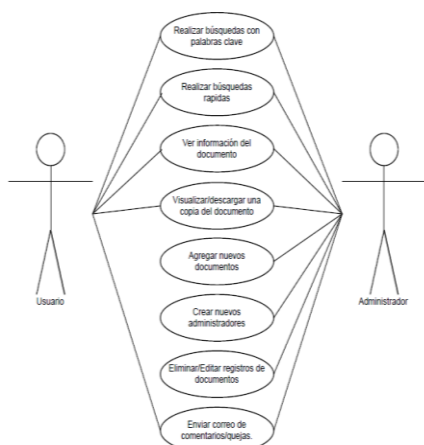


Figura 4. Diagrama general de casos de uso del sistema

Diagrama de secuencia.

En la figura 5 se muestra el diagrama de secuencia de la funcionalidad principal del sistema: búsqueda, visualización y descarga de documentos.

- Primero el usuario ingresa las palabras clave para realizar la búsqueda.
- El sistema genera una consulta en base a las palabras ingresadas y la envía al SGBD.
- El SGBD ejecuta la consulta en la base de datos y retorna las coincidencias encontradas.
- El sistema recibe el resultado, lo convierte en arreglo y lo muestra al usuario en forma de lista.
- En este punto el usuario puede observar la lista y seleccionar uno de los documentos.
- Aquí el usuario tiene dos opciones:
 - Puede visualizar el documento en el navegador y después descargarlo.
 - Descargar el documento directamente sin una previa visualización.

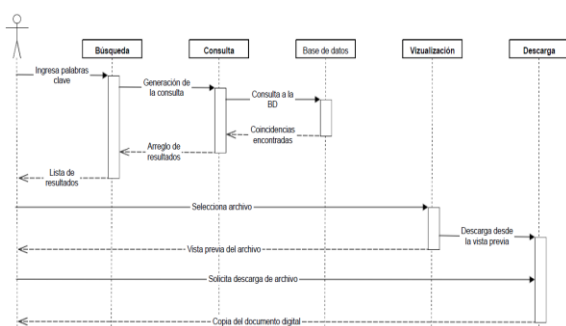


Figura 5. Diagrama de secuencias de búsqueda, visualización y descarga.

Arquitectura del sistema basado en patrón MVC.

En este sistema se planteó la arquitectura basada en el patrón MVC para facilitar cambios, mejoras y aditamentos futuros, ya que este es solo un prototipo de un sistema final.

Aquí el usuario se comunica con el sistema a través de vistas que proporcionan las herramientas necesarias para interactuar.

Los controladores son los encargados de responder a las peticiones del usuario [22], es en esta parte donde se lleva el control del almacenamiento de los documentos digitales en el servidor al igual que las instrucciones de manipulación de datos en la base de datos, es decir, creación, edición y eliminación de registros.

Los modelos se encargan de establecer como se trabajará con los datos. La manipulación de la información en la base de datos se hace a través de estos modelos, para asegurar que todo se encuentre en orden [22].

Una vez que el sistema ha terminado de preparar las respuestas a las peticiones del usuario, estas son enviadas a las vistas mostrando en forma de lista los resultados.

Diagrama Entidad-Relación.

En la figura 6 se identifica la base de datos, que contiene dos tablas: una de usuarios para los administradores y otra de documentos para el almacén de los datos; con una relación simple de 1: N donde un administrador puede crear varios documentos, pero un documento puede ser creado por un solo administrador.

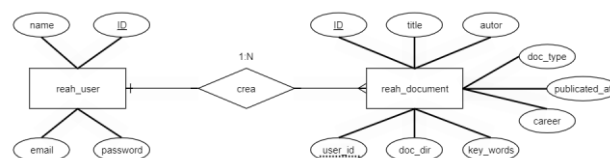


Figura 6. Diagrama E-R de la base de datos.

Indexación.

El motor de búsqueda del sistema emplea una instrucción de búsqueda en SQL avanzada conocida como Full-Text Search en MySQL.

Full-Text Search permite realizar búsquedas de coincidencias de palabras dentro de campos de texto extenso de forma más rápida en comparación a un Query Select normal.

