



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**ACTUALIZACIÓN DE LA PLATAFORMA DE DESARROLLO DE TRANSFERENCIA
TECNOLÓGICA DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS,
IMPLEMENTADO EN LOS MÓDULOS DEL ÁREA ACADÉMICA, FACULTAD DE
INGENIERÍA, UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

José Carlos I Alonzo Colocho

Asesorado por Ing. Bayardo Salomón Martínez Ruiz

Guatemala, noviembre de 2023

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**ACTUALIZACIÓN DE LA PLATAFORMA DE DESARROLLO DE TRANSFERENCIA
TECNOLÓGICA DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS,
IMPLEMENTADO EN LOS MÓDULOS DEL ÁREA ACADÉMICA, FACULTAD DE
INGENIERÍA, UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA

POR

JOSÉ CARLOS I ALONZO COLOCHO

ASESORADO POR LOS ING. BAYARDO SALOMÓN MARTINEZ RUIZ

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, NOVIEMBRE DE 2023

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. José Francisco Gómez Rivera (a.i)
VOCAL II	Ing. Mario Renato Escobedo Martinez
VOCAL III	Ing. José Milton de León Bran
VOCAL IV	Ing. Kevin Vladimir Armando Cruz Lorente
VOCAL V	Br. Fernando José Paz González
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. José Francisco Gómez Rivera (a.i.)
EXAMINADORA	Inga. Floriza Felipa Ávila Pesquera de Medinilla
EXAMINADOR	Ing. Sergio Leonel Gómez Bravo
EXAMINADOR	Ing. Carlos Alfredo Azurdia Morales
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

**ACTUALIZACIÓN DE LA PLATAFORMA DE DESARROLLO DE TRANSFERENCIA
TECNOLÓGICA DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS,
IMPLEMENTADO EN LOS MÓDULOS DEL ÁREA ACADÉMICA, FACULTAD DE
INGENIERÍA, UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha 22 de marzo del 2023.


José Carlos L. Alonzo Colocho

Guatemala, 23 de septiembre de 2023

Ing. Oscar Argueta Hernández
Unidad de EPS
Facultad de Ingeniería
Universidad de San Carlos de Guatemala

Estimado Ing. Argueta.

Por este medio atentamente hacemos de su conocimiento que como asesor de la práctica del Ejercicio Profesional Supervisado (EPS) del estudiante universitario de la carrera de Ingeniería en Ciencias y Sistemas, **José Carlos I Alonzo Colocho**, con registro académico **201700965** y CUI **2897 10723 0101**, he revisado el informe final titulado **“ACTUALIZACIÓN DE LA PLATAFORMA DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS, IMPLEMENTADO EN LOS MÓDULOS DEL ÁREA ACADÉMICA, FACULTAD DE INGENIERÍA, UNIVERSIDAD DE SAN CARLOS DE GUATEMALA”**.

En tal virtud, lo damos por aprobado, solicitando darle el trámite respectivo. Sin otro particular, es grato suscribirnos.

Atentamente.


Bayardo Salomón Martínez Ruiz
Ingeniero en sistemas, Informática
Y Ciencias de la Computación
Colegiado No. 17,599
Vo.Bo. Asesor proyecto
Ing. Bayardo Martínez

Universidad de San Carlos de
Guatemala



Facultad de Ingeniería
Unidad de EPS

Guatemala, 26 de septiembre de 2023.
REF.EPS.DOC.389.09.2023.

Ing. Oscar Argueta Hernández
Director Unidad de EPS
Facultad de Ingeniería
Presente

Estimado Ingeniero Argueta Hernández:

Por este medio atentamente le informo que como Supervisora de la Práctica del Ejercicio Profesional Supervisado, (E.P.S) del estudiante universitario de la Carrera de Ingeniería en Ciencias y Sistemas, **José Carlos I Alonzo Colocho, Registro Académico 201700965 y CUI 2897 10723 0101** procedí a revisar el informe final, cuyo título es **ACTUALIZACIÓN DE LA PLATAFORMA DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS IMPLEMENTADO EN LOS MÓDULOS DEL ÁREA ACADÉMICA, FACULTAD DE INGENIERÍA, UNIVERSIDAD DE SAN CARLOS DE GUATEMALA.**

En tal virtud, **LO DOY POR APROBADO**, solicitándole darle el trámite respectivo.

Sin otro particular, me es grato suscribirme.

Atentamente,

“Id y Enseñad a Todos”



Inga. Floriza Felipa Ávila Pesquera de Medinilla
Supervisora de EPS
Área de Ingeniería en Ciencias y Sistemas

FFAPdM/RA

Universidad de San Carlos de
Guatemala



Facultad de Ingeniería
Unidad de EPS

Guatemala, 29 de septiembre de 2023.
REF.EPS.D.317.09.2023.

