



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

**MANTENIMIENTO Y OPTIMIZACIÓN DE LA PLATAFORMA WEB DE DESARROLLO DE
TRANSFERENCIA TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS
Y SISTEMAS DE LA FACULTAD DE INGENIERÍA**

Carlos Augusto Valdez Siguenza

Asesorado por el Ing. Luis Alberto Arias Solórzano

Guatemala, octubre de 2019

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**MANTENIMIENTO Y OPTIMIZACIÓN DE LA PLATAFORMA WEB DE DESARROLLO DE
TRANSFERENCIA TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS
Y SISTEMAS DE LA FACULTAD DE INGENIERÍA**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA

POR

CARLOS AUGUSTO VALDEZ SIGUENZA

ASESORADO POR EL ING. LUIS ALBERTO ARIAS SOLÓRZANO

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, OCTUBRE DE 2019

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. Luis Diego Aguilar Ralón
VOCAL V	Br. Christian Daniel Estrada Santizo
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Pedro Antonio Aguilar Polanco
EXAMINADORA	Inga. Floriza Felipa Ávila Pesquera
EXAMINADOR	Ing. Sergio Leonel Gómez Bravo
EXAMINADOR	Ing. Marlon Antonio Pérez Türk
SECRETARIA	Inga. Lesbia Magalí Herrera López

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:

MANTENIMIENTO Y OPTIMIZACIÓN DE LA PLATAFORMA WEB DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS DE LA FACULTAD DE INGENIERÍA

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha 24 de julio de 2018.

A handwritten signature in blue ink, appearing to be 'CAV', is written over a circular stamp or seal.

Carlos Augusto Valdez Siguenza

Guatemala 18 de Julio de 2019.

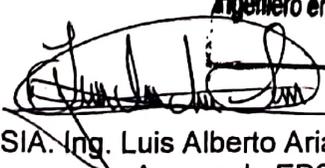
Ing. Oscar Argueta Hernández
Director Unidad EPS
Facultad de Ingeniería
Universidad de San Carlos de Guatemala

Ingeniero,

Por medio de la presente hago de su conocimiento que el estudiante CARLOS AUGUSTO VALDEZ SIGUENZA que se identifica con DPI CUI 1636867260101 y numero de camé 200217159 de la Facultad de Ingeniería USAC, de la Carrera de Ingeniería en Ciencias y Sistemas, desarrollo el proyecto: "MANTENIMIENTO Y OPTIMIZACIÓN DE LA PLATAFORMA WEB DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS DE LA FACULTAD DE INGENIERÍA", el cual he asesorado y puedo afirmar que el mismo se encuentra finalizado satisfactoriamente.

Atentamente.

Luis Alberto Arias Solórzano
Ingeniero en Ciencias y Sistemas
Colegiado 10402- USAC


MSIA. Ing. Luis Alberto Arias Solórzano
Asesor de EPS
Colegiado no. 10402

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA

UNIDAD DE EPS

Guatemala, 19 de julio de 2019.
REF.EPS.DOC.514.07.2019.

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, **Carlos Augusto Valdez Siguenza, Registro Académico 200217159 y CUI 1636 86726 0101** procedí a revisar el informe final, cuyo título es **MANTENIMIENTO Y OPTIMIZACIÓN DE LA PLATAFORMA WEB DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA, (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS DE LA FACULTAD DE INGENIERÍA**

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



Guatemala, 19 de julio de 2019.
REF.EPS.D.265.07.2019.

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

Estimado Ingeniero Alonzo:

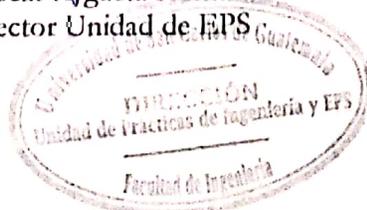
Por este medio atentamente le envío el informe final correspondiente a la práctica del Ejercicio Profesional Supervisado, (E.P.S) titulado **MANTENIMIENTO Y OPTIMIZACIÓN DE LA PLATAFORMA WEB DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA, (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS DE LA FACULTAD DE INGENIERÍA**, que fue desarrollado por el estudiante universitario **Carlos Augusto Valdez Siguenza, Registro Académico 200217159 y CUI 1636 86726 0101** quien fue debidamente asesorado por el Ing. Luis Alberto Arias Solórzano 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, 31 de julio de 2019

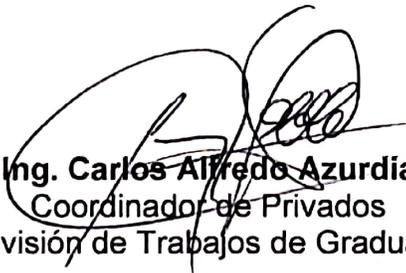
Ingeniero
Carlos Gustavo Alonzo
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Alonzo:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación-EPS del estudiante **CARLOS AUGUSTO VALDEZ SIGUENZA** camé 200217159 y CUI 1636 86726 0101, titulado: **"MANTENIMIENTO Y OPTIMIZACIÓN DE LA PLATAFORMA WEB DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA, (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS DE LA FACULTAD DE INGENIERÍA"** 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 Azurdía
Coordinador de Privados
y Revisión de Trabajos de Graduación



UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA EN
CIENCIAS Y SISTEMAS
TEL: 24767644

ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS

*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 con el visto bueno del revisor y del Licenciado en Letras, del trabajo de graduación **“MANTENIMIENTO Y OPTIMIZACIÓN DE LA PLATAFORMA WEB DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS DE LA FACULTAD DE INGENIERÍA”**, realizado por el estudiante, **CARLOS AUGUSTO VALDEZ SIGUENZA** aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑAD A TODOS”

Msc. Ing. Carlos Gustavo Alonzo

Director

Escuela de Ingeniería en Ciencias y Sistemas



Guatemala, 11 de octubre de 2019



La Decana 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: **MANTENIMIENTO Y OPTIMIZACIÓN DE LA PLATAFORMA WEB DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS DE LA FACULTAD DE INGENIERÍA**, presentado por el estudiante universitario: **Carlos Augusto Valdez Siguenza**, y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, se autoriza la impresión del mismo.

IMPRÍMASE.


Inga. Aurelia Anabela Cordova Estrada★
Decana



Guatemala, Octubre de 2019

/cc

ACTO QUE DEDICO A:

Dios	Permitirme completar esta meta en mi vida.
Mis padres	Carlos Valdez y Lidia Siguenza, por su amor y apoyo incondicional.
Mis hermanos	Amanda y Vinicio Valdez, por brindarme su ayuda siempre.

AGRADECIMIENTOS A:

Universidad de San Carlos de Guatemala	Por permitirme desarrollar mis conocimientos.
Facultad de Ingeniería	Por brindarme una formación académica.
Mi familia	Por inspirarme a seguir adelante para poder triunfar en la vida y lograr alcanzar esta meta.
Mis amigos de la Facultad	Por su apoyo y camaradería a lo largo de mi carrera.
Ing. Luis Arias	Por alentarme a continuar en la carrera y el asesoramiento de este trabajo de graduación.
Ing. Miguel Marín	Por asesorarme en el desarrollo del trabajo de graduación.
Iveth Flores	Por su respaldo y cariño.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES	V
LISTA DE SÍMBOLOS	VII
GLOSARIO	IX
RESUMEN	XI
OBJETIVOS.....	XIII
INTRODUCCIÓN	XV
1. MARCO TEÓRICO	1
1.1. Herramientas por utilizar	1
1.1.1. Lenguajes de programación.....	1
1.1.1.1. Python	1
1.1.1.2. Javascript	1
1.1.2. Manejador de código fuente.....	2
1.1.3. Control de versiones	2
1.1.3.1. GIT	2
1.1.3.2. Repositorio	2
1.1.3.3. Versión	3
1.1.4. Base de datos	3
1.1.4.1. Sistema de gestión de base de datos	3
1.1.4.2. SQL	3
1.1.4.3. MYSQL.....	4
1.1.5. Sistema operativo.....	4
1.2. Metodologías de desarrollo	4
1.2.1. Desarrollo ágil	5

	1.2.1.1.	Metodología de desarrollo XP (<i>Extreme Programming</i>).....	5
1.3.		Arquitectura del sistema.....	5
	1.3.1.	Descripción de la arquitectura.....	5
	1.3.2.	Elementos de la arquitectura	6
2.		FASE DE INVESTIGACIÓN	9
2.1.		Antecedentes de la empresa	9
	2.1.1.	Reseña histórica	9
	2.1.2.	Misión.....	10
	2.1.3.	Visión	10
	2.1.4.	Servicios que realiza.....	10
2.2.		Descripción de las necesidades.....	11
	2.2.1.	Ayuda y soporte	11
	2.2.1.1.	Módulo de reclutamiento de DTT.....	12
	2.2.1.2.	Módulo de Quiz.....	12
	2.2.1.3.	Módulo de Android.....	12
	2.2.1.4.	Módulo de visualización e ingreso de notas	12
	2.2.1.5.	Módulo 360	13
	2.2.1.6.	Módulo de envío de correos	13
	2.2.1.7.	Módulo de menú, recursos y horarios..	13
	2.2.1.8.	Módulo de solvencia de prácticas.....	14
	2.2.1.9.	Módulo Imprimir Reporte (botón Imprimir).....	14
	2.2.1.10.	Módulo de <i>login</i>	14
	2.2.1.11.	Módulo DSI	14
	2.2.1.12.	Levantar los ambientes en el nuevo servidor virtualizado	15

2.2.2.	Gestión de control de cambios en el ambiente de producción.....	15
2.3.	Priorización de las necesidades.....	15
3.	FASE TÉCNICO PROFESIONAL.....	19
3.1.	Descripción del proyecto.....	19
3.2.	Investigación preliminar para la solución del proyecto.....	19
3.3.	Presentación de la solución al proyecto.....	20
3.3.1.	Implementación de control de versiones.....	21
3.3.2.	Ayuda y soporte de errores de la aplicación.....	22
3.3.3.	Traspaso de plataforma a nuevo servidor.....	24
3.4.	Costos del proyecto.....	25
3.4.1.	Recursos humanos.....	25
3.4.2.	Recursos materiales.....	25
3.4.3.	Costos.....	26
3.5.	Beneficios del proyecto.....	26
4.	FASE DE ENSEÑANZA Y APRENDIZAJE.....	29
4.1.	Perfil del usuario.....	29
4.2.	Transferencia de conocimientos.....	30
4.2.1.	Uso de las herramientas.....	30
4.2.2.	Casos prácticos de utilización.....	30
4.3.	Metodología de aprendizaje.....	30
4.3.1.	Sesiones de aprendizaje.....	31
4.4.	Capacitación propuesta.....	31
4.5.	Material elaborado.....	32
5.	DESCRIPCIÓN PROCESOS DEL SISTEMA Y SEGURIDAD.....	33
5.1.	Descripción de los procesos automáticos del servidor.....	33

5.1.1.	Scheduler	33
5.1.2.	Regeneración de certificado SSL.....	34
5.1.3.	Copia de seguridad semanal	35
5.2.	Seguridad en el servidor de aplicaciones	36
5.2.1.	Descripción de la seguridad del servidor de aplicaciones	36
5.2.2.	Descripción de la seguridad de base de datos	37
5.3.	Seguridad en el repositorio de versionamiento.....	38
CONCLUSIONES.....		41
RECOMENDACIONES		43
BIBLIOGRAFÍA.....		45
APÉNDICES		47

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Arquitectura global	6
2.	Componentes internos.....	7
3.	Forma de trabajo de GIT.....	22
4.	Diagrama de seguridad de <i>firewall</i>	36
5.	Formulario de generación de <i>password</i>	38
6.	Esquema de seguridad de GIT	39

TABLAS

I.	Prioridad de desarrollo.....	16
II.	Costos del proyecto	26
III.	Capacitaciones propuestas.....	31

LISTA DE SÍMBOLOS

Símbolo	Significado
Q	Quetzal

GLOSARIO

Caché	Memoria temporal de acceso rápido, usada para almacenar y consultar datos, frecuentemente utilizados.
DTT	Plataforma de Desarrollo de Transferencia Tecnológica.
Hardware	Conjunto de características físicas que integran un computador o dispositivo móvil.
MARIADB	Base de datos persistente, ampliamente difundida y utilizada en proyectos de desarrollo de software.
Módulo	En programación, un módulo es una porción de un programa de ordenador.
Plataforma	Sistema informático que sirve para ejecutar diversas tareas.
REDIS	Base de datos de caché utilizada para guardar datos no persistentes.

RESUMEN

El objetivo del proyecto DTT es entregar al estudiante una herramienta que le facilite la interacción de los estudiantes con la información académica relevante para sus fines, como horarios, notas, etc. además de brindar a los catedráticos y tutores una herramienta para control ordenada que provee muchas funciones útiles para la gestión de cursos

Es una plataforma utilizada para diversos fines en la escuela de Ciencias y sistemas, con la cual interactúan más de 1500 estudiantes, además de catedráticos y tutores académicos.

Entre las funciones del sistema se encuentran el ingreso de notas, generación de constancias de prácticas, generación de Quiz, etc. Otorga una gama de funciones que lo hacen un sistema robusto.

No obstante, presenta algunas fallas, por lo cual, en la actividad se subutiliza, de tal forma que varios catedráticos prefieren no interactuar con el mismo y utilizan sistemas alternos.

El proyecto actual pretende solucionar muchos de estos problemas, tratando de ganar la confianza de los usuarios, que se han alejado debido a los errores.

OBJETIVOS

General

Solucionar los errores recurrentes que hacen que se sub-utilice la plataforma DTT, también realizar un cambio de servidores para optimizar el desempeño de la misma y crear un repositorio en GIT para llevar un control eficiente de los cambios.