Se indica en la base de datos qué campo es empleado para realizar la búsqueda en la creación del index Full-Text. Para este caso se creó el index en el campo `key_words` de la tabla `documents` con el nombre `documents_key_words`, agregando el siguiente código a la migración de la tabla `documents` del proyecto:

```
DB::statement('
    create fulltext index doc
uments_key_words_fulltext
on documents(key_words);
');
```


Después se refrescaron las tablas ejecutando el comando php artisan migration: refresh en la terminal desde el directorio del proyecto.

Finalmente, de esta manera se indexaron los registros en base a la columna *key_words* que ayudará en las búsquedas mediante palabras clave.

Diagrama de navegación.

En la figura 7 se puede apreciar el diagrama de navegación entre las vistas del sistema. El cual permite que de cada página se pueda retroceder a la anterior, y también regresar hasta la página index.

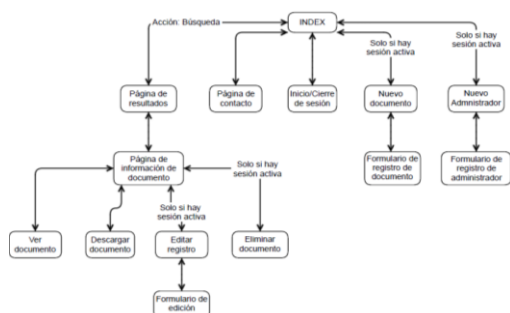


Figura 7. Diagrama de navegación entre vistas.

Diseño de la interfaz, Layout de encabezado y pie de página.

Primeramente, se genera el layout del encabezado y pie de página del sistema. Teniendo en cuenta los métodos para diseño de interfaces que proporciona Laravel, éste permite incluir dicha plantilla en las páginas donde se requiera, logrando agregar el encabezado y el pie de página del sitio sin necesidad de anexar el código de dichos componentes en cada archivo de código fuente de las vistas. Es en esta layout donde se encuentran las opciones de inicio de sesión y el formulario de contacto, así como el menú de opciones del administrador que aparecen después del inicio de sesión.

En la figura 8 se muestra el encabezado y pie de página de las vistas del sistema.



Figura 8. Vista de encabezado y pie de página. (Elaboración propia).

Interfaz de inicio.

En la interfaz de inicio del sistema (Figura 9) el usuario realiza una búsqueda por palabras clave. En caso de no

introducir palabras en el campo asignado y solo hacer clic al botón buscar, el sistema retornará todos los registros existentes como respuesta a la solicitud.



Figura 9. Interfaz de inicio del sistema. (Elaboración propia).

Formulario de búsqueda rápida.

La figura 10 muestra el formulario donde se realizan consultas rápidas conociendo el título del documento o el nombre del autor.

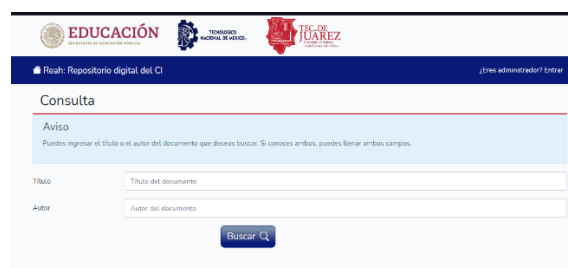


Figura 10. Vista de búsqueda rápida. (Elaboración propia).

DISCUSIÓN Y ANÁLISIS DE RESULTADOS

Los resultados obtenidos de la investigación “Prototipo de Gestor de Contenidos para Publicación de Tesis de Grado en TecNM Campus Ciudad Juárez Mediante Patrón MVC”, se enlistan a continuación:

- Almacenar los documentos: permite cargar documentos nuevos al sistema desde la interfaz de usuario, alojándolos en una dirección del servidor.
- La base de datos organiza la información general de los documentos, así como su ubicación en el servidor.
- Búsqueda rápida mediante palabras clave relacionadas a los mismos: apoyándose del mecanismo de indexación.

En la figura 11 se muestra un ejemplo de esta vista donde se enlistan todos los resultados que coincidieron con las palabras clave usadas al realizar la búsqueda. Se puede seleccionar un documento haciendo clic sobre el título de este, acción que llevará a la vista de información.