Ing. Carlos Gustavo Alonzo
Director Escuela de Ingeniería Ciencias y Sistemas
Facultad de Ingeniería
Presente

Estimado Ingeniero Alonzo:

Por este medio atentamente le envío el informe final correspondiente a la práctica del Ejercicio Profesional Supervisado, (E.P.S) titulado **ACTUALIZACIÓN DE LA PLATAFORMA DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS IMPLEMENTADO EN LOS MÓDULOS DEL ÁREA ACADÉMICA, FACULTAD DE INGENIERÍA, UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**, que fue desarrollado por el estudiante universitario **José Carlos I Alonzo Colocho, Registro Académico 201700965 y CUI 2897 10723 0101** quien fue debidamente asesorado por el Ing. Bayardo Salomón Martínez Ruiz y supervisado por la Inga. Floriza Felipa Ávila Pesquera de Medinilla.

Por lo que habiendo cumplido con los objetivos y requisitos de ley del referido trabajo y existiendo la aprobación del mismo por parte del Asesor y la Supervisora de EPS, en mi calidad de Director apruebo su contenido solicitándole darle el trámite respectivo.

Sin otro particular, me es grato suscribirme.

Atentamente,
"Id y Enseñad a Todos"


Ing. Oscar Argueta Hernández
Director Unidad de EPS

/ra



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala 3 de octubre de 2023

Ingeniero
Carlos Gustavo Alonzo
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Alonzo:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación-EPS del estudiante **JOSÉ CARLOS I ALONZO COLOCHO** carné **201700965** y CUI **2897 10723 0101**, titulado: **“ACTUALIZACIÓN DE LA PLATAFORMA DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS IMPLEMENTADO EN LOS MÓDULOS DEL ÁREA ACADÉMICA, FACULTAD DE INGENIERÍA, UNIVERSIDAD DE SAN CARLOS DE GUATEMALA”** y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,



Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA

LNG.DIRECTOR.225.EICCSS.2023

El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del Asesor, el visto bueno del Coordinador de área y la aprobación del área de lingüística del trabajo de graduación titulado: **ACTUALIZACIÓN DE LA PLATAFORMA DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS, IMPLEMENTADO EN LOS MÓDULOS DEL ÁREA ACADÉMICA, FACULTAD DE INGENIERÍA, UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**, presentado por: **José Carlos I Alonzo Colocho**, procedo con el Aval del mismo, ya que cumple con los requisitos normados por la Facultad de Ingeniería.

“ID Y ENSEÑAD A TODOS”



Ing. Carlos Gustavo Alonzo

Msc. Ing. Carlos Gustavo Alonzo

Director

Escuela de Ingeniería en Ciencias y Sistemas

Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, noviembre de 2023



LNG.DECANATO.OI.730.2023

El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al Trabajo de Graduación titulado: **ACTUALIZACIÓN DE LA PLATAFORMA DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS, IMPLEMENTADO EN LOS MÓDULOS DEL ÁREA ACADÉMICA, FACULTAD DE INGENIERÍA, UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**, presentado por: **José Carlos I Alonzo Colocho**, después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, autoriza la impresión del mismo.

IMPRÍMASE:


Ing. José Francisco Gómez Rivera
Decano a.i.



Guatemala, noviembre de 2023

JFGR/gaoc

ACTO QUE DEDICO A:

- | | |
|-------------------|---|
| Dios | Por permitirme completar esta meta en mi vida. |
| Mis padres | Stuardo Alonzo y Gloria Colocho, por su amor y apoyo incondicional. |
| Mi hermano | Saul Alonzo por brindar su apoyo. |

AGRADECIMIENTOS A:

Universidad de San Carlos de Guatemala	Por brindarme distintas experiencias y herramientas para sobrellevar los distintos retos presentados a lo largo de mi carrera universitaria.
Familia Alonzo Colocho	Por inspirarme a seguir adelante para lograr esta meta.
Ing. Bayardo Martinez	Por su apoyo en el desarrollo de este trabajo y sus aportes para la mejora de este.
Ing. Miguel Marín	Por brindarme un espacio para poder desarrollar mis habilidades técnicas en una pieza esencial en el funcionamiento de la Escuela de Ingeniería en Ciencias y Sistemas.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES	V
LISTA DE SÍMBOLOS	VII
GLOSARIO	IX
RESUMEN.....	XIII
OBJETIVOS.....	XV
INTRODUCCIÓN	XVII
1. MARCO TEÓRICO.....	1
1.1. Lenguaje de programación	1
1.2. Código fuente	1
1.3. Marco de trabajo.....	1
1.4. Revisión de código	2
1.5. Optimización de código	2
1.6. MVC.....	2
1.6.1. Modelo.....	2
1.6.2. Vista.....	3
1.6.3. Controlador	3
1.7. Repositorio de código.....	3
1.7.1. GitLab	3
1.8. Contenedores	3
1.8.1. Docker	4
2. FASE DE INVESTIGACIÓN.....	5
2.1. Institución	5
2.1.1. Misión	5