Específicos

1. Solucionar errores del sistema de DTT e implementar una estrategia para el rastreo de estos. Además, rediseñar el ingreso de datos de varios módulos ya que, en la actualidad el proceso es inadecuado y provoca fallas.
2. Optimización de la plataforma en velocidad de despliegue de las páginas.
3. Levantar un repositorio en GIT para el versionamiento del proyecto DTT.
4. Cambiar la plataforma a los nuevos servidores.

INTRODUCCIÓN

En la actualidad, la tecnología juega un papel muy importante en el aprendizaje y en la administración de recursos porque mejora su uso. Por ello, la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería, implementó un sistema para llevar el control de recursos, estudiantes, catedráticos y tutores llamado DTT (Sistema de Transferencia Tecnológica).

El sistema DTT es una plataforma utilizada para diversos fines en la escuela de Ingeniería en Ciencias y sistemas. Entre ellos, el ingreso de notas, generación de constancias de prácticas, generación de Quiz, entre otros. Estas funciones están subutilizadas porque el diseño presenta varios problemas ya que surgen errores que en el entorno de pruebas no se vieron.

1. MARCO TEÓRICO

1.1. Herramientas por utilizar

A continuación, se definirán los conceptos más recurrentes que serán utilizados durante la elaboración del documento.

1.1.1. Lenguajes de programación

“En informática un lenguaje de programación es un programa destinado a la construcción de otros programas. Su función es organizar algoritmos y procesos que serán llevados a cabo por un sistema.”¹

1.1.1.1. Python

Python es un “lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible.”²

1.1.1.2. Javascript

JavaScript es un “lenguaje de programación que te permite realizar actividades complejas en una página web.”³

¹ María Estela Raffino. *Lenguaje de Programación*. <https://concepto.de/lenguaje-de-programacion>. Consulta: Octubre de 2019.

² Colaboradores de Wikipedia. *Python*. <https://es.wikipedia.org/wiki/Python>. Consulta: Septiembre de 2018.

³ Mozilla Foundation. *What is JavaScript?*. https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/Qu%C3%A9_es_JavaScript#%C2%BFQu%C3%A9_es_JavaScript. Consulta: Septiembre de 2018.

1.1.2. Manejador de código fuente

Es la herramienta que sirve para mantener el código generado de tal manera que, cuando se realice un cambio y este sea incorrecto, sea fácil de regresar a la versión previa.

1.1.3. Control de versiones

“El control de versiones es un sistema que registra los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, de modo que las versiones específicas se puedan recuperar más adelante.”⁴

1.1.3.1. GIT

“Es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente.”⁵

1.1.3.2. Repositorio

“Un repositorio es un espacio centralizado donde se almacena, organiza, mantiene y difunde información digital, habitualmente archivos informáticos, que pueden contener trabajos científicos, conjuntos de datos o software.”⁶

⁴ Software Freedom Conservancy. *Empezando - Acerca del control de versiones*. <https://git-scm.com/book/es/v1/Empezando-Acerca-del-control-de-versiones>. Consulta: Septiembre de 2018.

⁵ Colaboradores de Wikipedia. *Git*. <https://es.wikipedia.org/wiki/Git>. Consulta: Septiembre de 2018.

⁶ Colaboradores de Wikipedia. *Repositorio*. <https://es.wikipedia.org/wiki/Repositorio>. Consulta: Septiembre de 2018.

1.1.3.3. Versión

“Una revisión, versión o edición es un informe de estado de uno o varios archivos, marcado con un nombre identificativo, de manera que sea fácil de reconocer.”⁷

1.1.4. Base de datos

“Una base de datos es un ‘almacén’ que guarda grandes cantidades de información organizada para encontrarla y utilizarla fácilmente.”⁸

1.1.4.1. Sistema de gestión de base de datos

Los Sistemas de Gestión de Base de Datos (en inglés *DataBase Management System*) son un tipo de software muy específico dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.

1.1.4.2. SQL

“SQL (*Structured Query Language*) es un lenguaje de programación estándar e interactivo para la obtención de información desde una base de datos y para actualizarla.”⁹

⁷ Colaboradores de Wikipedia. *Repositorio*. <https://es.wikipedia.org/wiki/Repositorio>. Consulta: Septiembre de 2018.

⁸ Damián Pérez Valdés. ¿Qué son las bases de datos?. <http://www.maestrosdelweb.com/que-son-las-bases-de-datos/>. Consulta: Septiembre de 2018.

⁹ searchdatacenter. *SQL o lenguaje de consultas estructuradas*. <https://searchdatacenter.techtarget.com/es/definicion/SQL-o-lenguaje-de-consultas-estructuradas>. Consulta: Septiembre de 2018.

1.1.4.3. MYSQL

“MySQL es un sistema de gestión de base de datos relacional (RDBMS) de código abierto, basado en lenguaje de consulta estructurado (SQL).”¹⁰

1.1.5. Sistema operativo

“Un sistema operativo puede ser definido como un conjunto de programas especialmente hechos para la ejecución de varias tareas, en las que sirve de intermediario entre el usuario y la computadora. Este conjunto de programas que manejan el hardware de una computadora u otro dispositivo electrónico provee de rutinas básicas para controlar los dispositivos del equipo y administrar, escalar y realizar interacción de tareas.”¹¹

El sistema operativo sobre el cual corre la plataforma es Ubuntu, que es un sistema operativo basado en Linux. En concreto la distribución DEBIAN, además de ser de licencia libre, por lo que su uso no genera ningún costo.

1.2. Metodologías de desarrollo

Son un conjunto de prácticas para llevar el control del desarrollo de un proyecto. Dependiendo de cuál se siga, estas pueden ser más tolerantes a cambios de requerimientos.

¹⁰ searchdatacenter. MySQL. <https://searchdatacenter.techtarget.com/es/definicion/MySQL>. Consulta: Septiembre de 2018.

¹¹ tecnologia-informatica.com. *El sistema operativo*. <https://tecnologia-informatica.com/el-sistema-operativo/>. Consulta: Septiembre de 2018.

1.2.1. Desarrollo ágil

Es una metodología de desarrollo, que pretende ser flexible, para tolerar cambios, cuando el desarrollo del software está en marcha.

1.2.1.1. Metodología de desarrollo XP (*Extreme Programming*)

Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software. Promueve el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores y propiciando un buen clima de trabajo¹².

1.3. Arquitectura del sistema

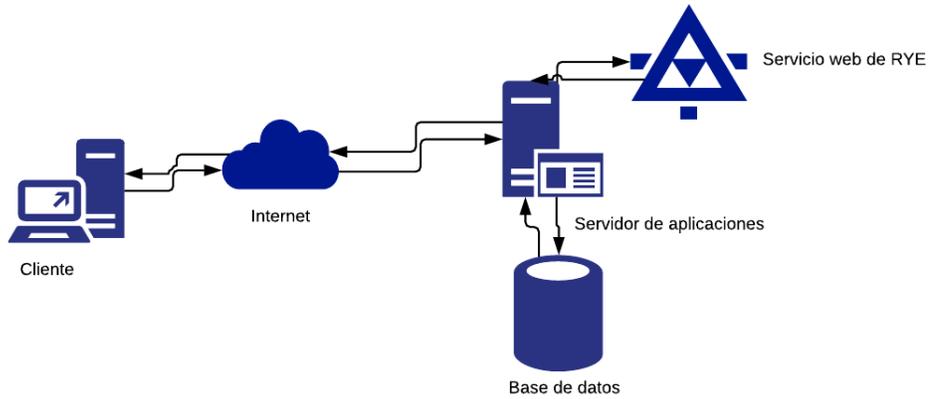
La arquitectura indica cómo se acoplan los componentes de software. Por ello, al modificar un sistema, es necesario consultarla o tratar de deducirla, porque provee un mapa de los componentes que interactúan en el sistema.

1.3.1. Descripción de la arquitectura

La arquitectura del sitio está basada en una arquitectura web, por lo que se compone de un servidor central, en el cual se tiene presente la aplicación y la base de datos, además de un servicio extra proporcionado por el Departamento de Registro y Estadística.

¹² Jose M. Bautista Q. *PROGRAMACION EXTREMA XP*. http://ingenieriadesoftware.mex.tl/52753_xp---extreme-programing.html. Consulta: Septiembre de 2018.

Figura 1. **Arquitectura global**



Fuente: elaboración propia, empleando LucidCharts.

1.3.2. **Elementos de la arquitectura**

Internamente, el sistema contiene varios componentes, el siguiente diagrama, especifica cómo estos interactúan entre sí.

Figura 2. **Componentes internos**



Fuente: elaboración propia, empleando LucidCharts.

2. FASE DE INVESTIGACIÓN

2.1. Antecedentes de la empresa

A continuación, se presenta una historia breve de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala.

2.1.1. Reseña histórica

La Universidad de San Carlos, fue fundada en 1676. En aquella época se dedicaba a formar teólogos, abogados y, más tarde, médicos.

En 1879 se estableció la escuela de Ingeniería que, a su vez, por acuerdo gubernativo de 1882 se elevó a Facultad y enfrentó problemas de inestabilidad en los años subsecuentes.

En 1930 se crea la carrera de Ingeniería civil con lo cual inicia la historia moderna de la facultad. En 1965, comienza el funcionamiento del primer centro de cálculo electrónico dotado de computadoras y equipo periférico, lo que constituyó un evento importante en el ámbito nacional y regional.

En 1970 se crea la carrera de Ingeniería en ciencias y sistemas. Desde entonces la Escuela de Ingeniería en Ciencias y Sistemas ha formado profesionales competentes que satisfagan las necesidades locales e internacionales.

La plataforma DTT (Sistema de Transferencia Tecnológica) surgió en el año 2014 para simplificar muchas tareas que, hasta ese momento, se llevaban en archivos físicos. Así se creó la primera versión de la plataforma que crecería con los años.

2.1.2. Misión

“En su carácter de única universidad estatal le corresponde con exclusividad dirigir, organizar y desarrollar la educación superior del estado y la educación estatal, así como la difusión de la cultura en todas sus manifestaciones. Promoverá por todos los medios a su alcance la investigación en todas las esferas del saber humano y cooperará al estudio y solución de los problemas nacionales.”¹³

2.1.3. Visión

La Universidad de San Carlos de Guatemala es la institución de educación superior estatal, autónoma, con cultura democrática, con enfoque multi e intercultural, vinculada y comprometida con el desarrollo científico, social, humanista y ambiental, con una gestión actualizada, dinámica, efectiva y con recursos óptimamente utilizados, para alcanzar sus fines y objetivos, formadora de profesionales con principios éticos y excelencia académica.

2.1.4. Servicios que realiza

El trabajo que se realizará se divide en dos servicios:

¹³ Facultad de Ingeniería, USAC. *Antecedentes.* <https://portal.ingenieria.usac.edu.gt/index.php/aspirante/antecedentes>. Consulta: Septiembre de 2018.

- Ayuda y soporte: consiste en resolver errores de la plataforma, además de apoyar con el cambio de servidor, resolviendo dudas y atendiendo peticiones de cambios.
- Gestión de control de cambios en el ambiente de producción: comprende el proceso de edición, así como la puesta en producción de archivos que se necesitan modificar, para resolver algún problema de funcionalidad del software, utilizando para ello una herramienta de versionamiento moderna y actualizada.

2.2. Descripción de las necesidades

Al analizar los servicios que necesitan mejoras e implementar una estrategia de rastreo de errores, se solucionan los problemas de forma y de fondo que presente la aplicación.

El sistema maneja diversos niveles de usuarios, con ello, se segmenta la información que cada usuario puede ver. De esta manera la información se maneja con seguridad. Además, permite o niega la ejecución de diversas actividades dentro del sitio.

El proyecto DTT provee una plataforma donde el catedrático, tutor y estudiantes mantienen una comunicación más continua. Por lo tanto, el sistema podrá ser utilizado únicamente por usuarios previamente registrados.

2.2.1. Ayuda y soporte

Comprende todo lo relacionado con el soporte al sistema, además de la ayuda en resolución de errores que se presenten en el mismo.

2.2.1.1. Módulo de reclutamiento de DTT

- Refactoring en el proceso de transmisión de datos, en el ingreso de solicitudes de práctica como en la calificación de puestos.
- Agregar la opción de descarga de archivo Excel de los interesados del proceso de reclutamiento, según el evento.
- Agregar el CUI en los formularios que muestra al rol Admin.
- Agregar el CUI a los Informes del rol Admin.
- Corregir los errores que se den, luego de implementar las correcciones y nuevas opciones.

2.2.1.2. Módulo de Quiz

- Agregar opción de modificación y eliminación de pruebas.
- Agregar reporte de notas por alumnos, en los roles, alumno, tutor, catedrático y administrador.

2.2.1.3. Módulo de Android

- Agregar la opción de ingreso de notas por grupo.
- Publicar aplicación en el sitio de DTT.

2.2.1.4. Módulo de visualización e ingreso de notas

- Optimización de consultas a base de datos, en la tabla de visualización de datos

2.2.1.5. Módulo 360

- Agregar número de alumnos que evaluaron en las pruebas de evaluación de catedráticos y tutores.
- Agregar funcionalidad de envío de correo al administrador, que se dispare cuando se activa una prueba, además de corregir, porque no envía correo a los roles que participan en dicha prueba.

2.2.1.6. Módulo de envío de correos

- Corregir los errores en visualización de mensajes en todos los roles.
- Corregir error en la redacción de un correo, que propicia la pérdida de su contenido cuando se adjuntan varios archivos al mensaje.
- Agregar al rol administrador, la opción de enviar copia a destinatarios ingresados de forma manual.

2.2.1.7. Módulo de menú, recursos y horarios

- Corrección de errores en el módulo de Recursos y Horarios, que hace que no se puedan visualizar periodos anteriores.
- Corregir los informes del rol administrador, debido a que presenta datos de períodos variables mezclados con períodos semestrales en los informes de “Entregables” y “Reporte por estado entregados”.
- Optimización de la vista de menú, para que pueda visualizarse correctamente en todas las pantallas.