Título	Autor	Tipo de documento	Fecha de publicación	Carrera
Et laborum culpa corporis praesentium tenetur dolores ipsa.	Mr. Harry Oberbrunnere	Tesis de grado	1970-01-01	Mecánica
Et aut aliquam laborum.	Sister Wile	Tesis de grado	1970-01-01	Electrónica
Eos enim magni omnis.	Palma Johnson	Tesis de grado	1970-01-01	Electrónica
Vel exercitationem et est rem ipsum aut.	Alexandra Klado	Tesis de grado	1970-01-01	Electrónica
Molestiae aut qui praesentium.	Melina Mayer	Tesis de grado	1970-01-01	Electrónica
Cum repellat molestias qui consequuntur suscipit tempora.	Maxime Armstrong III	Tesis de grado	1970-01-01	Electrónica
Enim itaque est at ea.	Orion Welch	Tesis de grado	1970-01-01	Mecánica
Qui odio qui sed at consequatur.	Talitha Leffler	Tesis de grado	1970-01-01	Electrónica

Figura 1 Vista de resultados.
(Elaboración propia).

Vista de información del documento.

Aquí se presenta la información concerniente al usuario del documento seleccionado, como es: título, autor, carrera para la cual fue elaborado, entre otra información.

Como se puede observar en la figura 12 en esta vista se encuentran opciones para:

- Obtener una vista previa desde el servidor del documento sin necesidad de descargar el archivo digital.
- Descargar una copia del archivo digital del documento seleccionado.
- En caso de ser el administrador: la opción para editar la información del documento que se encuentra almacenada en el registro de la base de datos, la opción para eliminar el registro del documento en la base de datos, así como el archivo digital almacenado en el servidor.

Reah: Repositorio digital del CI

Información del documento

Título: Et laborum culpa corporis praesentium tenetur dolores ipsa.

Autor: Mr. Harry Oberbrunnere

Tipo: Tesis de grado

Fecha de publicación: 1970-01-01

Carrera: Mecánica

Resumen: Eos temporibus aut nobis sit et. In occaecati aut harum quo facilis. Tenetur voluptatibus eos necessitatibus dolorem. Maiores laborum vel vitae at consequuntur sed. Eos temporibus aut nobis sit et. In occaecati aut harum quo facilis. Tenetur voluptatibus eos necessitatibus dolorem. Maiores laborum vel vitae at consequuntur sed. Eos temporibus aut nobis sit et. In occaecati aut harum quo facilis. Tenetur voluptatibus eos necessitatibus dolorem. Maiores laborum vel vitae at consequuntur sed. Eos temporibus aut nobis sit et. In occaecati aut harum quo facilis. Tenetur voluptatibus eos necessitatibus dolorem. Maiores laborum vel vitae at consequuntur sed.

Editar Eliminar Ver Descargar

Repositorio digital del Centro de información del Tecnológico Nacional de México Campus Ciudad Juárez. ¿Tienes dudas o necesitas ayuda? Contáctanos

Figura 2. Vista de información del documento.
(Elaboración propia).

Vista de formulario para editar el registro del documento

La figura 13 muestra el formulario de edición de información del registro de un documento en la base de datos. Esta vista solo está disponible para administradores.

Reah: Repositorio digital del CI

Editar documento

Aviso: Solo puedes editar la información de este registro, mas no el documento. Si deseas cambiar el documento nuevo.

Título: Et laborum culpa corporis praesentium tenetur dolores ipsa.

Autor: Mr. Harry Oberbrunnere

Tipo de documento: Tesis de grado

Fecha de publicación: 01/01/1970

Carrera: Mecánica

Resumen: Eos temporibus aut nobis sit et. In occaecati aut harum quo facilis. Tenetur voluptatibus eos necessitatibus dolorem. Maiores laborum vel vitae at consequuntur sed. Eos temporibus aut nobis sit et. In occaecati aut harum quo facilis. Tenetur voluptatibus eos necessitatibus dolorem. Maiores laborum vel vitae at consequuntur sed.

Palabras clave: Est officis quia ut commodi molestias magni ratione ut culpa laborum

Figura 3. Formulario de edición de registros.
(Elaboración propia).

Vista de formulario para subir un nuevo documento.

Esta vista muestra el formulario que permite cargar nuevos documentos al servidor a la vez que crea el registro de los datos en la base de datos. Solo administradores tienen acceso a esta vista (figura 14).

Reah: Repositorio digital del CI

Nuevo documento

Título: Título del documento

Autor: Autor del documento

Tipo de documento: Tipo del documento

Fecha de publicación: dd/mm/aaaa

Carrera:

Resumen: Resumen sobre el documento

Palabras clave: Palabras clave relacionadas con el documento

Documento: Seleccionar archivo...

