



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**


Miguel Marín de León
INGENIERO EN CIENCIAS Y SISTEMAS
COL. 8312

José Carlos I Alonzo Colocho
Asesorado por Ing. Bayardo Salomón Martínez Ruiz

Guatemala, agosto 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 E ING.
MIGUEL MARÍN DE LEÓN

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, AGOSTO DE 2023

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Inga. Aurelia Anabela Cordova Estrada
VOCAL I	Ing. José Francisco Gómez Rivera
VOCAL II	Ing. Mario Renato Escobedo Martínez
VOCAL III	Ing. José Milton de León Bran
VOCAL IV	Br. Kevin Vladimir Cruz Lorente
VOCAL V	Br. Fernando José Paz González
SECRETARIA	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	De ese período
EXAMINADOR	Examinador 1
EXAMINADOR	Examinador 2
EXAMINADOR	Examinador 3
SECRETARIO	De ese período

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 3 de febrero del 2023.

José Carlos I Alonzo Colocho

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.
Mi familia	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	IV
LISTA DE SÍMBOLOS	VII
GLOSARIO	IX
RESUMEN	XII
OBJETIVOS.....	XIV
INTRODUCCIÓN	XVII
1. MARCO TEÓRICO.....	19
1.1. Lenguaje de programación	19
1.2. Código fuente	19
1.3. Marco de trabajo.....	19
1.4. Revisión de código	20
1.5. Optimización de código	20
1.6. MVC.....	20
1.6.1. Modelo.....	20
1.6.2. Vista.....	20
1.6.3. Controlador.....	21
1.7. Repositorio de código	21
1.8. Contenedores	21
1.8.1. Docker	21
2. FASE DE INVESTIGACIÓN	23
2.1. Institución	23
2.1.1. Misión	23
2.1.2. Visión.....	23
2.2. Reseña de la plataforma DTT.....	24
2.3. Descripción de los problemas encontrados	24

2.3.1.	Código fuente sin estandarizar	24
2.3.2.	Código fuente muerto	25
2.3.3.	Consultas sobrecargadas	25
2.3.4.	Código fuente ilegible	25
2.3.5.	Funciones nativas desperdiciadas	25
2.3.6.	Complejidad algorítmica	26
2.3.7.	Herramientas desactualizadas	26
3.	FASE TÉCNICO PROFESIONAL	28
3.1.	Descripción del problema	28
3.2.	Estado actual de la plataforma	28
3.3.	Presentación de la solución al proyecto	29
3.3.1.	Versiones por utilizar	29
3.4.	Recursos involucrados en el proyecto	30
3.4.1.	Recursos humanos	30
3.4.2.	Recursos materiales	30
3.4.3.	Costo monetario	31
3.5.	Beneficios del proyecto	31
	CONCLUSIONES	34
	RECOMENDACIONES	36
	REFERENCIAS	38
	APÉNDICES	39
	ANEXOS	40

ÍNDICE DE ILUSTRACIONES

FIGURAS

Figura 1.	Título de la figura 1 ¡Error! Marcador no definido.
Figura 2.	Título de figura 2 debe estar en una línea¡Error! Marcador no definido.
Figura 3.	Título de figura 3..... ¡Error! Marcador no definido.
Figura 4.	Título de figura 4..... ¡Error! Marcador no definido.
Figura 5.	Título de figura 5..... ¡Error! Marcador no definido.
Figura 6.	Título de figura 6..... ¡Error! Marcador no definido.
Figura 7.	Título de figura 7..... ¡Error! Marcador no definido.
Figura 8.	Título de figura 8..... ¡Error! Marcador no definido.
Figura 9.	Título de figura 9..... ¡Error! Marcador no definido.
Figura 10.	Título de figura 10..... ¡Error! Marcador no definido.
Figura 11.	Título de figura 11..... ¡Error! Marcador no definido.
Figura 12.	Título de figura 12..... ¡Error! Marcador no definido.
Figura 13.	Título de figura 13..... ¡Error! Marcador no definido.
Figura 14.	Título de figura 14..... ¡Error! Marcador no definido.
Figura 15.	Título de figura 15..... ¡Error! Marcador no definido.
Figura 16.	Título de figura 16..... ¡Error! Marcador no definido.
Figura 17.	Título de figura 17..... ¡Error! Marcador no definido.
Figura 18.	Título de figura 18..... ¡Error! Marcador no definido.
Figura 19.	Título de figura 19..... ¡Error! Marcador no definido.
Figura 20.	Título de figura 20..... ¡Error! Marcador no definido.