2.2.1.8. Módulo de solvencia de prácticas

- Agregar funcionalidad, para que genere solvencia de prácticas individual y masivo.
- Agregar la funcionalidad necesaria, para que se puedan generar solvencia de prácticas, aun si el periodo no ha terminado.
- Agregar al Informe de Practicantes que, al Generar Excel, lleve el CUI de cada estudiante en la exportación a Excel.
- Corregir el error que impide, que se muestre, el jefe de proyecto verdadero, debido a que cuando se edita el jefe de proyecto, se pierde el histórico.

2.2.1.9. Módulo Imprimir Reporte (botón Imprimir)

- Reorganización del código *HTML* del reporte, para que el formato sea visualizado e impreso de manera correcta.

2.2.1.10. Módulo de *login*

- Agregar la funcionalidad que permita la verificación de si ya tiene correo ingresado en DTT, si es así, que obtenga el correo de la base de datos de DTT si no, que lo haga a través del servicio web de registro y estadística.

2.2.1.11. Módulo DSI

- Soporte del módulo por problemas que se puedan dar durante el segundo semestre del año 2018.

2.2.1.12. Levantar los ambientes en el nuevo servidor virtualizado

- Realizar todas las tareas necesarias, las cuales abarcan, creación de base de datos, levantado de servidor de aplicaciones y creación de aplicación, para que el proyecto DTT, pueda funcionar sin problemas en el nuevo servidor virtualizado.

2.2.2. Gestión de control de cambios en el ambiente de producción

- Creación y configuración de repositorio de DTT utilizando una plataforma dinámica, que permita a los usuarios una fácil adopción.
- Creación de un manual de introducción a la nueva herramienta de versionado.
- Especificaciones para cambios y versiones.
 - Roles Read, Write, Read/Write, Administrator.
 - Usuarios Asignados a roles.
 - Ramas por proyecto EPS asignado.
 - Acceso a usuarios solamente a ramas.
 - Debe indicarse archivos y funcionalidades versionadas.
 - Documentación de los cambios realizados.
 - Plan de recuperación en caso falle las pruebas.

2.3. Priorización de las necesidades

Para garantizar el éxito del proyecto, se dará prioridad a los arreglos de problemas más recurrentes, a las migraciones o servicios que se necesiten para poder iniciar el trabajo de manera eficiente.

A continuación, se presenta un orden sugerido en base a lo anterior.

Tabla I. **Prioridad de desarrollo**

Tarea	Prioridad	Comentario
Gestión de control de cambios en ambiente de producción	1	
Levantar ambiente en nuevo servidor virtualizado	2	No necesariamente debe llevarse en este orden, pero sería lo deseado.
Módulo imprimir reporte	3	Se ha indicado que causa muchos problemas, por lo que se recomienda ser el primer bug en resolver.
Módulo de reclutamiento de DTT	4	
Módulo de Quiz	4	
Módulo de Android	4	
Módulo de visualización e ingreso de notas	4	
Módulo 360	4	
Módulo de envío de correos	4	
Módulo de menú, recursos y horarios	4	
Módulo de solvencia de prácticas	4	

Continuación tabla I.

Módulo de login	4	
Módulo DSI	4	

Fuente: elaboración propia.

3. FASE TÉCNICO PROFESIONAL

3.1. Descripción del proyecto

La plataforma DTT es un proyecto estructurado por varios módulos. Ha crecido desde su primera implementación, en 2014. Por ello, muchas personas han participado en su desarrollo. Actualmente, se han generado muchos problemas debido a un desarrollo inadecuado o a una falta de experiencia en el desarrollo. Por esta razón es imperativo resolverlos para garantizar la buena usabilidad de este y recuperar la confianza que los usuarios han perdido debido a errores del sistema.

3.2. Investigación preliminar para la solución del proyecto

El proyecto se encuentra elaborado en el lenguaje de programación Python en su versión 2.7, utilizando un FRAMEWORK de desarrollo llamado *we2py*, el cual provee de facilidades y enmarca el desarrollo en un modelo de diseño MVC.

La plataforma, se encuentra alojada en un servidor web, proveído por el servicio DIGITALOCEAN, una empresa de Cloud Computing dedicada a crear contenedores y a darles mantenimiento.

Se está utilizando también un gestor de base de datos MARIADB. Esta es una base de datos flexible y robusta capaz de soportar grandes cantidades de información.

Además de lo anterior, se utiliza un servidor de caché, cuya función es agilizar las consultas frecuentes a base de datos. El servidor seleccionado para esta función es REDIS, que provee una interfaz con el software FRAMEWORK de desarrollo, esto permite tener el control de todo lo que se está almacenando en el caché.

Se tiene también un servidor de aplicaciones llamado NGINX. Este es un servidor ágil, muy popular, actualmente, por su versatilidad y rapidez. Usa el módulo *UWGSCI* para levantar aplicaciones de PYTHON en un ambiente web.

Además, en el pasado, se tenía un control de versiones con la herramienta SVN (subversión). Mantenía los cambios de código separados del ambiente de producción, hasta que eran probados adecuadamente. De esta forma aseguraba un buen funcionamiento del sitio. Sin embargo, debido a un problema en el soporte, el servidor de versionamiento fue borrado, por lo cual, el control de versiones, no se ha estado utilizando desde hace mucho tiempo.

También, se necesita cambiar el servidor para la plataforma porque está implementada en un servidor cuyo tiempo de soporte ha finalizado.

Lo anterior evidencia problemas, algunos más complicados que otros. Inicialmente, se controlarán las versiones y se resolverán los problemas de fondo con el uso de la aplicación, para recuperar la confianza de los usuarios.

3.3. Presentación de la solución al proyecto

Para solucionar los problemas descritos en la sección anterior, primero se debe crear un plan de acción porque varios frentes están involucrados. El proyecto cuenta con tres actividades:

- Implementación del control de versiones.
- Ayuda y soporte de errores de la aplicación.
- Traspaso de plataforma a nuevo servidor.

A continuación, se describe la solución para cada problema.

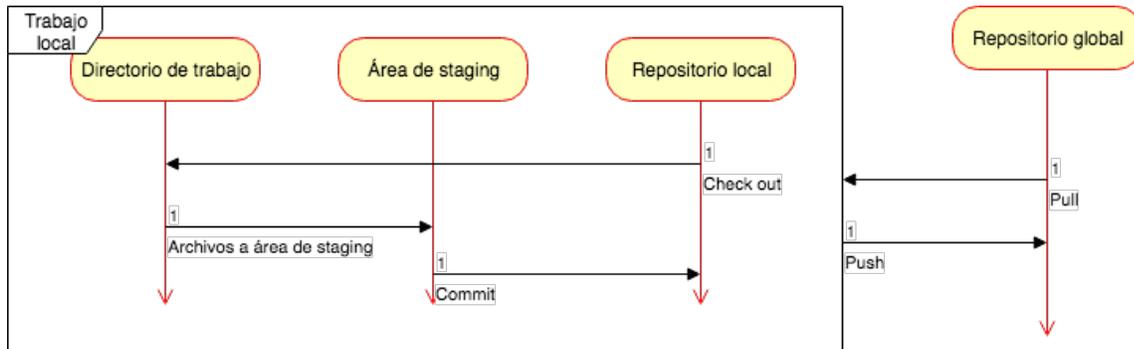
3.3.1. Implementación de control de versiones

Con el software de control de versiones los desarrolladores llevan una gestión de los cambios que se realizan en el código. Anteriormente, se tenía un control de versiones basado en SVN (Subversión). Este es un software de control de versiones lanzado en el año 2000. Desde entonces, han surgido otras opciones para llevar el control de los cambios.

Para llevar el control de la gestión de cambios, se ha seleccionado el software GIT, el cual presenta varias ventajas sobre SVN, la principal es que GIT permite el control de versiones de manera *offline* e incorporar estos cambios con el repositorio global, cuando se tenga acceso al repositorio.

Para lo anterior, GIT crea un repositorio local donde se pueden manejar Ramas diferentes, dependiendo de la persona que está trabajando.

Figura 3. Forma de trabajo de GIT



Fuente: elaboración propia, empleando LucidCharts.

El servidor utilizado para llevar el control de versiones estará alojado en DIGITALOCEAN.

3.3.2. Ayuda y soporte de errores de la aplicación

Existe una lista grande de errores de la aplicación, la mayoría de ellos surgieron, por un mal manejo del código o por inexperiencia de los desarrolladores.

Para solucionar los errores se realizarán reuniones frecuentes con el encargado del proyecto. De esas reuniones se crearán historias de usuario para llevar un control adecuado de la solución. Inicialmente, el encargado del proyecto, ha provisto una lista de errores que deben resolverse. Las reuniones servirán para poder identificar en qué parte de la aplicación se están generando y cómo se pueden replicar para poder determinar una causa y proponer una solución.

Se ha investigado y se ha llegado a la conclusión de que la aplicación puede sufrir dos clases de errores:

- Errores de configuración: son los errores que se producen al inicializar un componente de una manera inadecuada.
- Errores de concepto: son la mayoría de los errores que tiene la aplicación y consisten en el mal uso de los recursos de software por desconocimiento de la herramienta o *FRAMEWORK* utilizado.

Para la presentación y el desarrollo de la solución se utilizará una metodología de desarrollo ágil llamada “Extreme Programming” XP, en la cual no es necesario crear una gran documentación para iniciar el trabajo. Sin embargo, se documentarán los problemas por medio de historias de usuario, las cuales, son recursos importantes para saber cómo es que se necesitan hacer las cosas.

Adicionalmente a todo lo anterior, además de los artefactos de software utilizados se hará uso de otras herramientas para aumentar y facilitar la comunicación con el encargado del proyecto, las cuales son:

- SLACK: es una herramienta de comunicación que incorpora módulos que agilizan la comunicación de una o más personas.
- GoogleDocs: herramienta de edición de documentos que puede ser utilizada por un grupo.
- GoogleDuo: es una herramienta de comunicación para realizar llamadas y compartir pantallas, ideal para enseñar un nuevo proceso o una nueva funcionalidad, que deba ser aprendida.

3.3.3. Traspaso de plataforma a nuevo servidor

Esta tarea se refiere a una migración debido a que el sistema operativo, utilizado para la primera versión de la plataforma DTT, ya tiene 5 años, es prioritario migrar a un sistema operativo que tenga más tiempo de soporte.

Esta tarea tiene varias subtareas, las cuales se detallan a continuación.

- Instalación de software: se refiere a la instalación del software de base de datos, caché y *FRAMEWORKS* requeridos, para que la plataforma funcione de manera adecuada.
- Configuración de software: se refiere a las configuraciones necesarias, para que los componentes de software interactúen correctamente.

Los componentes de software que se instalarán en el servidor son los siguientes:

- Maria DB: base de datos.
- NGINX: Servidor de aplicaciones.
- REDIS: Servidor de cache, el cual servirá para poder acceder a información recurrente de manera rápida.
- UWSGI: Complemento de NGINX, que sirve, para poder servir aplicaciones desarrolladas en PYTHON.
- PYTHON: lenguaje de programación, se utilizará la versión 2.7.

3.4. Costos del proyecto

Para la elaboración del proyecto, se utilizará tanto recursos humanos como materiales, a continuación, se detalla una explicación de lo anterior.

3.4.1. Recursos humanos

Los recursos humanos para la realización del proyecto lo cual incluye las fases de análisis, diseño e implementación son los siguientes:

- Ingeniero Miguel Marín, Coordinador del proyecto DTT, y contacto con la institución, además de ser el asesor por parte de la institución.
- Ingeniero Luis Alberto Arias, asesor de EPS.
- Desarrollador, encargado del desarrollo e implementación del proyecto.

3.4.2. Recursos materiales

Debido a que el desarrollador trabajará a distancia, teniendo reuniones programadas con los contactos de la institución para presentar avances, este utilizará sus propios recursos, los cuales son los siguientes.

- Computadora personal con los requerimientos mínimos siguientes:
 - Sistema operativo: Windows XP
 - Procesador 2.4 GHZ
 - Memoria RAM 512 MB
 - Impresora
- Software para el desarrollo
 - ATOM (IDE de programación libre)
 - NGINX

- Explorador de internet (Google Chrome, Firefox, Internet Explorer, etc.)
- Google Docs.
- Trello
- MYSQL
- LucidCharts para el desarrollo de presentaciones y manuales

3.4.3. Costos

Los costos por la elaboración del proyecto son detallados a continuación:

Tabla II. **Costos del proyecto**

Recursos	Cantidad	Costo Unitario	Subtotal
<i>Desarrollador</i>	476 horas	Q. 150,00	Q. 71 400,00
<i>Asesor de EPS</i>	50 horas	Q. 300,00	Q. 15 000,00
<i>Supervisor de EPS</i>	50 horas	Q. 300,00	Q. 15 000,00
<i>Equipo de cómputo de desarrollo</i>	1	Q. 6 000,00	Q. 6 000,00
<i>Servidor</i>	1	Q. 30 000,00	Q. 30 000,00
<i>Servicios varios</i>		Q. 10 000,00	Q. 10 000,00
<i>Capacitaciones</i>		Q. 1 500,00	Q. 1 500,00
<i>Gastos varios</i>		Q. 8 000,00	Q. 8 000,00
<i>Total</i>			Q. 156 900,00

Fuente: elaboración propia.

3.5. Beneficios del proyecto

Se estima que los beneficios obtenidos, al finalizar el proyecto, serán:

- Mayor aceptación de la plataforma DTT por parte de los usuarios.

- Mejora en el rendimiento al revisar cómo se están realizando las operaciones en este momento.
- Mejora de estabilidad al corregir errores que están impactando al sistema.