2.1.2.	Visión.....	6
2.2.	Reseña de la plataforma DTT	6
2.3.	Descripción de los problemas encontrados.....	6
2.3.1.	Código fuente sin estandarizar	6
2.3.2.	Código fuente muerto	7
2.3.3.	Consultas sobrecargadas.....	7
2.3.4.	Código fuente ilegible	7
2.3.5.	Funciones nativas desperdiciadas	8
2.3.6.	Complejidad algorítmica	8
2.3.7.	Herramientas desactualizadas	8
3.	FASE TÉCNICO PROFESIONAL	9
3.1.	Descripción del problema.....	9
3.2.	Estado actual de la plataforma.....	9
3.3.	Presentación de la solución al proyecto.....	10
3.3.1.	Versiones por utilizar	11
3.4.	Recursos involucrados en el proyecto.....	11
3.4.1.	Recursos humanos	11
3.4.2.	Recursos materiales.....	12
3.4.3.	Costo monetario	12
3.5.	Beneficios del proyecto	13
4.	FASE DE ENSEÑANZA APRENDIZAJE	15
4.1.	Transferencia de conocimientos para el desarrollo	15
4.1.1.	Estandarización.....	15
4.1.2.	Buenas prácticas.....	18
4.1.3.	Anotaciones sobre Web2Py	19
4.2.	Transferencia de conocimiento para el ambiente de desarrollo	19

CONCLUSIONES 21
RECOMENDACIONES 23
REFERENCIAS 25

ÍNDICE DE ILUSTRACIONES

FIGURAS

Figura 1.	Estandarización de consultas para código fuente SQL	17
Figura 2.	Estandarización de código fuente HTML	18

TABLAS

Tabla 1.	Costos del proyecto	13
Tabla 2.	Estandarización para código Python y JavaScript	16

LISTA DE SÍMBOLOS

Símbolo	Significado
Q	Quetzal
%	Porcentaje

GLOSARIO

Compilar	Proceso de traducción de un lenguaje de programación fuente hacia otro lenguaje o formato.
DBeaver Community	<i>Software</i> utilizado para conectarse a distintos tipos de bases de datos e interactuar con ellas.
Deprecado	Aplicable a un atributo o funcionalidad que indica que debe de ser abandonado ya sea por ser removido o porque obtuvo una nueva implementación, esto para evitar fallas técnicas en el futuro.
Desarrollador	Persona que tiene habilidades y conocimiento técnico relacionado al mundo del desarrollo de <i>software</i> .
DTT	Plataforma de Desarrollo de Transferencia Tecnológica.
Equipo de desarrollo	Conjunto de desarrolladores, que se encargan de la creación de un programa informático.
Estandarización	Lograr la uniformidad en la sintaxis del código fuente.
Hardware	Conjunto de componentes físicos que componen un computador.

IDE	Entorno de desarrollo integrado, editor de texto que contiene más herramientas para facilitar el desarrollo de código fuente.
Interpretar	Acción que realiza un programa informático, en la cual se analiza y ejecuta el código fuente en tiempo real.
Legibilidad	Estado del código fuente, con el cual este puede ser entendido con mayor facilidad por cualquiera que interactúe con él.
Mantenibilidad	Capacidad que tiene el código fuente para ser transformado de forma eficiente y efectiva, de acuerdo con nuevas necesidades.
Optimización	Método por el cual se determina el estado del código fuente y se mejora su rendimiento.
Plataforma	Sistema informático que sirve para ejecutar diversas tareas.
Reutilización	Técnica utilizada en el desarrollo de <i>software</i> para hacer uso de una porción de código fuente en otra, para evitar la repetición de esta.
Sintaxis	Este es el conjunto de reglas que definen aquellas combinaciones de símbolos que son aceptables para un lenguaje de programación.

Software	Conjunto de instrucciones, reglas y programas informáticos que se comunican con el <i>hardware</i> del computador.
Visual Studio Code	IDE propiedad de Microsoft, popular en la actualidad.

RESUMEN

El proyecto DTT tiene como objetivo entregar al estudiante una herramienta que le facilite la interacción de los estudiantes con la información académica relevante para sus fines, tales como horarios, notas, notificaciones de cursos, mensajes, entre otros. A su vez también brindar a catedráticos y tutores académicos una herramienta para poder mantener de una forma ordenada la gestión de los cursos que imparten y todo lo que esto conlleva.

Esta plataforma como ya fue mencionado es utilizada para diversos fines académicos, pero también con fines administrativos en los que se ve envuelta la Escuela de Ingeniería en Ciencias y Sistemas, con la cual interactúan aproximadamente 2,500 estudiantes, además de catedráticos, tutores académicos y personal administrativo.