TABLAS

Tabla 1.	Costos del proyecto	31
-----------------	---------------------------	----

LISTA DE SÍMBOLOS

Símbolo

Significado

GLOSARIO

Desarrollador	Persona que tiene habilidades y conocimiento técnico relacionado al mundo del desarrollo de software.
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.
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 software 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.
Deprecado	Aplicable a un atributo o funcionalidad que indica que debe de ser abandonado ya sea por ser removido o que obtuvo una nueva implementación, esto para evitar fallas técnicas en el futuro.
IDE	Entorno de desarrollo integrado, editor de texto que contiene mas herramientas para facilitar el desarrollo.
Visual Studio Code	IDE propiedad de Microsoft, popular en la actualidad.
DBeaver Community	Software utilizado para conectarse a distintos tipos de bases de datos e interactuar con ellas.

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, etc. 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 tanto académicos como administrativos en los que se ve envuelta la Escuela de Ingeniería en Ciencias y Sistemas, con la cual interactúan aproximadamente 2500 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 dos; 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 pérdidas 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

Eficientar 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 dicha escuela, por el creciente uso de la tecnología, trae por consecuencia el 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 al 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 la versión de la tecnología 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.

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.

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 encargo de manejar los datos y la lógica del proyecto

1.6.2. Vista

Se encarga de presentar los datos y la interacción con el usuario final

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.

1.8. Contenedores

No deja de tener una analogía apegada a la realidad es decir, un recipiente que es usado para almacenar distintos tipos de objetos adentro de él, en el campo informático un contenedor es un software el cual lleva adentro de si todas las herramientas necesarias para poder hacer uso de otro software o conjunto de software que también se encuentra en el contenedor.

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, etc.), 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 que con todos los campos de las tablas que interactúan en la consulta, pero que al momento de hacer uso de campos no son utilizados todos, derivando en que se consultan más datos de los necesarios haciendo esto un proceso mas tardado de manera innecesaria.

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 esta 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 perdidas 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 mas actuales de la misma, 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 del ejercicio práctico supervisado o la tesis; 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 framework, 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 framework 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 mas recientes de las tecnologías nos brinda las bondades que estas traen, tales como mayor eficiencia, mas 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 framework, como del lenguaje de programación; el framework 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 59,634.00; 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
Equipo de computo	1 computadora	Q 8,000.00	Q 8,000.00
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 59,634.00

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 óptimo, 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 framework y la lógica de la plataforma para facilitar y agilizar el desarrollo.

CONCLUSIONES

RECOMENDACIONES

REFERENCIAS

Atlassian. (10 de agosto de 2022). *¿Qué es Scrum?*.
<https://www.atlassian.com/es/agile/scrum>

Coghlan, N., Rossum, G., Warsaw, B. (05 de julio de 2001). *PEP 8 - Style Guide for Python Code* | [peps.python.org](https://peps.python.org/pep-0008/). <https://peps.python.org/pep-0008/>

DTT. (10 de agosto de 2022). *DTT | Escuela de Ciencias y Sistemas*. https://dttecys.org/about_us

Martin, R. (2008). *Clean code: A handbook of agile software craftsmanship*. Prentice Hall.

Mulone, M., Maldonado, J., Pierro, M. (10 de septiembre de 2022). *Web2py – Prólogo*.
<http://www.web2py.com/books/default/chapter/36>

W3Schools. (19 de septiembre de 2022). *HTML Style Guide and Coding Conventions*.
https://www.w3schools.com/html/html5_syntax.asp

APÉNDICES

ANEXOS