Figura 4 Formulario para registro de documentos nuevos.
(Elaboración propia).

Formulario para registro de nuevo administrador

La figura 15 ilustra la pantalla que permite el registro de nuevos administradores para el sistema. Solamente administradores del sistema tienen acceso a ella.

Reah: Repositorio digital del CI

Registro

Nombre:

Correo electrónico:

Contraseña:

Confirmar contraseña:

Registro

Repositorio digital del Centro de información del Tecnológico Nacional de México Campus Ciudad Juárez.

Figura 5 Formulario para crear nuevos administradores.
(Elaboración propia).

Implementación.

Durante la implementación se generó una instalación de prueba en un servidor de host para el sistema, optando por Hostinger, el cual comprende un plan de hospedaje CPanel con un costo de MX\$36.00 por mes (al pagar dos años de servicio) y hace uso del nombre de dominio reah.xyz como se muestra en la figura 16:



Figura 16. Nombre de dominio reah.xyz
(Elaboración propia)

Almacenamiento.

Como se muestra en la figura 17, el servicio de host cuenta con un sistema de Administración de archivos que permite manipular los archivos del proyecto y almacenar los archivos digitales de los documentos:

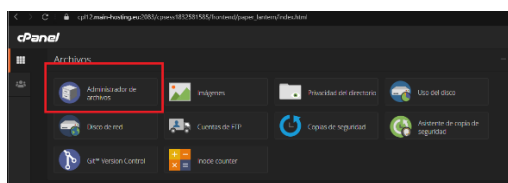


Figura 17. Panel de control de CPanel.
(Elaboración propia)

Al ingresar a esta opción, despliega la estructura de archivos del servidor (figura 18):

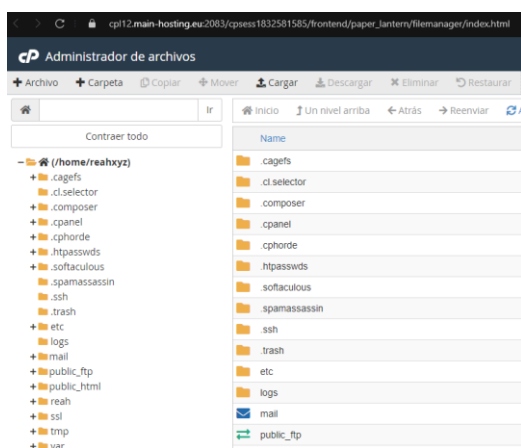


Figura 18. Administrador de archivos de CPanel.
(Elaboración propia)

En el directorio público public_html>documents (figura 19), se encuentran almacenados los archivos digitales:

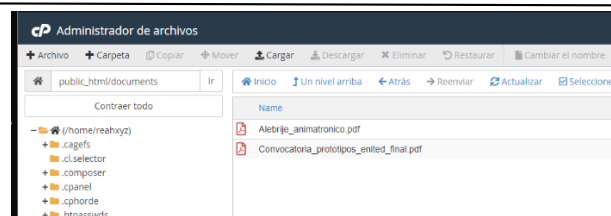


Figura 19. Almacenamiento de archivos.
(Elaboración propia)

Indexación y organización.

Se implementó un método de indexación brindado por el motor de bases de datos: Full-Text Search Index, especificado previamente en el desarrollo de la base de datos.

De igual forma, se almacenan en la base de datos el nombre del archivo digital del documento y su título debido que no siempre son iguales, a la vez permite la manipulación e identificación del archivo desde el sistema.

Búsqueda mediante palabras clave

Para esta búsqueda se usó el Full-Text Search Index, lo cual permitió indexar registros en base a un campo de texto largo, para luego realizar consultas de coincidencias de palabras a la tabla de documentos, lo que mostró solo resultados que coincidieron con las palabras.

Por ejemplo, al intentar con la palabra clave "inteligencia" como se muestra en la figura 20,



Figura 20. Ejemplo de búsqueda "Inteligencia".
(Elaboración propia)

se obtiene como resultado el documento de inteligencia artificial como se aprecia en la figura 21.



Figura 21. Resultado de búsqueda "Inteligencia".
(Elaboración propia).