Esta herramienta no se ha mantenido estancada en el tiempo, sino que ha ido evolucionando en diferentes niveles, siendo estos; cantidad de usuarios y cantidad de funcionalidades, tomando el enfoque de las funcionalidades estas han sido elaboradas por muchos desarrolladores diferentes a lo largo del tiempo, lo cual ha derivado en un sistema que posee un código fuente muy variado tanto en lógica como en estilado del mismo.

Tomando en cuenta lo anterior, el sistema actualmente presenta algunas perdidas de rendimiento derivado de la evolución descrita anteriormente, y esto deriva en que algunos usuarios prefieran usar plataformas alternativas en vez de una que se encuentra totalmente especializada para su uso dentro de la Escuela de Ingeniería en Ciencias y Sistemas.

OBJETIVOS

General

Mejorar el rendimiento general de la plataforma de Desarrollo de Transferencia Tecnológica de la Escuela de Ingeniería en Ciencias y Sistemas, para brindar una mejor experiencia a todos los usuarios que interactúan con ella.

Específicos

1. Optimizar el código fuente para mejorar el rendimiento individual de cada funcionalidad que conforman la plataforma.
2. Estandarizar el código para mantener un formato en futuras adiciones o actualizaciones de la plataforma.
3. Utilizar buenas prácticas de programación para corregir el código fuente que no haga uso de ellas.
4. Actualizar las versiones de las tecnologías empleadas en la plataforma.

INTRODUCCIÓN

Actualmente la Escuela de Ingeniería en Ciencias y Sistemas, atiende a 2500 estudiantes y 60 catedráticos, y gran parte de las actividades que son asociadas al área profesional del desarrollo de cada estudiante dentro de la carrera en la cual se encuentra inmiscuida la Escuela de Ingeniería en Ciencias y Sistemas, pasa por manos de la plataforma DTT; esta plataforma se encarga de llevar muchos de los trámites, peticiones, récords académicos y solicitudes que son útiles tanto para estudiantes, catedráticos y personal administrativo dentro de la escuela.

Sin embargo, debido al crecimiento exponencial con el que cuenta la escuela, trae por consecuencia un aumento en el uso de dicha plataforma tanto en lo que respecta a usuarios como en funcionalidades, por ello es necesario poder siempre brindar la mejor calidad de servicio posible y poder soportar dichas demandas; con lo cual, con el pasar de los años desde el momento de su creación, esta plataforma empieza a darnos indicios de algunos inconvenientes en momentos críticos, esto debido no solo a lo ya descrito, sino también a que las versiones de las tecnologías, junto con las configuraciones con la cual fue desarrollada, si bien no es obsoleta, para los tiempos que corren, puede no ser suficiente para satisfacer la demanda de capacidad y funcionalidades en la plataforma.

1. MARCO TEÓRICO

A continuación, se describirán los conceptos importantes que forman parte del escrito del presente documento.

1.1. Lenguaje de programación

Es aquel programa informático, que es utilizado para poder desarrollar más programas informáticos, estos son utilizados para administrar el comportamiento de los dispositivos lógicos y físicos de los distintos equipos informáticos que existen.

1.2. Código fuente

Es el texto que se encuentra distribuido en uno o varios archivos que posteriormente son compilados o interpretados por un lenguaje de programación junto a sus herramientas y nos da como resultado un programa informático que puede ser ejecutado por un usuario u otro programa informático.

1.3. Marco de trabajo

Un marco de trabajo en el contexto informático es conjunto de herramientas desarrolladas para un lenguaje de programación en concreto, el cual le brinda a los equipos de desarrollo o desarrolladores individuales que lo utilizan, una estructura previa para poder desarrollar de una forma más fácil un programa informático.

1.4. Revisión de código

Es una acción en la cual se revisa una fracción o la totalidad del código fuente posterior a su desarrollo, en esta acción, más que verificar el funcionamiento, se verifica el rendimiento, estandarización, legibilidad, reutilización y mantenibilidad de este (Martin, 2008).

1.5. Optimización de código

Teniendo en cuenta la definición presentada anteriormente, luego de hacer la revisión del código fuente, se dan los aportes necesarios al desarrollador o desarrolladores, para realizar las acciones correctivas de ser necesarias, para que con esto se puedan mejorar todos o algunos de los parámetros que conforman la revisión de código fuente (Martin, 2008).

1.6. MVC

Este es un patrón de diseño en el cual cada acción o conjunto de acciones asociadas a un fragmento de la lógica del proyecto en tres componentes: modelo, vista y controlador

1.6.1. Modelo

Es el encargado de manejar los datos y la lógica del proyecto, por lo que contiene diversos mecanismos para acceder a la información e interactuar con la misma.

1.6.2. Vista

Contiene todo el código fuente relacionado a la interacción, presentación y visualización de los datos, para que estos puedan ser utilizados por los usuarios finales.