4. FASE DE ENSEÑANZA Y APRENDIZAJE

La enseñanza y aprendizaje del usuario es muy importante porque en ella se demuestra si el esfuerzo realizado para la elaboración del sistema cumple con las expectativas de los usuarios.

El aprendizaje del usuario es determinante para el éxito del proyecto porque de este depende que los cambios en el sistema sean adoptados y utilizados adecuadamente.

4.1. Perfil del usuario

El usuario que se capacitará necesita algunos conocimientos de informática para utilizar a cabalidad las herramientas de gestión de repositorios, así como para la resolución de errores que puedan darse en el futuro. Entre las habilidades que se sugiere que el usuario tenga están las siguientes:

- Conocimiento de alguna herramienta de versionamiento de software.
- Experiencia básica en administración de servidores LINUX.
- Conocimiento básico del lenguaje PYTHON.
- Conocimiento de administración de sistemas de bases de datos, en especial MARIADB o MYSQL.
- Conocimiento básico en herramientas de caché como memcache o REDIS.

4.2. Transferencia de conocimientos

Es necesario determinar el grado de aprendizaje del usuario. Se deben valorar algunas variables como la aplicación de los conocimientos obtenidos, el cumplimiento de las prácticas enseñadas y el manejo de las herramientas para la operativa diaria, así como la resolución de problemas.

4.2.1. Uso de las herramientas

El usuario debe aprender a gestionar los cambios de código y lidiar con los conflictos producidos, cuando dos o más personas editan el mismo archivo.

Para lo anterior el usuario debe dominar los comandos básicos y tener ejemplos prácticos, los cuales están descritos en el apéndice número 2 de este documento.

4.2.2. Casos prácticos de utilización

Para el presente proyecto, es necesario que el usuario aprenda nuevos conceptos, sobre el control de versiones GIT. Por eso, se ofrecerá una sesión de capacitación y asistencia, mientras el usuario se acostumbra a usar esta herramienta de versionamiento.

4.3. Metodología de aprendizaje

Para administrar el progreso en el que se incrementarán los conocimientos y habilidades es necesario realizar ciertas capacitaciones, evaluaciones y observaciones. Se dividirán en varias sesiones relacionadas con los objetivos de cada una de las herramientas.

4.3.1. Sesiones de aprendizaje

Se plantea llevar a cabo sesiones presenciales o virtuales para explicar los conceptos fundamentales sobre el uso de la herramienta, que no excedan las 2 horas por sesión.

Cada sesión cuenta con una sección introductoria, desarrollo y conclusión de esta.

4.4. Capacitación propuesta

Se llevarán a cabo tres sesiones para explicar los conceptos necesarios para el traspaso de conocimientos de la siguiente manera.

Tabla III. **Capacitaciones propuestas**

Tema	Duración (horas)
Uso y conceptos básicos de GIT	2
Uso de la plataforma BitBucket	2
Copias de seguridad y restauración de bases de datos REDIS	2

Fuente: elaboración propia.

Si el usuario necesita reforzar conocimientos en alguno de estos temas, podrá solicitar una nueva capacitación.

4.5. Material elaborado

Se proporcionará material para que el usuario aprenda a utilizar la herramienta GIT porque con ella se integrarán, en el futuro, los cambios que los usuarios necesiten.

Además se desarrollará una sesión de aprendizaje donde se abordará el uso de la plataforma BitBucket que servirá para administrar a los usuarios del repositorio.

Se proporciona, también, un manual de migración de la plataforma en el apéndice número 1, donde se describe paso a paso, el proceso que se debe llevar a cabo, cambiar de servidor en el futuro, con la limitante, que pueden cambiar algunas cosas, dependiendo de algún cambio en la arquitectura del servidor o de los cambios de futuros proyectos.

5. DESCRIPCIÓN PROCESOS DEL SISTEMA Y SEGURIDAD

En un sistema tan amplio y con tantas opciones como es DTT, generalmente, quedan opciones a las que no se les presta la atención debida. Por eso, en esta sección, se esclarecerán estas opciones importantes para el funcionamiento correcto de la aplicación. Adicionalmente, se trata de explicar los métodos de seguridad que se tienen tanto en el servidor de aplicaciones como en el servidor de control de versiones.

5.1. Descripción de los procesos automáticos del servidor

En esta sección, se abordarán los procesos automáticos que DTT necesita para funcionar correctamente, se explicará lo que hacen, no así cómo se instalan y se ponen en marcha.

DTT tiene 3 procesos importantes que corren automáticamente en el servidor, los cuales son:

5.1.1. Scheduler

Este proceso sirve para crear variables que se utilizan a lo largo del proyecto, generalmente, estas variables son almacenadas en base de datos.

Scheduler se encuentra funcionando continuamente dentro del sistema operativo debido a que está levantado como un servicio del sistema. Este utiliza PYTHON como lenguaje de programación, además del FRAMEWORK WEB2PY, que es el que se utiliza para el desarrollo de la aplicación DTT.

Scheduler se encarga de crear variables muy importantes, también para inicializar otras ya existentes.

Entre las funciones de Scheduler tenemos las siguientes:

- Inicializar el semestre cuando la fecha indique que debe cambiar
- Envío de correos programados alertando inicio de pruebas
- Envío de resultados de pruebas
- Envío de notificaciones al administrador

Las anteriores son algunas de las funcionalidades que lleva acabo Sheduler, pero este tiene otras muchas más.

5.1.2. Regeneración de certificado SSL

Actualmente, que un sitio tenga un dominio HTTPS da la seguridad al usuario de que es confiable. El certificado SSL es una firma de identidad que da constancia de que el sitio es de la institución de la cual dice ser.

Por tal motivo se ha vuelto imperativo que el sistema DTT cuente con un dominio HTTPS, por ello se implementó un método para renovarlo, en el cual no es necesaria la intervención humana.

Para ello, se utiliza el software CERTBOT de la entidad LetsEncrypt, la cual es una entidad capacitada para extender certificados SSL.

CERTBOT provee de una función de regeneración la cual no es automática, pero es útil porque, con un solo comando, regenera el certificado SSL.

Dicho certificado tiene una vigencia de 3 meses, por ello, se creó un script que, usando las capacidades del servidor, permite volverlo una tarea programada que se ejecuta a diario.

No existe ningún problema con que el certificado no haya expirado, porque CERTBOT verifica si la renovación de este es necesaria.

5.1.3. Copia de seguridad semanal

Debido a que todos los sistemas en algún momento fallan por causas directas o indirectas, se implementó una copia de seguridad de la base de datos, la cual funciona semanalmente.

Para lo anterior, se utilizó el software MariaBackup, el cual permite realizar copias de seguridad completas de la base de datos. Además, maneja cierto grado de encriptamiento, lo cual permite que, si esta copia cayera en manos equivocadas, este tenga problemas para poder descifrar su contenido.

Además, las copias de base de datos se almacenan en un servidor distinto al de la aplicación DTT. Para conectarse a este servidor se utiliza la aplicación SSHPASS, la cual es un cliente de BASH, que permite la conexión remota a otro servidor.

Finalmente, esta es una tarea programada y, como tal, su definición se encuentra en el archivo `"/etc/crontab"` del servidor de DTT.

5.2. Seguridad en el servidor de aplicaciones

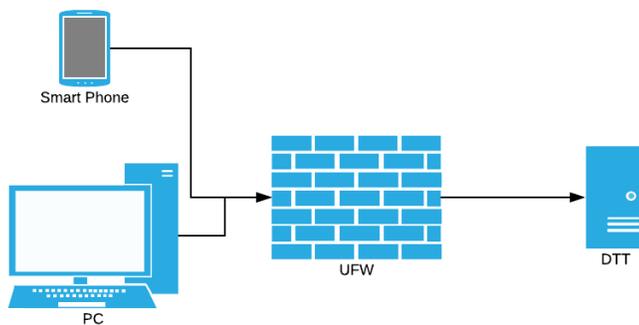
Actualmente, es imprescindible un esquema de seguridad por los problemas que pueda causar ignorar este tema.

A continuación, se describirá el esquema de seguridad tanto del servidor de aplicaciones como del repositorio.

5.2.1. Descripción de la seguridad del servidor de aplicaciones

Las conexiones al servidor son monitoreadas por UFW el cual es un firewall desarrollado por UBUNTU (distribución de Linux utilizada como servidor de aplicaciones de DTT) que además es de fácil uso.

Figura 4. Diagrama de seguridad de *firewall*



Fuente: elaboración propia, empleando LucidCharts.

En el firewall se restringieron los puertos 80 (HTTP) y 443 (HTTPS) para el tráfico de peticiones, todos los demás puertos están restringidos.

También se manejó un nivel de seguridad en el usuario de la aplicación, la cual está restringida a un usuario específico, creado puntualmente para poder ejecutar la aplicación. No existe ninguna carpeta que tenga todos los permisos, sino que se les fue cambiado el dueño por el usuario que utiliza la aplicación.

Para lo anterior, se modificó la configuración del servidor de aplicaciones porque tenía por defecto el usuario “www-data” el cual es un usuario utilizado en la mayoría de sitios para que ejecute las aplicaciones y podría suponer una amenaza de seguridad dejar el mismo.

La aplicación tiene las claves de los usuarios encriptadas con PBKDF2 (*Password-Based Key Derivation Function 2*), que es un método que utiliza un diccionario muy extenso para prevenir posibles ataques de fuerza bruta.

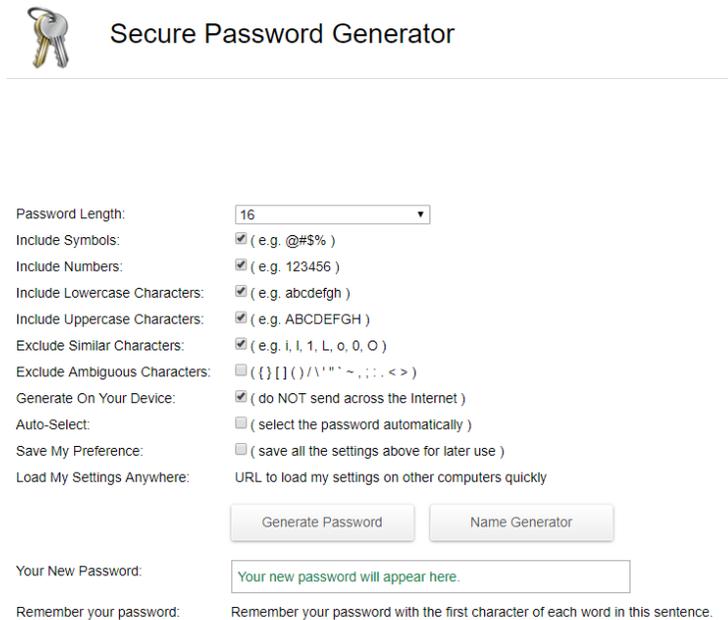
5.2.2. Descripción de la seguridad de base de datos

Con respecto a la seguridad de base de datos, todas las conexiones están restringidas a servidor local, por lo que los usuarios que intenten conectarse desde otro servidor no podrán hacerlo. Adicionalmente, el puerto está restringido por firewall, por lo que todas las peticiones a esta serán denegadas.

El password del usuario de base de datos fue generado aleatoriamente para prevenir algún patrón descifrable y con ello garantizar la seguridad de estos.

Para lo anterior, fue necesario el uso de la web “*passwordsgenerator.net*”, la cual permite generar *passwords* complejos.

Figura 5. Formulario de generación de *password*



The image shows a web form titled "Secure Password Generator" with a key icon. The form includes several settings: Password Length (16), Include Symbols (checked), Include Numbers (checked), Include Lowercase Characters (checked), Include Uppercase Characters (checked), Exclude Similar Characters (checked), Exclude Ambiguous Characters (unchecked), Generate On Your Device (checked), Auto-Select (unchecked), Save My Preference (unchecked), and Load My Settings Anywhere (URL field). There are two buttons: "Generate Password" and "Name Generator". Below the buttons, there is a text box for "Your New Password" with the placeholder "Your new password will appear here." and a note: "Remember your password: Remember your password with the first character of each word in this sentence."

Fuente: Elaboración propia, empleando <https://passwordsgenerator.net/>

De igual manera, los datos almacenados en cache necesitan estar protegidos para que nadie pueda obtener los datos almacenados por el motor de cache REDIS, por eso, también se generó un password en “*passwordsgenerator.net*”. Además, también se restringió la consulta al servidor local, por lo que personas desde otros equipos, no podrán consultar el cache.

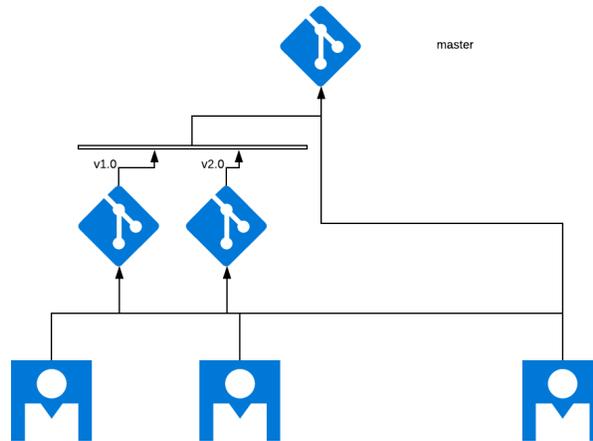
5.3. Seguridad en el repositorio de versionamiento

Debido a que el repositorio es utilizado por muchas personas es necesario tener cierto nivel de seguridad, para prevenir cualquier error ocasionado por la mala manipulación de este.

Actualmente, la rama del repositorio que se encuentra en el ambiente productivo es la “máster”, la cual contiene todo el código probado anteriormente y aprobado por el administrador de DTT.

Por eso, se debe restringir el acceso a la rama “máster” para evitar inconvenientes producidos por errores o por mala voluntad.