Ahora bien, si se cambia por la búsqueda por la palabra "electromecánica" (figura 22) se obtiene un resultado diferente (figura 23):



Figura 22. Ejemplo de búsqueda "electromecánica".
(Elaboración propia)



Figura 23. Resultados de búsqueda "electromecánica".
(Elaboración propia).

Y como se observa en la figura 24, ambos registros existen a la vez en el sistema:



Figura 24. Existencia de documentos en el sistema.
(Elaboración propia)

Mostrar el documento y descargar copias

El sistema ofrece dichas opciones para cualquiera de los usuarios en la página de información del documento como se puede apreciar en la figura 25.



Figura 25. Opciones de Ver y Descargar.

Al presionar el botón "Ver", se obtiene una vista del documento que utiliza el visor de documentos PDF del navegador desde el cual se accede a la página como se observa en la figura 26.

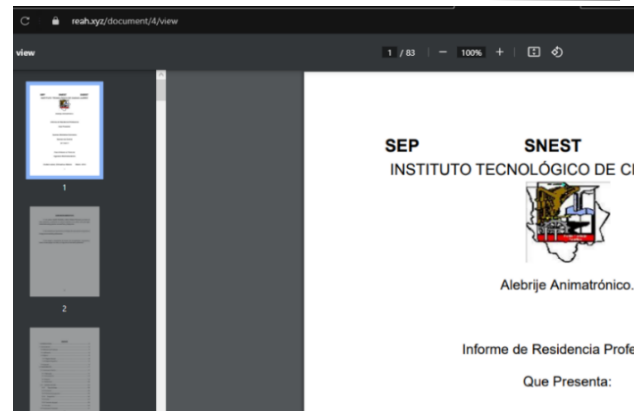


Figura 26. Visor de documentos PDF.
(Elaboración propia)

Por otro lado, al presionar el botón "Descargar", iniciará la descarga del archivo digital del documento hacia el dispositivo desde el cual se accede al sitio (figura 27):

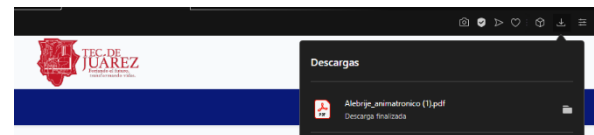


Figura 27. Descarga de archivo digital.
(Elaboración propia)

Proyección de almacenamiento de los documentos.

El almacenamiento es el recurso más relevante para el sistema debido a que consiste en la recopilación de los documentos en línea.

Actualmente se encuentran almacenados en el sistema 52 documentos del año 2014 que ocupan un total de 123.84MB de capacidad, resultando en un promedio aproximado de 2.38MB por cada archivo.

La tabla 5 muestra la información de egresados de los años 2017 al 2022 obtenida de las notas publicadas en la página oficial del TecNM Campus Ciudad Juárez.

Tabla 5. Cantidad de egresados por generación del 2017 al 2022

Generación	Egresados
Junio 2017	160 egresados [11]
Diciembre 2017	196 egresados [12]
Junio 2018	210 egresados [14]
Diciembre 2018	310 egresados [13]
Junio 2019	150 egresados [15]
Diciembre 2019	230 egresados [16]
Junio 2020	148 egresados
Diciembre 2020	170 egresados
Junio 2021	338 egresados
Diciembre 2021	303 egresados
Junio 2022	245 egresados

Considerando esta información se crearon estimaciones del espacio de almacenamiento requerido para los documentos de todos estos egresados (Tabla 6). En la Figura 28 se puede apreciar de forma visual.

Tabla 6. Capacidad de almacenamiento requerida por generación

Generación	Egresados	Espacio requerido en MB
Junio 2017	160 egresados	380.80
Diciembre 2017	196 egresados	466.48
Junio 2018	210 egresados	499.80
Diciembre 2018	310 egresados	737.80
Junio 2019	150 egresados	357.00
Diciembre 2019	230 egresados	547.40
Junio 2020	148 egresados	352.24
Diciembre 2020	170 egresados	404.60
Junio 2021	338 egresados	804.44
Diciembre 2021	303 egresados	721.14
Junio 2022	245 egresados	583.10