1.6.3. Controlador

Es el encargado de direccionar a hacia otros componentes del modelo, como otras vista y modelos.

1.7. Repositorio de código

Este es un *software* encargado del almacenamiento centralizado de código fuente, al cual pueden acceder los desarrolladores para poder modificar o verificar el mismo, desde cualquier ubicación siempre y cuando tengan los permisos necesarios.

1.7.1. GitLab

Proyecto de *software* de código abierto, utilizado para el almacenaje de código fuente.

1.8. Contenedores

No deja de tener una analogía apegada a la realidad, es decir, un recipiente que es usado para almacenar distintos tipos de entidades adentro de él, en el campo informático, un contenedor es un programa informático el cual lleva adentro de si todas las herramientas necesarias para poder hacer

uso de un programa informático que también se encuentra y ejecuta adentro de este.

1.8.1. Docker

Docker es uno de los proyectos más famosos que brinda herramientas para el tratamiento de contenedores.

2. FASE DE INVESTIGACIÓN

2.1. Institución

La Escuela de Ingeniería en Ciencias y Sistemas, fue fundada en el año de 1970, a la cual se le adjudicó la carrera de Ingeniería en Ciencias y Sistemas; esta es una carrera que se encarga de desarrollar las habilidades necesarias para el manejo de la tecnología mayoritariamente a nivel de *software*, aunque también se tocan temas de *hardware*.

2.1.1. Misión

“Desarrollar en el alumno las competencias que garantizan el éxito en la construcción del conocimiento a través de los diferentes estilos de aprendizaje y fomentar la investigación permanente para permitir una mejor calidad de vida a la comunidad. Teniendo en cuenta las opciones del mercado actual del país (logística, administración, tecnología de la información, finanzas, contabilidad, comercio, entre otros), y también el mercado internacional, hacen una gran demanda global y competitividad en la actualidad.” (DTT, 2022).

2.1.2. Visión

“El estudiante de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala será reconocido como un profesional superior, en base al conocimiento incorporado en el currículo de estudios para capacitar a los estudiantes de manera integral, dándoles las herramientas adecuadas para su desarrollo profesional”. (DTT, 2022).

2.2. Reseña de la plataforma DTT

La plataforma surge en el año 2014, esto con el fin de simplificar muchas tareas, las cuales, hasta el momento de su creación, se llevaban en archivos físicos y haciendo uso de una completa presencialidad para la creación, seguimiento y finalización de los distintos procesos que tenían que realizar los estudiantes, catedráticos, auxiliares de cátedra y personal administrativo; la cual es una plataforma que ha visto un crecimiento enorme a lo largo del tiempo, tanto en funcionalidades, como en usuarios.

2.3. Descripción de los problemas encontrados

Al realizar el análisis de los distintos archivos de código fuente que integran el proyecto, se encuentran distintos tipos de problemas, los cuáles afectan el rendimiento de la plataforma en general.

2.3.1. Código fuente sin estandarizar

El código fuente presenta una combinación de convenciones de escritura que hacen que el código si bien no funcione mal o con bajo

rendimiento, si hace que los distintos desarrolladores que interactúan con este no puedan identificar de forma rápida los distintos elementos que lo conforman y lleguen a duplicar código por esta razón.

2.3.2. Código fuente muerto

Existen porciones de código que no se están empleando y solo consumen tiempo de procesamiento haciendo uso de los distintos recursos de la computadora y por ende ralentizando la entrega de la funcionalidad que se está utilizando.

2.3.3. Consultas sobrecargadas

Al ser un sistema que utiliza una base de datos para la permanencia de los datos que se manejan en la plataforma, también se hace uso de distintos tipos de consultas para poder obtener los datos de esta, sin embargo, existen muchas consultas que son obtenidas con todos los campos de las tablas que interactúan en esta, pero que al momento de hacer uso de los campos obtenidos no son utilizados todos estos, derivando en que se consultan más datos de los necesarios.

2.3.4. Código fuente ilegible

En algunos fragmentos del código fuente, se pueden observar instrucciones que, si bien son funcionales, para un desarrollador que está trabajando en la plataforma pueden resultar confusas y tediosas al momento de analizarlo u optimizarlo, logrando con esto que el desarrollador pierda tiempo entiendo el código y pueda derivar incluso en atrasos.

2.3.5. Funciones nativas desperdiciadas

Tanto en el lenguaje de programación como en el marco de trabajo, existen funciones que ya están implementadas de forma nativa en estos; sin embargo, en el código fuente hay algunas porciones que fueron implementadas por los distintos desarrolladores de forma manual, haciendo con esto que se tenga código innecesario y que además no se encuentra optimizado como el código nativo.