Figura 6. **Esquema de seguridad de GIT**



Fuente: elaboración propia, empleando LucidCharts.

Afortunadamente la plataforma utilizada para el repositorio nos permite gestionar permisos por usuarios y por ramas, por lo que entonces se restringió el permiso de escritura a la rama “master”.

Adicionalmente se excluyeron todos los archivos de configuración que son sensibles, es decir que almacenan claves o direcciones de servidores, por lo que cuando un nuevo usuario clona el repositorio necesitara solicitar los archivos de configuración genéricos al administrador.

CONCLUSIONES

1. El sistema presentó errores, la mayoría recurrentes, estos se corrigieron alguna vez en versiones anteriores, pero nuevamente se presentaron y se han perdido funcionalidades debido a los constantes cambios. Por eso, fue necesario recodificar varios controladores y realizar una documentación detallada de los procesos que funcionan en la plataforma, así como de los componentes que dichos procesos requieren para su buen funcionamiento.
2. Se encontró software desactualizado en la plataforma. Esto genera inestabilidad y lentitud en el despliegue de las páginas. Entonces, se actualizaron varias piezas de software, desde el sistema operativo hasta el software encargado de las capas de datos y aplicación. Con ello, se mejoró la respuesta de la plataforma en general.
3. El manejador de versiones anterior no recibió el mantenimiento adecuado. Debido a esto, se regresaron a prácticas de versionamiento no recomendadas para la naturaleza del proyecto; adicionalmente, no se contaba con un ambiente de desarrollo. Se implementó un repositorio GIT utilizando las mejores prácticas de versionamiento sobre la plataforma BitBucket; también se creó un ambiente de pruebas similar al ambiente de producción para lograr una implementación de cambios más transparente y que reduzca el impacto negativo para el usuario final debido a los constantes cambios.

4. Los servidores donde se encontraba la plataforma DTT iban a ser decomisados por el proveedor debido a que ya no contarían con el soporte adecuado. Debido a esto, se movió la plataforma completa a nuevos servidores que contarán con el soporte adecuado del lado del proveedor.

RECOMENDACIONES

Con base en el estudio y el trabajo realizado, se pueden realizar las siguientes recomendaciones, para que el funcionamiento de la aplicación mejore:

1. La plataforma debe mantenerse en el esquema Cloud Computing, debido a la ventaja que un tercero es el encargado de ver el correcto funcionamiento de los servidores. No importa si hay un corte de energía el sistema siempre seguirá funcionando. En caso contrario, al alojar el sistema en un servidor propio, se necesita una infraestructura robusta para garantizar la disponibilidad y confiabilidad de la plataforma, esto incrementa los costos fijos del funcionamiento del sistema.
2. El ambiente de producción necesita dividirse en al menos dos servidores, uno que contenga la capa de datos y otro que aloje la capa de aplicación. Esto podría beneficiar enormemente el rendimiento de la plataforma, debido a que los recursos se utilizarían y distribuirían de mejor manera.
3. No se debe ignorar el proceso implementado para las pruebas y se debe utilizar adecuadamente el ambiente creado para este propósito, garantizando que los cambios realizados tengan un correcto funcionamiento antes de su despliegue en el ambiente de producción.
4. Se debe utilizar la herramienta REDIS solo para consultar catálogos que son frecuentemente utilizados (caché). Almacenar otra clase de

información, como formularios, información transaccional o datos persistentes a través del tiempo, generan múltiples problemas en la arquitectura actual de la plataforma.

5. Para el control de versiones, en la plataforma “BitBucket”, se sugiere que se cree una rama del repositorio por cada EPS para mantener un mejor control acerca de los cambios realizados por los desarrolladores. Es más fácil probar ramas individuales previas a la rama “máster”, esta última debe mantenerse solo para el ambiente de producción.

6. Para mejorar el mantenimiento se necesita incluir mejores prácticas de desarrollo y crear una nomenclatura estandarizada para los proyectos que se encuentran bajo la plataforma DTT. Por ejemplo, darle un nombre significativo a variables, procedimientos, archivos, módulos y ramas de cambios. Adicionalmente añadir un prefijo predefinido al nombre para saber que exactamente es lo que se está modificando.

BIBLIOGRAFÍA

1. Centro de Cálculo. Universidad de San Carlos, Facultad de Ingeniería. *Antecedentes*. [en línea]. <<https://portal.ingenieria.usac.edu.gt/index.php/aspirante/antecedentes>>. [Consulta: 4 de Julio de 2018].
2. Git-scm.com. *Git - Acerca del control de versiones*. [en línea]. <<https://git-scm.com/book/es/v1/Empezando-Acerca-del-control-de-versiones>>. [Consulta 20 de Agosto de 2018].
3. Es.wikipedia.org. *Python (lenguaje de programación)*. [en línea]. <<https://es.wikipedia.org/wiki/Python>>. [Consulta: 2 de octubre de 2018].
4. Documentación web de MDN. *¿Qué Es JavaScript?*. [en línea]. <https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/Qué_es_JavaScript>. [Consulta: 15 de agosto de 2018].
5. SearchDataCenter en español. *¿Qué es SQL o lenguaje de consultas estructuradas? - Definición en WhatIs.com*. [en línea]. <<http://searchdatacenter.techtarget.com/es/definicion/SQL-o-lenguaje-de-consultas-estructuradas>>. [Consulta: 30 de agosto de 2018].

6. Maestros del Web. *¿Qué son las bases de datos?*. [en línea].
<<http://www.maestrosdelweb.com/que-son-las-bases-de-datos/>>.
[Consulta: 5 de octubre de 2018].

7. Es.wikipedia.org. *Repositorio*. [en línea].
<<https://es.wikipedia.org/wiki/Repositorio>>. [Consulta: 23 de agosto de 2018].

8. Ingenieriaedesoftware.mex.tl. *XP - Extreme Programing Ingenieria de Software*. [en línea]. <http://ingenieriaedesoftware.mex.tl/52753_XP--Extreme-Programing.html>. [Consulta: 3 de agosto de 2018].

APÉNDICES

Apendice 1. Manual de migración

Instalación de NGINX , Ptyhon, REDIS y MariaDB

Actualizar los repositorios de Ubuntu

```
apt-get update
```

```
apt-get -y upgrade
```

Figura 1. Actualización de repositorios de ubuntu

```
carlos@carlos-VirtualBox:~$ sudo apt upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following NEW packages will be installed:
  bubblewrap libbrotli1 libwoff1 linux-headers-4.15.0-43 linux-headers-4.15.0-43-generic linux-image-4.15.0-43-generic
  linux-modules-4.15.0-43-generic linux-modules-extra-4.15.0-43-generic python3-netifaces
The following packages will be upgraded:
  cups cups-bsd cups-client cups-common cups-core-drivers cups-daemon cups-ipp-utils cups-ppdc cups-server-common firefox
  firefox-locale-en fonts-opensymbol fwupd ghostscript ghostscript-x gir1.2-gnomedesktop-3.0 gir1.2-javascriptcoregtk-4.0
  gir1.2-webkit2-4.0 gkbd-capplet gnome-desktop3-data gnome-software gnome-software-common gnome-software-plugin-snap libcups2
  libcupsctl libcupsimage2 libcupsmtlme1 libcupsppdc1 libfreerdp-client2-2 libfreerdp2-2 libfwupd2 libgnome-desktop-3-17
  libgnomekbd-common libgnomekbd8 libgs9 libgs9-common libjavascriptcoregtk-4.0-18 libperl5.26 libpoppler-glib8 libpoppler73
  libraw16 libssl1.0.0 libssl1.1 libwvpack1 libwebkit2gtk-4.0-37 libwinpr2-2 linux-generic linux-headers-generic
  linux-image-generic linux-libc-dev netplan.io nplan openssl perl perl-base perl-modules-5.26 poppler-utils
  python3-software-properties python3-update-manager software-properties-common software-properties-gtk ubuntu-software
  unattended-upgrades update-manager update-manager-core usbmuxd
66 upgraded, 9 newly installed, 0 to remove and 0 not upgraded.
Need to get 142 MB/158 MB of archives.
After this operation, 339 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Fuente: elaboración propia.

Instalar NGINX ,Python y REDIS

```
apt-get -y install nginx-full
apt-get -y install build-essential python-dev libxml2-dev python-pip unzip
apt-get -y install redis-server
```

Continuación apéndice 1.

Figura 2. Instalación de servidor de aplicaciones NGINX

```
carlos@carlos-VirtualBox:~$ sudo apt-get -y install nginx-full
```

Fuente: elaboración propia.

Figura 3. Instalación de librerías de python para desarrollo

```
carlos@carlos-VirtualBox:~$ sudo apt-get -y install build-essential python-dev libxml2-dev python-pip unzip
```

Fuente: elaboración propia.

Figura 4. Instalación de redis caché

```
carlos@carlos-VirtualBox:~$ sudo apt-get -y install redis-server
```

Fuente: elaboración propia.

Se restauran los datos del caché de redis, primero se ubica el lugar donde se debe poner el backup.

```
redis-cli
127.0.0.1:6379> auth password
127.0.0.1:6379> config get dir
```

Esto retorna una dirección, en este caso “/var/lib/redis”, lo que significa que el archivo de datos de redis está guardado en esta carpeta.

Se detiene el servicio de REDIS.

```
service redis-server stop
```

Continuación apéndice 1.

Se verifica si existe algún proceso de REDIS corriendo.

```
ps -A | grep redis
```

Si existe, se detiene

```
kill -9 <número de proceso>
```

Se nombra de nuevo el archivo de datos de REDIS dump.rdb a dump.rdb.old.

```
mv /var/lib/redis/dump.rdb /var/lib/redis/dump.rdb.old
```

Se copia el archivo de backup en el directorio.

```
cp -p dump.rdb /var/lib/redis/
```

Se cambian los permisos del archivo, junto con el propietario de este a redis.

```
chown redis:redis /var/lib/redis/dump.rdb  
chmod 660 /var/lib/redis/dump.rdb
```

Se arranca el servicio de redis y verifica si hubo algún error, por el que no pueda funcionar.

```
service redis-server start  
service redis-server status
```

Continuación apéndice 1.

Figura 5. Arranque de redis caché

```
root@svrddt:/home/root# service redis-server start
root@svrddt:/home/root# service redis-server status
● redis-server.service - Advanced key-value store
   Loaded: loaded (/lib/systemd/system/redis-server.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2019-01-13 22:21:20 UTC; 9s ago
     Docs: http://redis.io/documentation,
           man:redis-server(8)
   Process: 22889 ExecStop=/bin/kill -s TERM $MAINPID (code=exited, status=0/SUCCESS)
   Process: 23012 ExecStart=/usr/bin/redis-server /etc/redis/redis.conf (code=exited, status=0/SUCCESS)
  Main PID: 23027 (redis-server)
    Tasks: 4 (limit: 1152)
   CGroup: /system.slice/redis-server.service
           └─23027 /usr/bin/redis-server 127.0.0.1:6379

Jan 13 22:21:19 svrddt systemd[1]: Starting Advanced key-value store...
Jan 13 22:21:20 svrddt systemd[1]: redis-server.service: Can't open PID file /var/run/redis/redis-server.pid (yet?) after start: No such file or directory
Jan 13 22:21:20 svrddt systemd[1]: Started Advanced key-value store.
```

Fuente: elaboración propia.

Si todo funciona bien se realiza una prueba.

```
redis-cli -n 4
auth password
keys *
```

Figura 6. Lista de llaves almacenadas en redis caché

```
[root@svrddt:/home/root# redis-cli -n 4
127.0.0.1:6379[4]> keys *
 1) "uid:1077:curso:26:quiz:16"
 2) "uid:1614:curso:69:quiz:18"
 3) "uid:1523:curso:14:quiz:14"
 4) "uid:968:curso:70:quiz:6"
 5) "uid:1550:curso:27:quiz:19"
 6) "idquiz"
 7) "uid:1077:curso:26:quiz:17"
 8) "uid:723:curso:62:quiz:2"
 9) "uid:674:curso:34:quiz:8"
10) "uid:376:curso:64:quiz:7"
11) "uid:707:curso:84:quiz:10"
12) "uid:272:curso:109:quiz:4"
13) "uid:674:curso:34:quiz:5"
14) "uid:723:curso:62:quiz:3"
15) "uid:272:curso:62:quiz:1"
16) "uid:1533:curso:24:quiz:13"
17) "uid:272:curso:109:quiz:20"
18) "uid:376:curso:64:quiz:11"
19) "uid:699:curso:85:quiz:9"
20) "uid:240:curso:18:quiz:12"
21) "uid:1077:curso:26:quiz:15"
127.0.0.1:6379[4]>
```

Fuente: elaboración propia.

Continuación apéndice 1.

Instalación de MariaDB

```
apt-get install gnupg2
apt-get install software-properties-common
#Se agrega la llave del repositorio
apt-key adv --recv-keys --keyserver hkp://keyserver.ubuntu.com:80
0xF1656F24C74CD1D8
#se agrega el nuevo repositorio
add-apt-repository 'deb [arch=amd64]
http://mirror.zol.co.zw/mariadb/repo/10.3/ubuntu bionic main'
```

Figura 7. Importación de llave de nuevo repositorio

```
carlos@carlos-VirtualBox:~$ sudo apt-key adv --recv-keys --keyserver hkp://keyserver.ubuntu.com:80 0xF1656F24C74CD1D8
Executing: /tmp/apt-key-gpghome.QTUZH2LFew/gpg.1.sh --recv-keys --keyserver hkp://keyserver.ubuntu.com:80 0xF1656F24C74CD1D8
gpg: key F1656F24C74CD1D8: 6 signatures not checked due to missing keys
gpg: key F1656F24C74CD1D8: public key "MariaDB Signing Key <signing-key@mariadb.org>" imported
gpg: Total number processed: 1
gpg:      imported: 1
```

Fuente: elaboración propia.

Figura 8. Adición de nuevo repositorio de MariaDB