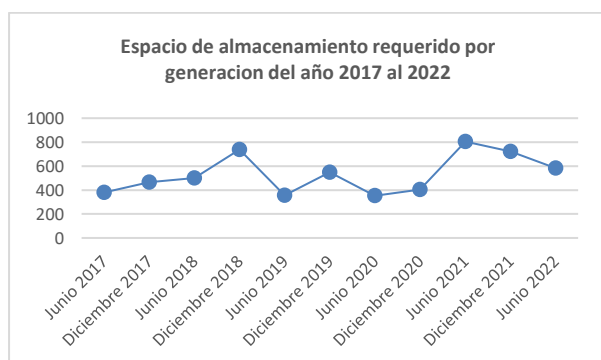


Figura 28. Capacidad de almacenamiento requerida por generación.
(Elaboración propia)

Bajo este enfoque, la siguiente fase del proyecto estará encaminado a dejar de ser un prototipo, mediante un trabajo de seguimiento para implementarlo de manera total y que los estudiantes de todos los niveles académicos del Instituto Tecnológico de Ciudad Juárez puedan disponer de los documentos a través del gestor de contenidos.

CONCLUSIONES

Este prototipo está terminado, es un sistema funcional y cumple con los objetivos establecidos al inicio del proyecto.

Al ser este sistema solo un prototipo inicial de un sistema final más robusto, se enlistan recomendaciones:

- Mejorar la interfaz gráfica, que sea más atractivo a la vista y que facilite la navegación por el sitio a los usuarios agregando enlaces en la página de inicio.
- Cambiar el método de indexación por algún otro, sugiriendo la implementación de una librería como ElasticSearch para mayor funcionalidad.
- Instalación del sistema en un servidor local evitando que sobrepase capacidad de almacenamiento.
- Agregar filtros para búsquedas más específicas y generando un sistema más eficiente.

BIBLIOGRAFÍA

- [1] Aitana soluciones ERP y CRM. (16 de octubre de 2018). *Visual Studio Code: Funcionalidades y extensiones*. Obtenido de Aitana: <https://blog.aitana.es/2018/10/16/visual-studio-code/>
- [2] Abellán, E. (5 de marzo de 2020). *Scrum: qué es y cómo funciona esta metodología*. Obtenido de We Are Marketing (WAM): <https://www.wearemarketing.com/es/blog/metodologia-scrum-que-es-y-como-functiona.html#>
- [3] Alvarez, M. A. (28 de julio de 2020). *Qué es MVC*. Obtenido de Desarrolloweb.com : <https://desarrolloweb.com/articulos/que-es-mvc.html>
- [4] B., G. (3 de diciembre de 2020). *¿Qué es MySQL? Explicación detallada para principiantes*. Obtenido de Hostinger : <https://www.hostinger.mx/tutoriales/que-es-mysql#Que-es-MySQL>
- [5] Cuatroclicks. (16 de julio de 2020). *Software Gestor de Contenido*. Obtenido de Cuatroclicks: <https://cuatroclicks.com/software-gestor-de-contenido/>
- [6] E. G. Maida y J. Pacienza, (2015). «Metodologías de desarrollo de software»
- [7] García Cobian, V. (1 de mayo de 2019). *Patrón de diseño MVC [Figura]*. Obtenido de nearsoftjobs: <https://blog.nearsoftjobs.com/patrón-de-diseño-mvc-2366948b5fc7>
- [8] Gómez, R. (11 de noviembre de 2015). *Modelo Vista Controlador [Figura]*. Obtenido de rodrigogr: <http://rodrigogr.com/blog/modelo-vista-controlador/>
- [9] IBM Corp. (11 de agosto de 2016). *Definición de terminos: patrón de arquitectura*. Obtenido de cgr.go.cr: https://cgrw01.cgr.go.cr/rup/RUP.es/SmallProjects/core.base_rup/guidances/termdefinitions/architectural_pattern_E2E8EB79.html
- [10] Innovamediamkt. (16 de marzo de 2018). *¿Qué es un gestor de contenidos o CMS?* Obtenido de innovamedia : <https://www.innovamediaconsultores.com/blog/gestor-contenido-cms/>
- [11] ITCJ. (26 de junio de 2017). *GRADUAN 160 ALUMNOS DEL ITCJ*. Obtenido de ITCJ: <http://cdjuarez.tecnm.mx/blog/nota/329/GRADUAN-160-ALUMNOS-DEL-ITCJ>
- [12] ITCJ. (8 de diciembre de 2017). *GRADUÁN 196 ESTUDIANTES EN EL ITCJ*. Obtenido de ITCJ: <http://www.itcj.edu.mx/blog/nota/374/GRAD%C3%9AAN-196-ESTUDIANTES-EN-EL-ITCJ>
- [13] ITCJ. (17 de diciembre de 2018). *EGRESAN 310 PROFESIONISTAS DEL IT DE CIUDAD JUÁREZ*. Obtenido de ITCJ: <http://cdjuarez.tecnm.mx/blog/nota/501/EGRESAN-310-PROFESIONISTAS-DEL-IT-DE-CIUDAD-JU%C3%81REZ>
- [14] ITCJ. (25 de junio de 2018). *ITCJ realiza su Graduación Junio 2018*. Obtenido de ITCJ: <http://cdjuarez.tecnm.mx/blog/nota/441/ITCJ-realiza-su-Graduaci%C3%B3n-Junio-2018>