2.3.6. Complejidad algorítmica

Una parte del proyecto presenta pérdidas de rendimiento debido a que existen fracciones del código fuente que hacen uso de procesamiento innecesario, debido a los algoritmos implementados, ya fuera por problemas externos o internos al desarrollador, los propios algoritmos o la implementación de estos, no son necesariamente los óptimos.

2.3.7. Herramientas desactualizadas

El código fuente en general se encuentra desarrollado sobre una versión tanto del marco de trabajo como el lenguaje de programación que si bien no es obsoleta, presenta ciertas carencias al respecto de versiones más actuales de las mismas, las cuales han mejorado con el tiempo, tanto en funcionalidades como en rendimiento además, que nos brindan mayores facilidades al momento de desarrollar *software* con ellas y una mayor compatibilidad y variedad de librerías externas que pueden ser necesarias en futuras implementaciones a la plataforma.

3. FASE TÉCNICO PROFESIONAL

3.1. Descripción del problema

La plataforma DTT, es un proyecto que, desde su fundación en el año 2014, ha sido intervenido por una gran variedad de desarrolladores, desde estudiantes que se encuentran en el momento de las practicas finales hasta estudiantes que ya culminaron sus estudios y se encuentran en la fase de sus trabajos de graduación; derivando esto en que exista una amplia diversidad de criterios y conocimientos al momento de desarrollar las funcionalidades que cada uno implemento; sumado a esto, desde su fundación, los distintos elementos que conforman la plataforma como lo son el marco de trabajo, lenguaje de programación y librerías externas no han sido actualizadas.

3.2. Estado actual de la plataforma

La plataforma se encuentra elaborada en el lenguaje de programación Python, en su versión 2.7, haciendo uso de un marco de trabajo para desarrollo de proyectos web haciendo uso del modelo de diseño MVC, llamado web2py, en su versión 2.9.5.

El almacenamiento del código fuente se realiza en un repositorio en la plataforma web GitLab, en el cual se dan los accesos necesarios a los distintos desarrolladores para poder trabajar sobre el código fuente.

Para poder desarrollar de manera local los requerimientos solicitados a los desarrolladores, se hace uso de contenedores de Docker, los cuales hacen

una réplica del ambiente utilizado en el sitio oficial de la plataforma, para enfocarse únicamente en el desarrollo dejando de lado la replicación del ambiente de forma manual.

3.3. Presentación de la solución al proyecto

Para dar solución a los distintos problemas presentados, se pretende realizar una migración completa a nivel de las tecnologías que integran la plataforma, siempre haciendo uso de las mismas tecnologías como base, pero en versiones más recientes.

Se harán uso de las mismas tecnologías para no afectar la compatibilidad existente y la funcionalidad, así como también al conocimiento histórico sobre la plataforma del cual hace uso el encargado de esta para poder brindar orientación al momento de integrar nuevos elementos al crecimiento de la plataforma.

El hacer uso de versiones más recientes de las tecnologías nos brinda las bondades que estas traen, tales como mayor eficiencia, más funcionalidades nativas, mayor soporte y tener acceso a más librerías externas.

Dicha migración contempla que todas las funcionalidades que contiene la plataforma no se vean afectadas en lo absoluto a nivel de lo que podría ver y utilizar un usuario final.

3.3.1. Versiones por utilizar

Para realizar la migración se actualizará la versión tanto del marco de trabajo, como del lenguaje de programación; el marco de trabajo pasara de la versión 2.9.5 a la 2.23.0 y el lenguaje de programación pasara de la versión 2.7 a la 3.7.4; estas serán las únicas con versiones específicas ya que las tecnologías web “HTML5, CSS3 y JavaScript ECMAScript” dependen del navegador y su versión debido a la integración que estos realizan para las funcionalidades de estos lenguajes; así que únicamente se actualizarán aquellas funciones que se encuentren deprecadas, tomando en cuenta los navegadores web más comúnmente utilizados.

3.4. Recursos involucrados en el proyecto.

Para la elaboración de la migración, se utilizarán tanto recursos humanos como materiales, por lo que, a continuación, se detallarán los mismos.

3.4.1. Recursos humanos

Los recursos humanos que se utilizarán para las distintas fases del desarrollo del proyecto son los siguientes:

- Asesor institucional: Ing. Miguel Marín de León
- Asesor de la Escuela de Ingeniería en Ciencias y Sistemas: Ing. Bayardo Salomón Martínez Ruiz.
- Supervisor de EPS: Inga. Floriza Ávila
- Encargado de la institución: Msc. Ing. Carlos Gustavo Alonzo
- Desarrollador: José Carlos I Alonzo Colocho

3.4.2. Recursos materiales

Debido a que como desarrollador trabaje de forma remota, tanto para la migración de la plataforma en si, como también para las distintas reuniones programadas y no programadas para la presentación de avances y revisiones; utilice mis propios recursos, los cuales son:

- Servicio de internet residencial.
- Computadora personal.
- Servicio de energía eléctrica.
- Mobiliario estándar de oficina.
- *Software* para desarrollo (Visual Studio Code, Docker, DBeaver Community, distintos navegadores web)

3.4.3. Costo monetario

El costo monetario del proyecto asciende a un total de Q 106,634.00; el cual será desglosado en la siguiente tabla.