```
carlos@carlos-VirtualBox:~$ sudo add-apt-repository 'deb [arch=amd64] http://mirror.zol.co.zw/mariadb/repo/10.3/ubuntu bionic main'
Hit:1 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:2 http://gt.archive.ubuntu.com/ubuntu bionic InRelease
Hit:3 http://gt.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:4 http://gt.archive.ubuntu.com/ubuntu bionic-backports InRelease
Get:5 http://mariadb.mirror.liquidtelecom.com/repo/10.3/ubuntu bionic InRelease [3,887 B]
Get:6 http://mariadb.mirror.liquidtelecom.com/repo/10.3/ubuntu bionic/main amd64 Packages [8,246 B]
Fetched 12.1 kB in 3s (3,879 B/s)
Reading package lists... Done
```

Fuente: elaboración propia.

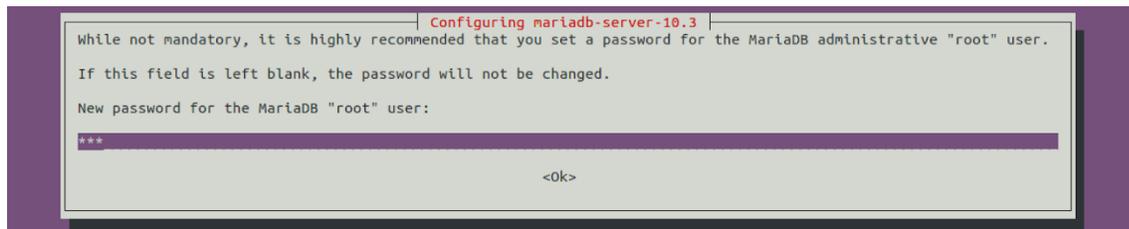
Se actualizan los repositorios y se instala mariadb

```
sudo apt update
sudo apt -y install mariadb-server mariadb-client
```

Se escribe el password de root para la base de datos y se confirma

Continuación apéndice 1.

Figura 9. **Configuración de password maestro de base de datos**

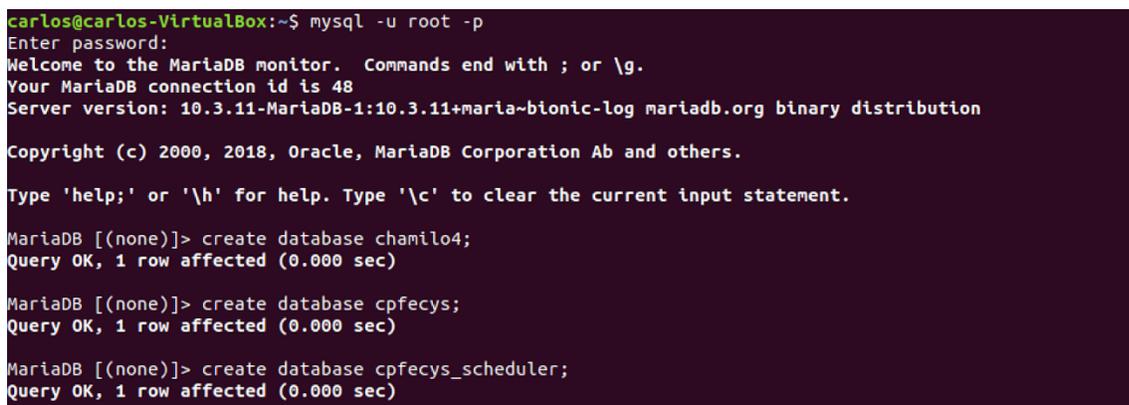


Fuente: elaboración propia.

Se ingresa al administrador de base de datos y se crean las bases de datos necesarias.

```
mysql -u root -p
#creo las bases de datos necesarias
create database chamilo4;
create database cpfecys;
create database cpfecys_scheduler;
```

Figura 10. **Creación de bases de datos del proyecto**



Fuente: elaboración propia.

Continuación apéndice 1.

Se levantan los backup de base de datos:

```
mysql -u root -p chamilo4 < chamilo4.sql
mysql -u root -p cpfecys < cpfecys.sql
mysql -u root -p cpfecys_scheduler < cpfecys_scheduler.sql
```

Figura 11. Importación de bases de datos de backup

```
carlos@carlos-VirtualBox:~/Documents/dtt_src$ mysql -u root -p chamilo4 < chamilo4.sql
Enter password:
carlos@carlos-VirtualBox:~/Documents/dtt_src$ mysql -u root -p cpfecys < cpfecys.sql
Enter password:
carlos@carlos-VirtualBox:~/Documents/dtt_src$ mysql -u root -p cpfecys_scheduler < cpfecys_scheduler.sql
Enter password:
carlos@carlos-VirtualBox:~/Documents/dtt_src$
```

Fuente: elaboración propia.

Se descarga el controlador de REDIS para Python.

```
pip install redis
```

Figura 12. Instalación de plugin de redis cache, para Python

```
carlos@carlos-VirtualBox:~/Documents/dtt_src$ pip install redis
Collecting redis
  Downloading https://files.pythonhosted.org/packages/f5/00/5253aff5e747faf10d8ceb35fb5569b848cde2fdc13685d42fcf63118bbc/redis-3.0.1-py2.py3-none-any.whl (61kB)
    100% |#####| 71kB 619kB/s
Installing collected packages: redis
Successfully installed redis-3.0.1
carlos@carlos-VirtualBox:~/Documents/dtt_src$
```

Fuente: elaboración propia.

Instalación de plugin mysql2redis

Existen funciones específicas para agilizar la lectura a base de datos, para ello, se mezcló mariadb con redis para un mayor performance.

Continuación apéndice 1.

El proceso de instalación es el siguiente:

Se descargan las dependencias y se descomprimen:

```
wget https://github.com/redis/hiredis/archive/v0.13.3.tar.gz
tar -xzvf v0.13.3.tar.gz
cd hiredis-0.13.3/
```

Se instala la dependencia hiredis.

```
make
make install
```

Figura 13. Instalación de plugin hiredis

```
root@svrddt:/home/root/hiredis-0.13.3# make
cc -std=c99 -pedantic -c -O3 -fPIC -Wall -W -Wstrict-prototypes -Wwrite-strings -g -ggdb net.c
cc -std=c99 -pedantic -c -O3 -fPIC -Wall -W -Wstrict-prototypes -Wwrite-strings -g -ggdb hiredis.c
cc -std=c99 -pedantic -c -O3 -fPIC -Wall -W -Wstrict-prototypes -Wwrite-strings -g -ggdb sds.c
cc -std=c99 -pedantic -c -O3 -fPIC -Wall -W -Wstrict-prototypes -Wwrite-strings -g -ggdb async.c
cc -std=c99 -pedantic -c -O3 -fPIC -Wall -W -Wstrict-prototypes -Wwrite-strings -g -ggdb read.c
cc -shared -Wl,-soname,libhiredis.so.0.13 -o libhiredis.so net.o hiredis.o sds.o async.o read.o
ar rcs libhiredis.a net.o hiredis.o sds.o async.o read.o
cc -std=c99 -pedantic -c -O3 -fPIC -Wall -W -Wstrict-prototypes -Wwrite-strings -g -ggdb test.c
cc -O3 -fPIC -Wall -W -Wstrict-prototypes -Wwrite-strings -g -ggdb -o hiredis-test test.o libhiredis.a
Generating hiredis.pc for pkgconfig...
root@svrddt:/home/root/hiredis-0.13.3# █
```

Fuente: elaboración propia.

Se verifica que se encuentren las librerías instaladas.

```
ldconfig -p |grep hiredis
```

Si este comando no muestra ninguna línea, entonces hay que vincular la librería. Para localizar la librería se utiliza el comando locate:

Continuación apéndice 1.

```
updatedb  
locate libhiredis.so
```

Figura 14. Localizando ruta de instalación de hiredis

```
[root@svrdtt:/home/root/hiredis-0.13.3# updatedb  
[root@svrdtt:/home/root/hiredis-0.13.3# locate libhiredis.so  
/home/root/hiredis-0.13.3/libhiredis.so  
/usr/local/lib/libhiredis.so  
/usr/local/lib/libhiredis.so.0.13
```

Fuente: elaboración propia.

Ahora se vincula la librería

```
echo "/usr/local/lib" > /etc/ld.so.conf.d/local.conf  
ldconfig  
ldconfig -p |grep hiredis
```

Ahora se descarga mysql2redis

```
wget https://github.com/jackeylu/mysql2redis/archive/master.zip  
unzip master.zip  
cd mysql2redis-master/
```

Se instalan las dependencias.

```
apt-get install libmysqlclient-dev
```

Se edita el archivo "src/lib_mysqludf_redis.c", en él se busca el string "%ld" y se reemplaza por "%lld", después se corre el instalador y se siguen los pasos.

```
./install.sh
```

Continuación apéndice 1.

Con esto está ya todo listo.

Instalacion de framework WEB2PY

La plataforma DTT, utiliza una versión específica del framework Web2Py, la cual es la 2.9.5

```
sudo wget https://github.com/web2py/web2py/archive/R-2.9.5.zip
```

Figura 15. Descargando el framework Web2Py

```
carlos@carlos-VirtualBox: /var/www$ sudo wget https://github.com/web2py/web2py/archive/R-2.9.5.zip
--2018-12-26 17:57:44-- https://github.com/web2py/web2py/archive/R-2.9.5.zip
Resolving github.com (github.com)... 192.30.253.112, 192.30.253.113
Connecting to github.com (github.com)|192.30.253.112|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/web2py/web2py/zip/R-2.9.5 [following]
--2018-12-26 17:57:44-- https://codeload.github.com/web2py/web2py/zip/R-2.9.5
Resolving codeload.github.com (codeload.github.com)... 192.30.253.121, 192.30.253.120
Connecting to codeload.github.com (codeload.github.com)|192.30.253.121|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/zip]
Saving to: 'R-2.9.5.zip'

R-2.9.5.zip          [          ] 6.38M  1.44MB/s   in 4.9s
2018-12-26 17:57:50 (1.31 MB/s) - 'R-2.9.5.zip' saved [6685742]
carlos@carlos-VirtualBox: /var/www$
```

Fuente: elaboración propia.

El archivo fue descargado en la carpeta “/var/www”, se desempaqueta.

```
sudo unzip R-2.9.5.zip
```

Ahora, se nombre de nuevo la carpeta “web2py-R-2.9.5” a “web2py”

```
sudo mv web2py-R-2.9.5 web2py
```

Continuación apéndice 1.

Instalación de GIT

Antes de tener la aplicación, se clona desde el repositorio, para lo cual se instala GIT

```
sudo apt install git
```

Figura 16. Instalación de GIT

```
carlos@carlos-VirtualBox:~/Documents/dtt_src$ sudo apt install git
[sudo] password for carlos:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  git-man liberror-perl
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-el git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  git git-man liberror-perl
0 upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 4,733 kB of archives.
After this operation, 33.9 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Fuente: elaboración propia.

Clonación de la aplicación DTT

Navegar hasta el directorio “/var/www/web2py/applications/” y clonar el repositorio.

```
sudo git clone https://cvaldez@bitbucket.org/pryddtt/cpfecys.git
```

(La dirección del repositorio debe ser proveída por el administrador de DTT)

Continuación apéndice 1.

Figura 17. Clonación de repositorio DTT

```
carlos@carlos-VirtualBox:/var/www/web2py/applications$ sudo git clone https://cvaldez@bitbucket.org/pryddt/cpfecys.git
Cloning into 'cpfecys'...
Password for 'https://cvaldez@bitbucket.org':
remote: Counting objects: 1383, done.
remote: Compressing objects: 100% (964/964), done.
remote: Total 1383 (delta 370), reused 1309 (delta 328)
Receiving objects: 100% (1383/1383), 6.65 MiB | 1.70 MiB/s, done.
Resolving deltas: 100% (370/370), done.
```

Fuente: relaboración propia.

Se crea el archivo de configuración, el cual está ubicado en “/var/www/web2py/applications/cpfecys/modules/config.py”, por motivos de seguridad, no será especificado su contenido.

Ahora, se genera el certificado digital, se genera en la raíz de la carpeta del framework

```
sudo openssl req -x509 -new -newkey rsa:4096 -days 3652 -nodes -keyout web2py.key -out web2py.crt
```

Figura 18. Creación de certificado SSL provisional

```
carlos@carlos-VirtualBox:/var/www/web2py$ sudo openssl req -x509 -new -newkey rsa:4096 -days 3652 -nodes -keyout web2py.key -out web2py.crt
Generating a 4096 bit RSA private key
.....++
.....++
writing new private key to 'web2py.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:New York
Locality Name (eg, city) []:New York City
Organization Name (eg, company) [Internet Widgits Pty Ltd]:USAC
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:localhost
Email Address []:ecysusac@gmail.com
```

Fuente: elaboración propia.

Ahora, se va a la raíz del directorio web2py y se prueba el sitio, ejecutando la siguiente instrucción.