- [15] ITCJ. (21 de junio de 2019). *EGRESAN 150 LIEBRES DEL TEC DE JUÁREZ GENERACIÓN JUNIO 2019*. Obtenido de ITCJ:
<http://www.itcj.edu.mx/blog/nota/595/EGRESAN-150-LIEBRES-DEL-TEC-DE-JUREZ-GENERACIN-JUNIO-2019>
- [16] ITCJ. (16 de 12 de 2019). *EGRESAN 230 LIEBRES DEL TEC DE JUÁREZ GENERACIÓN DICIEMBRE 2019*. Obtenido de ITCJ:
<http://www.itcj.edu.mx/blog/nota/674/EGRESAN-230-LIEBRES-DEL-TEC-DE-JU%C3%81REZ-GENERACI%C3%93N-DICIEMBRE-2019>
- [17] Muent, G. (8 de enero de 2020). *Guía completa del Framework: qué es, cuáles tipos existen y por qué es importante en Internet*. Obtenido de Rockcontent:
<https://rockcontent.com/es/blog/framework/>
- [18] Otto, M., & Thornton, J. (19 de enero de 2021). *Bootstrap Doc*. Obtenido de Getbootstrap:
<https://getbootstrap.com/docs/4.6/getting-started/introduction/>
- [19] Otwell, T., & Et. (5 de noviembre de 2019). *Laravel 7.x Doc*. Obtenido de Laravel:
<https://laravel.com/docs/7.x>
- [20] Ramos, P. (14 de septiembre de 2018). *Que es y para qué sirve SQL*. Obtenido de styde.net:
<https://styde.net/que-es-y-para-que-sirve-sql/>
- [21] Rodríguez, M. (24 de diciembre de 2020). *Scrum: el pasado y el futuro [Figura]*. Obtenido de netmind:
<https://netmind.net/es/scrum-el-pasado-y-el-futuro/>
- [22] Servicio de Informática. (22 de abril de 2013). *Modelo vista controlador (MVC)*. Obtenido de Universidad de Alicante:
<https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html>
- [23] Software para todo. (11 de marzo de 2019). *¿Qué es un prototipo de software?* Obtenido de Software para todo: <https://softwareparatodo.com/que-es-un-prototipo-de-software/>
- [24] Villán, V. R. (15 de marzo de 2019). *Las metodologías ágiles más utilizadas y sus ventajas dentro de la empresa*. Obtenido de IeBS:
<https://www.iebschool.com/blog/que-son-metodologias-agiles-agile-scrum/>

Validación	Víctor Manuel Valenzuela Leyva
Visualización	María Eugenia Sánchez Leal Claudia Anglés Barrios Verónica Farías Veloz
Redacción borrador original	María Eugenia Sánchez Leal Verónica Farías Veloz
Redacción, revisión y edición	Claudia Anglés Barrios, Alma Patricia Gallegos Borunda, Verónica Farías Veloz



Esta obra está bajo una licencia internacional Creative Commons Atribución 4.0.

ROLES DE CONTRIBUCIONES

Rol	Autor(es)
Conceptualización Metodología Administración del Proyecto	Víctor Manuel Valenzuela Leyva María Eugenia Sánchez Leal
Curación de Datos Recursos Software	Víctor Manuel Valenzuela Leyva
Supervisión	María Eugenia Sánchez Leal Alma Patricia Gallegos Borunda