Tabla 1.

Costos del proyecto

Recurso	Cantidad	Costo unitario	Subtotal
Desarrollador	1 desarrollador * 6 meses	Q 8,000.00	Q 48,000.00
Asesor	1 asesor de escuela * 6 meses	Q 4,000.00	Q 24,000.00
Asesor	1 asesor de institución * 6 meses	Q 4,000.00	Q 24,000.00
Equipo de computo	1 computadora	Q 7,000.00	Q 7,000.00
Servicio de internet residencial	Servicio * 6 meses	Q 289.00	Q 1,734.00
Servicio de energía eléctrica	Servicio * 6 meses	Q 150.00	Q 900.00
Mobiliario estándar de oficina	1 silla, 1 mesa	Q 1,000.00	Q 1,000.00
Total			Q 106,634.00

Nota. Detalle de los costos para la realización de la migración. Elaboración propia.

3.5. Beneficios del proyecto

Al momento de dar por finalizado el proyecto se podrán apreciar ciertos beneficios para todos los usuarios que interactúen con la plataforma tanto a nivel de usuario final, como de usuario desarrollador.

Para los usuarios finales, se mejorará la experiencia de usuario al momento que estos interactúen con la plataforma, obteniendo como resultado un aumento en las velocidades de respuesta hacia las peticiones que hacen en la plataforma, así como también evitar aquellos momentos en los que la plataforma decaía en rendimiento.

Y para los usuarios que interactúen en el desarrollo, la experiencia que tengan al desarrollar se verá mejorada, presentando así un código fuente más

optimo, ordenado, limpio, estandarizado y entendible para poder usarlo de referencia; además de ellos también se les brindará un manual en el cual se detallarán los estándares que deberán aplicarse al momento de desarrollar, así como distintas ayudas respecto al uso del marco de trabajo y la lógica de la plataforma para facilitar y agilizar el desarrollo

4. FASE DE ENSEÑANZA APRENDIZAJE

Como ya hemos mencionado anteriormente el sistema DTT, está en continua evolución e interactúa con diferentes desarrollares que cuentan con distintas capacidades y conocimientos por las etapas a nivel académico en las que se encuentran al momento de la interacción, por ende, es necesario que se realicen lineamientos para que dichos desarrolladores puedan llevar a cabo sus tareas en la plataforma con la máxima calidad posible y una mayor facilidad al momento de integrarse.

4.1. Transferencia de conocimientos para el desarrollo

Para poder realizar lo antes mencionado, se creará un manual el cual contendrá las distintas recomendaciones y lineamientos que se quieren transmitir hacia los futuros desarrolladores, el cual se dividirá de varias partes.

4.1.1. Estandarización

Para esta parte del manual se describirán los lineamientos que se deben de seguir al momento de nombrar las distintas partes del código fuente, como serán según lo descrito a continuación.

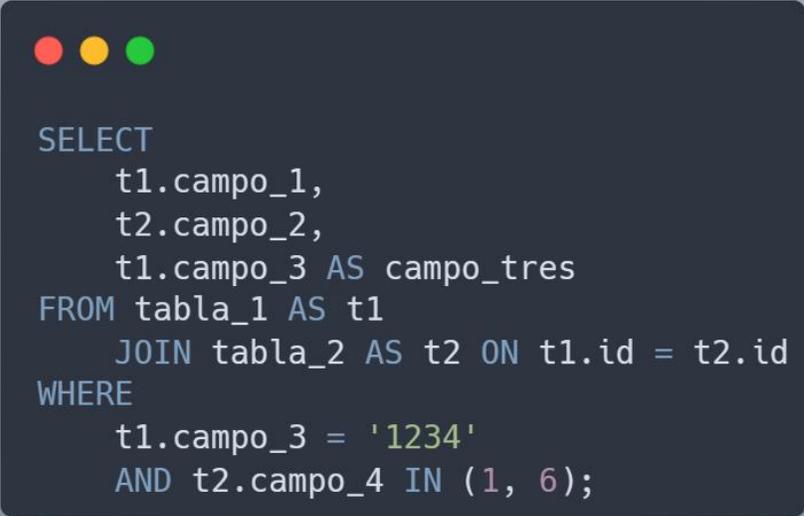
Tabla 2.*Estandarización para código Python y JavaScript*

Tipo de código fuente	Estilado	Ejemplo
Funciones	Snake case	funcion / mi_funcion
Variables	Snake case	variable / mi_variable
Clases	Camel case	Clase / MiClase
Métodos	Snake case	metodo / mi_metodo
Constantes	Snake case mayúscula	CONSTANTE / MI_CONSTANTE
Módulos	Snake case	modulo .py .js / mi_modulo .py .js
Paquetes	Minúscula	paquete / mipaquete

Nota. Detalle de los distintos tipos de estildados que se deben emplear en el código fuente de Python y JavaScript. Elaboración propia, realizado con Microsoft Word.