Continuación apéndice 1.

```
sudo python web2py.py -k web2py.key -c web2py.crt -i 0.0.0.0 -p 443
```

Figura 19. Iniciando la aplicación DTT

```
carlos@carlos-VirtualBox:/var/www/web2py$ sudo python web2py.py -k web2py.key -c web2py.crt -i 0.0.0.0 -p 443
ERROR:root:New installation error: unable to create welcome.w2p file
web2py Web Framework
Created by Massimo Di Pierro, Copyright 2007-2018
Version 2.9.5-stable+timestamp.2014.03.15.21.24.06
Database drivers available: SQLite(sqlite3), MySQL(pymysql), PostgreSQL(pg8000), IMAP(imaplib)
WARNING:web2py:GUI not available because Tk library is not installed
choose a password:

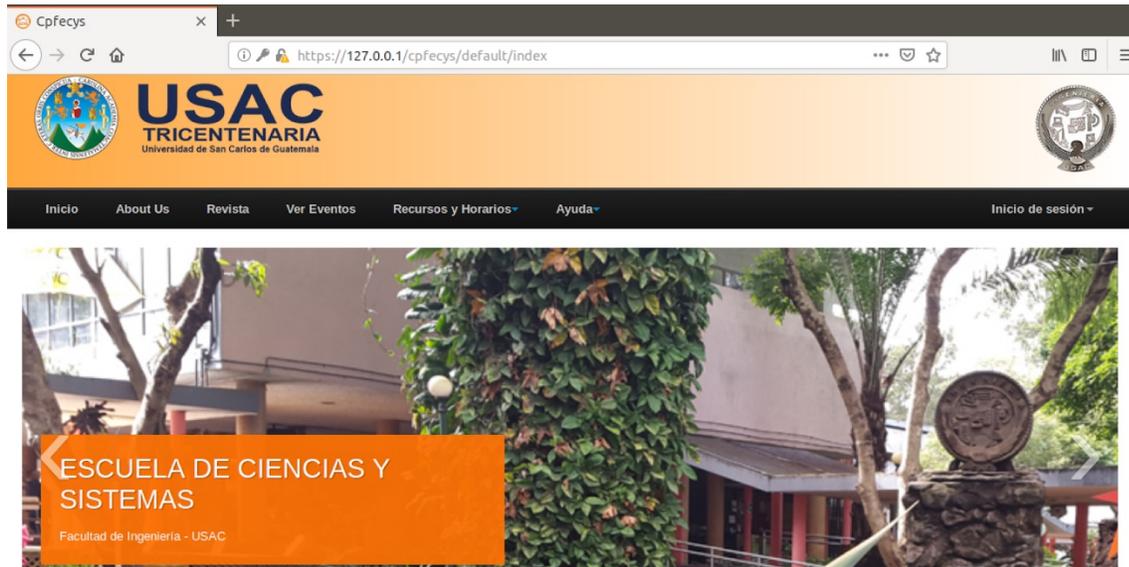
please visit:
  https://127.0.0.1:443/
use "kill -SIGTERM 4122" to shutdown the web2py server
```

Fuente: elaboración propia.

Se ingresa al navegador web y se escribe la dirección “https://127.0.0.1:443”, o la dirección IP del servidor que se ha asignado. En el navegador se verá una advertencia debido a que el certificado no es reconocido, entonces se entra en la configuración avanzada y aceptar el certificado, con lo que se resolverá el problema y se verá la siguiente pantalla:

Continuación apéndice 1.

Figura 20. Verificando que la aplicación inicie



Fuente: elaboración propia.

Configuración de NGINX y UWSGI

Se instalan los utilitarios para poder de python para UWSGI.

```
sudo pip install setuptools --no-use-wheel --upgrade
```

Figura 21. Instalación de módulo UWSGI Python

```
carlos@carlos-VirtualBox:~/var/www/web2py$ pip install setuptools --no-use-wheel --upgrade
DEPRECATION: --no-use-wheel is deprecated and will be removed in the future. Please use --no-binary :all: instead.
Collecting setuptools
  Downloading https://files.pythonhosted.org/packages/37/1b/b25507861991beeade31473868463dad0e58b1978c209de27384ae541b0b/setuptools-40.6.3.zip (839kB)
    100% |#####| 839kB 348kB/s
Skipping bdist_wheel for setuptools, due to binaries being disabled for it.
Installing collected packages: setuptools
  Running setup.py install for setuptools ... done
Successfully installed setuptools-40.6.3
```

Fuente: elaboración propia.

Instaló el módulo UWSGI con el siguiente comando.

Continuación apéndice 1.

```
/usr/bin/pip install --upgrade uwsgi
```

Se crea el directorio de web2py para NGINX.

```
sudo mkdir /etc/nginx/conf.d/web2py
```

Se crean los siguientes archivos.

```
sudo nano /etc/nginx/conf.d/web2py/gzip_static.conf
```

```
gzip_static on;  
gzip_http_version 1.1;  
gzip_proxied expired no-cache no-store private auth;  
gzip_disable "MSIE [1-6]\.";  
gzip_vary on;
```

```
sudo nano /etc/nginx/conf.d/web2py/gzip.conf
```

```
gzip on;  
gzip_disable "msie6";  
gzip_vary on;  
gzip_proxied any;  
gzip_comp_level 6;  
gzip_buffers 16 8k;  
gzip_http_version 1.1;  
gzip_types text/plain text/css application/json application/x-javascript text/xml application/xml  
application/xml+rss text/javascript;
```

Creó el archivo de configuración de web2py para NGINX.

```
sudo nano /etc/nginx/sites-available/web2py
```

Continuación apéndice 1.

```
server {
    listen 80 default_server;
    listen [::]:80 default_server ipv6only=on;
    root /usr/share/nginx/html;
    index index.html index.htm;

    server_name localhost;
    location ~* /(\w+)/static/ {
        root /var/www/web2py/applications/;
    }
    location / {
        include uwsgi_params;
        uwsgi_pass unix:/var/www/web2py/web2py.sock; uwsgi_read_timeout 300;
    }
}
# HTTPS server
server {
    listen 443;
    server_name localhost;
    root html;
    location ~* /(\w+)/static/ {
        root /var/www/web2py/applications/;
    }
    index index.html index.htm;
    ssl on;
    ssl_certificate /var/www/web2py/web2py.crt;
    ssl_certificate_key /var/www/web2py/web2py.key;
    ssl_session_timeout 5m;
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers HIGH:!aNULL:!MD5;
    ssl_prefer_server_ciphers on;
    location / {
        include uwsgi_params;
        uwsgi_pass unix:/var/www/web2py/web2py.sock;
        uwsgi_read_timeout 300;
    }
}
```

Se crea el enlace simbólico de la configuración para habilitarla.

```
sudo ln -s /etc/nginx/sites-available/web2py /etc/nginx/sites-enabled/web2py
```

Se elimina la configuración por defecto.

```
sudo rm /etc/nginx/sites-enabled/default
```

Continuación apéndice 1.

Se crean los directorios necesarios para que el módulo uwsgi funcione correctamente.

```
mkdir /etc/uwsgi
mkdir /var/log/uwsgi
mkdir /etc/systemd
mkdir /etc/systemd/system
```

Se crean los archivos de configuración de uwsgi.

```
sudo nano /etc/systemd/system/emperor.uwsgi.service
```

```
[Unit]
Description = uWSGI Emperor
After = syslog.target
[Service]
ExecStart = /usr/local/bin/uwsgi --ini /etc/uwsgi/web2py.ini
RuntimeDirectory = uwsgi
Restart = always
KillSignal = SIGQUIT
Type = notify
StandardError = syslog
NotifyAccess = all
[Install]
WantedBy = multi-user.target
```

```
sudo nano /etc/uwsgi/web2py.ini
```

Continuación apéndice 1.

```
[uwsgi]
chdir = /var/www/web2py
module = wsgihandler:application

master = true
process = 5

socket = /var/www/web2py/web2py.sock
chmod-socket = 666
logto = /var/log/uwsgi/%n.log

sudo nano /etc/init/uwsgi-emperor.conf
```

```
# Emperor uWSGI script
description "uWSGI Emperor"
start on runlevel [2345]
stop on runlevel [!2345]
###
#remove the comments in the next section to enable static file compression for the welcome
app
#in that case, turn on gzip_static on; on /etc/nginx/nginx.conf
###
#pre-start script
# python /var/www/web2py/web2py.py -S cpecys -R scripts/zip_static_files.py
# chown -R www-data:www-data /home/www-data/web2py/*
#end script

setuid www-data

exec uwsgi --master --die-on-term --emperor /etc/uwsgi --logto /var/log/uwsgi/uwsgi.log
```

Se comprueba la configuración de NGINX

```
sudo nginx -t
```

Continuación apéndice 1.

Figura 22. **Comprobación de configuración de NGINX**

```
carlos@carlos-VirtualBox:/var/www/web2py$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

Fuente: elaboración propia.

Se otorgan los permisos necesarios a la carpeta del Framework.

```
sudo chown -R www-data web2py/
sudo chgrp -R www-data web2py/
```

Se arrancan los servicios de NGINX y uwsgi.

```
sudo service nginx restart
sudo systemctl start emperor.uwsgi.service
sudo systemctl enable emperor.uwsgi.service
```

Fuente: elaboración propia.

Apéndice 2. Manual de uso de GIT

¿Qué es GIT?

GIT es un software de control de versiones, pensado en la eficiencia y confiabilidad al manejar muchos archivos.

¿Qué operaciones utiliza?

Como cualquier software de versionamiento, GIT utiliza varias operaciones, desde la clonación de un repositorio, hasta el envío de los cambios a dicho repositorio.

Entre las operaciones principales de GIT están las siguientes:

- **estatus.** Muestra una lista de los archivos modificados y su estado, es decir, si se encuentran en la fase de staging, o si no han sido incluidos.
- **add.** Esta operación sirve para agregar los archivos modificados a la fase de staging, generalmente es necesario hacer add de algún archivo que se necesite hacer commit.
- **commit** Esta operación permite pasar los cambios de la fase de staging al repositorio local.
- **push.** Este comando permite pasar los cambios alojados en el repositorio local, hacia el repositorio global.
- **pull.** Esta operación permite sincronizar los cambios en el repositorio global, con los del repositorio local.
- **merge.** Cuando se sincronizan los repositorios, si un archivo ha sido editado por varias personas, y no se pueden resolver los conflictos de

Continuación apéndice 2.

forma automática, se utiliza merge, que permitirá sincronizar los archivos con conflicto de forma manual.

¿Cómo clonar un repositorio?

Generalmente, el primer paso en GIT es clonar un repositorio, el cual contendrá el código a ser editado, a continuación, un ejemplo.

Se cuenta con el repositorio example, y para clonarlo, el administrador nos proporcionó la siguiente url:

```
https://cvaldez@bitbucket.org/prueba_cavs/example.git
```

Para clonar el repositorio, primero se ubica en la dirección donde se desea clonar, por ejemplo, yo tengo ubicados mis repositorios en mi carpeta de documentos, entonces, por consola nos ubicamos en la carpeta documentos. Después de ello ejecutamos el comando clone.

```
git clone https://cvaldez@bitbucket.org/prueba_cavs/example.git
```

Figura 1. Clonación de repositorio

```
MacBook-de-usuario:Documents usuario$ git clone https://cvaldez@bitbucket.org/prueba_cavs/example.git
Cloning into 'example'...
remote: Counting objects: 28, done.
remote: Compressing objects: 100% (13/13), done.
remote: Total 28 (delta 7), reused 0 (delta 0)
Unpacking objects: 100% (28/28), done.
```

Fuente: elaboración propia.

Continuación apéndice 2.

Se puede ver la creación de una carpeta con el contenido del repositorio en la carpeta *Documents*.

¿Cómo cambiar de rama?

Generalmente, hacer cambios en la rama “*master*” está restringido por motivos de seguridad, la rama “*master*”, es el que generalmente se encuentra en los ambientes de producción, y no es conveniente tener código sin probar en el, es por ello, se otorga una rama, en la cual se realizarán los cambios deseados sin que estos alteren el código de producción.

Digamos, que el administrador nos creó una rama llamado “*feature-a*”, lo único que tenemos que hacer, es ubicarnos dentro de la carpeta de nuestro repositorio y ejecutar el comando checkout.

```
git checkout feature-a
```

Figura 2. Cambio de rama

```
MacBook-de-usuario:example usuario$ git checkout feature-a  
Branch 'feature-a' set up to track remote branch 'feature-a' from 'origin'.  
Switched to a new branch 'feature-a'
```

Fuente: elaboración propia.

¿Cómo hago que los cambios que hice, lleguen al repositorio global?

Debido a que GIT, utiliza un repositorio local, esto porque garantiza que siempre podamos guardar los cambios, aunque nos encontremos desconectados del repositorio global, es necesario hacer un paso extra a la hora de realizar cambios en el repositorio global. Para todo esto, se debe utilizar

Continuación apéndice 2.

el comando `add`, que nos permite agregar nuestros cambios al área de staging, después utilizar el comando `commit`, que hace que los cambios que tengamos en staging vayan al repositorio local, y después el comando `push`, que hace que los cambios que tengamos en el repositorio local, vayan al repositorio global.

Primero vemos cuales son los archivos afectados con el comando estatus.

```
git status
```

Figura 3. Lista de cambios en el repositorio

```
MacBook-de-usuario:example usuario$ git status
On branch feature-a
Your branch is up to date with 'origin/feature-a'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

       modified:   prueba.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)

       prueba2.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

Fuente: elaboración propia.

Ahora agregamos los cambios al área de staging utilizando el comando `add` y enviamos los cambios al repositorio local con el comando `commit`.

```
git add .
git commit -m "primer Cambio"
```

Continuación apéndice 2.

Figura 4. Uso de commit y add

```
[MacBook-de-usuario:example usuario$ git add .
[MacBook-de-usuario:example usuario$ git commit -m "primer cambio"
[feature-a 57a807e] primer cambio
 2 files changed, 3 insertions(+), 1 deletion(-)
 create mode 100644 prueba2.txt
```

Fuente: elaboración propia.

Por último, se envían los cambios al repositorio global, para ello utilizamos el comando *push*.

```
git push
```

Figura 5. Enviando cambios al repositorio global

```
[MacBook-de-usuario:example usuario$ git push
Counting objects: 4, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 366 bytes | 366.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0)
remote:
remote: Create pull request for feature-a:
remote:   https://bitbucket.org/prueba_cavs/example/pull-requests/new?source=feature-a&t=1
remote:
To https://bitbucket.org/prueba_cavs/example.git
   8e2da77..57a807e feature-a -> feature-a
```

Fuente: elaboración propia.