Figura 1.

Estandarización de consultas para código fuente SQL

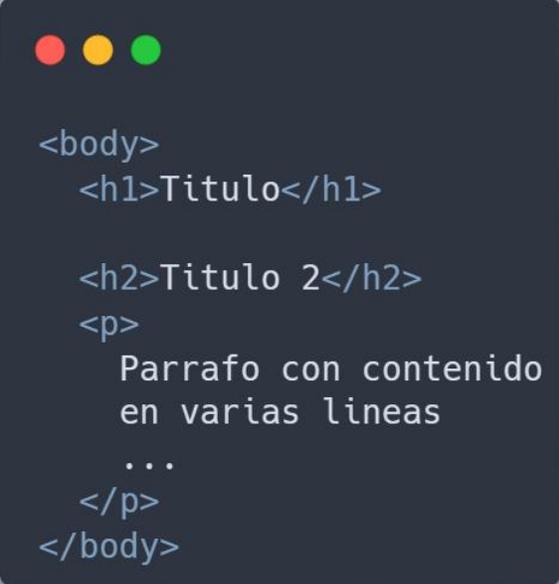


```
SELECT
    t1.campo_1,
    t2.campo_2,
    t1.campo_3 AS campo_tres
FROM tabla_1 AS t1
    JOIN tabla_2 AS t2 ON t1.id = t2.id
WHERE
    t1.campo_3 = '1234'
    AND t2.campo_4 IN (1, 6);
```

Nota. Consulta de ejemplo para el estilado y formateo de código fuente SQL. Elaboración propia, realizado con Carbon.

Figura 2.

Estandarización de código fuente HTML



```
<body>
  <h1>Titulo</h1>

  <h2>Titulo 2</h2>
  <p>
    Parrafo con contenido
    en varias lineas
    ...
  </p>
</body>
```

Nota. Código fuente en HTML de ejemplo para el estilado y formateado del mismo. Elaboración propia, realizado con Carbon.

4.1.2. Buenas prácticas

En este apartado se darán distintos consejos en base a los problemas encontrados en la plataforma y al conocimiento adquirido para poder brindar herramientas para mejorar la calidad del código fuente y por ende la calidad del producto final.

4.1.3. Anotaciones sobre Web2Py

Para el apartado final del manual se hará de conocimiento para el lector de forma general como trabajar en el marco de trabajo, los métodos más recurrentes; utilizados para el desarrollo en dicho marco de trabajo, además de los comportamientos de algunas funciones y métodos para emplearlos de manera eficaz y evitar el desperdicio de recursos.

4.2. Transferencia de conocimiento para el ambiente de desarrollo

Se dará un pequeño manual en el cual se detallará como trabajar con las herramientas de desarrollo de forma local y como corregir posibles errores que podrían aparecer al momento del desarrollo.

CONCLUSIONES

1. Se logro optimizar el código fuente y con ello se mejora el rendimiento de cada funcionalidad individual que este tiene.
2. El código fuente fue formateado y estilizado para seguir los estándares que se describieron en este mismo documento.
3. Se corrigieron malas prácticas en el código fuente para evitar su replicación en futuras adiciones y que no sean tomadas como referencia.
4. Fueron actualizadas las distintas versiones de las tecnologías hacia versiones más recientes, descritas en este documento y de esa manera mantener una plataforma con herramientas actualizadas.

RECOMENDACIONES

1. Tener periódicamente a personal externo al desarrollador o desarrolladores involucrados que puedan verificar la calidad del código fuente generado y aplicar las medidas necesarias para su optimización.
2. Seguir los parámetros que fueron definidos en los manuales entregados al encargado de la plataforma, para poder mantener un código fuente estandarizado y formateado según lo ya establecido.
3. Tomar en cuenta las anotaciones y recomendaciones descritas en el manual entregado al encargado de la plataforma y al propio código fuente para analizar las distintas implementaciones de este, para evitar aplicar malas prácticas en sus propias implementaciones.
4. Considerar un plazo prudente para analizar las versiones de las tecnologías actuales descritas en este documento y poder contrastarlas con las futuras versiones para poder definir si es necesario realizar una actualización de estas.

REFERENCIAS

- DTT. (10 de agosto de 2022). *DDT | Escuela de Ciencias y Sistemas*.
https://dtc-ecys.org/about_us
- Martin, R. (2008). *Clean code: A handbook of agile software craftsmanship*.
Prentice Hall.