¿Cómo unir dos ramas?

Una operación frecuente es pasar los datos de la rama a la rama b, lo cual a veces puede tener algo de dificultad si existen conflictos entre los archivos.

Continuación apéndice 2.

Para lo anterior, se utiliza el comando merge, que nos permite de manera automática resolver conflictos, pero si este no funciona tendremos que resolverlos de forma manual, a continuación, un ejemplo de cada caso:

Merge de ramas sin conflictos

El administrador ha creado una nueva rama de la cual se encargará otra persona, quien está haciendo sus cambios, resulta que ambos tienen cambios y necesitan probarlos en el ambiente de pruebas, y te tienes que encargar, de que se vean, tanto los cambios de tu rama, como los de tu compañero, para ello utilizamos el comando merge.

Se cambia a la rama deseada que tenga todos los cambios, en este caso sería "feature-a".

```
git checkout feature-a
```

Ahora, se obtienen los cambios de la otra rama con el comando *merge*.

```
git merge feature-b
```

Aparecerá una pantalla, donde se deben presionar las teclas: wq, después, se mandarán los cambios al repositorio global con push.

```
git push
```

Continuación apéndice 2.

Figura 6. Merge sin conflictos

```
[MacBook-de-usuario:example usuario$ git merge feature-b
Merge made by the 'recursive' strategy.
 prueba3.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 prueba3.txt
[MacBook-de-usuario:example usuario$ git push
Counting objects: 2, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 344 bytes | 344.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0)
remote:
remote: Create pull request for feature-a:
remote:  https://bitbucket.org/prueba_cavs/example/pull-requests/new?source=feature-a&t=1
remote:
To https://bitbucket.org/prueba_cavs/example.git
 84b5b50..0ea59af feature-a -> feature-a
```

Fuente: elaboración propia.

Merge de ramas con conflictos

Ahora se verá cómo resolver conflictos. Resulta que ambos tuvieron que editar el mismo archivo y modificaron algunas líneas en común, nuevamente se utiliza el comando merge para obtener los cambios.

```
git merge feature-b
```

Figura 7. Merge con conflictos

```
[MacBook-de-usuario:example usuario$ git merge feature-b
Auto-merging prueba3.txt
CONFLICT (content): Merge conflict in prueba3.txt
Automatic merge failed; fix conflicts and then commit the result.
```

Fuente: elaboración propia.

Primero, se abre el archivo afectado, aparecerá algo como lo que sigue:

Continuación apéndice 2.

Figura 8. **Archivo con conflictos**

```
|<<<<<< HEAD  
Esto es la tercera prueba conflicto conflicto  
=====  
Esto es la tercera prueba de conflicto  
>>>>>> feature-b
```

Fuente: elaboración propia.

Aquí se debe seleccionar el cambio que se desea que pase y luego se agregan los cambios con add y, por último, hacer el commit.

```
git add .  
git commit
```

Aparecerá una pantalla como la que sigue:

Continuación apéndice 2.

Figura 9. Pantalla de conflictos

```
Merge branch 'feature-b' into feature-a
# Conflicts:
#   prueba3.txt
#
# It looks like you may be committing a merge.
# If this is not correct, please remove the file
#   .git/MERGE_HEAD
# and try again.
#
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch feature-a
# Your branch is up to date with 'origin/feature-a'.
#
# All conflicts fixed but you are still merging.
#
# Changes to be committed:
#   modified:   prueba3.txt
#
~
~
~
~
```

Fuente: elaboración propia.

Donde se debe presionar “:+wq+enter”, y luego hacer push, para que los cambios queden en el repositorio global.

```
git push
```

Continuación apéndice 2.

Figura 10. Push al repositorio global

```
[feature-a f37c753] Merge branch 'feature-b' into feature-a
MacBook-de-usuario:example usuario$ git push
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 340 bytes | 340.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote:
remote: Create pull request for feature-a:
[remote:  https://bitbucket.org/prueba_cavs/example/pull-requests/new?source=feature-a&t=1
remote:
To https://bitbucket.org/prueba_cavs/example.git
   ac3319a..f37c753  feature-a -> feature-a
```

Fuente: elaboración propia.

Apéndice 3. **Uso de la plataforma BITBUCKET**

Descripción de la administración del control de versiones

El control de versiones se ha convertido en una herramienta de suma importancia para el desarrollo de aplicaciones, debido a que permite tener organizados los cambios que cada persona realiza sobre un proyecto de software.

Lo anterior se vuelve más importante si en el proyecto participan muchas personas, tanto si estas lo están modificando al mismo tiempo, como si deben modificar las funcionalidades de una aplicación estable.

El control de versiones es importante, porque ayuda a prevenir errores que podrían afectar a los usuarios de una plataforma.

Actualmente, con DTT se utiliza la herramienta GIT, que es un control de versiones sumamente versátil y ágil, el cual permite manejar un volumen muy amplio de cambios. Entre sus ventajas está que se puede utilizar fuera de línea, o que se pueden crear etiquetas, las cuales representan hitos del proyecto, en caso de que algún error ocurra.

La plataforma utilizada para administrar el repositorio GIT de la aplicación DTT es BitBucket. Esta se seleccionó porque permite tener repositorios privados, y tener un control de las personas que pueden ver el código de la aplicación. También posibilita manejar un volumen alto de transacciones, que es muy importante cuando varias personas están participando activamente en un proyecto de desarrollo de software.

Continuación apéndice 3.

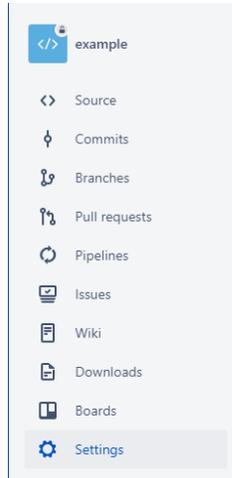
A continuación, se tratará de explicar, las funciones más importantes de BitBucker para que se aprenda a administrar el repositorio de DTT de manera eficaz, ágil y correcta.

Agregar a un usuario al repositorio

Para agregar a un usuario a un repositorio, este debe contar con una cuenta en BitBucket, además se deben seguir estos pasos:

Seleccionar la opción “Settings” del menú del repositorio del proyecto.

Figura 1. Menú de repositorio

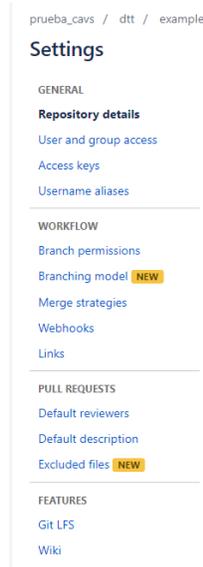


Fuente: elaboración propia.

En el menú de “Settings” se debe seleccionar la opción “User and group Access”

Continuación apéndice 3.

Figura 2. Menú Settings



Fuente: elaboración propia.

Aparecerá un formulario como el que sigue:

Figura 3. Formulario de ingreso de usuarios



Fuente: elaboración propia.

En este se debe ingresar el email o el nombre de usuario registrado en Bitbucket de la persona que deseas que pueda contribuir en el repositorio.

Continuación apéndice 3.

Las opciones "READ", "WRITE" y "ADMIN", representa lo siguiente:

- READ. Solo se puede ver el código del repositorio, pero no se puede cambiar el contenido.
- WRITE. Se puede ver el código y modificarlo, pero no se puede crear ramas ni hacer merges.
- ADMIN. Se tienen los permisos, además es posible hacer ramas y merges entre ramas.

Solamente se puede seleccionar un tipo de rol para un usuario.

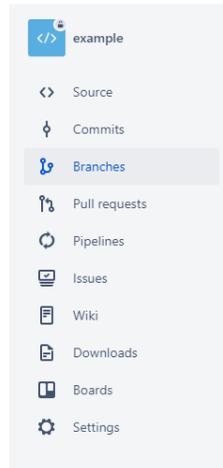
Crear una rama

Cuando se haya creado un usuario, lo siguiente es otorgarle una rama en el cual pueda trabajar.

Para ello, se ingresa en el repositorio al que el usuario tiene acceso y se hace click en el menú "Ramas".

Continuación apéndice 3.

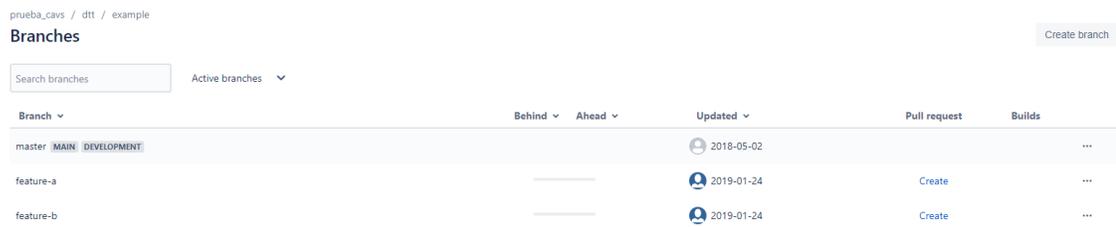
Figura 4. **Opción Branches**



Fuente: elaboración propia.

En el formulario que aparece se busca la opción “Create branch”.

Figura 5. **Opción Create branch**

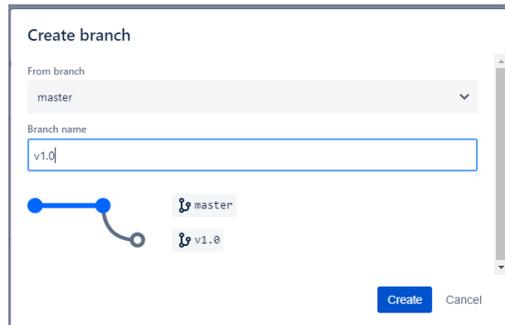


Fuente: elaboración propia.

Después de hacer clic en la opción “Create branch”, aparecerá un formulario como el que sigue:

Continuación apéndice 3.

Figura 6. **Formulario de creación de branch**



Fuente: elaboración propia.

Aquí se selecciona de qué rama se creará esta nueva rama, generalmente, se hace a partir del “*máster*”, ya que este debe ser el más actualizado y depurado de todos. Además, se debe seleccionar un nombre, de preferencia un valor numérico incremental o algo que indique qué se trabajará en esta rama, por último, se presiona el botón “*Create*”.

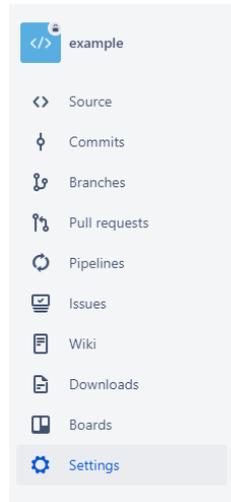
Administrar ramas

Debido a que muchas personas hacen cambios en las ramas, es conveniente definir a alguna persona (un administrador) que sea el encargado de hacer los cambios en la rama “*máster*” para prevenir que a este lleguen errores o pruebas de los desarrolladores, ya que por lo general esta rama se utiliza para publicar en el ambiente productivo.

Para restringir esta rama, se selecciona la opción “*Settings*” del repositorio.

Continuación apéndice 3.

Figura 7. Opción **Settings**

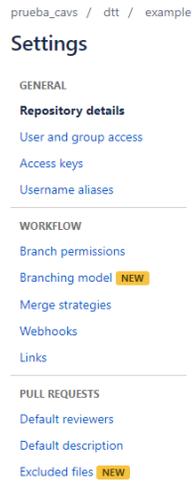


Fuente: elaboración propia.

Después se debe seleccionar la opción “Branch permissions”.

Continuación apéndice 3.

Figura 8. Opción **Branch permissions**



Fuente: elaboración propia.

Aparecerá un mensaje como el siguiente donde se selecciona la opción “Add branch permission”.

Continuación apéndice 3.

Figura 9. **Add branch permission**



Get a better handle on this repository's branches

Control who can merge into specific branches directly or only with pull requests.

Check that pull requests meet certain conditions before a merge — for example, the desired number of approvals or passing builds.

[Add a branch permission](#) [Learn more](#)

Fuente: elaboración propia.

Después de seleccionar la opción “Add branch permission”, aparecerá un formulario como el siguiente:

Continuación apéndice 3.

Figura 10. **Formulario Add branch permission**

Add a branch permission

Select branch By name or pattern

By type

Write access
Users and groups who can push or merge any changes to this branch directly

Merge via pull request
Users and groups who can merge to this branch via pull request

[> Merge checks](#)

Fuente: elaboración propia.

Seleccionamos la opción “By name or pattern”, y en el cuadro de texto escribimos el nombre de la rama, en este caso, se debe escribir “máster”.

En el campo “Write Access”, se selecciona al o a los usuarios que tendrán acceso a modificar esta rama.

Continuación apéndice 3.

Figura 11. Formulario lleno

Add a branch permission

Select branch By name or pattern

master

By type

Write access

Users and groups who can push or merge any changes to this branch directly

Allow rewriting branch history

Allow deleting this branch

Merge via pull request

Users and groups who can merge to this branch via pull request

> Merge checks

Save Cancel

Fuente: elaboración propia.

Por último, seleccionamos presionamos el botón “Save”, y con esta regla, solo el usuario seleccionado tendrá la opción de modificar la rama “máster”.

Fuente: elaboración propia.

Apéndice 4. Pasos previos para el correcto uso de la aplicación de ANDROID

Descripción del uso actual de la aplicación de Android

La aplicación de Android es un punto que se necesita hablar, debido a que puede ser de mucha utilidad, pero al momento se le da muy poco uso.

El problema es que, actualmente, no se sabe cómo usarla correctamente porque requiere un paso extra en el servidor.

Este paso que se explicara a continuación es la base para poder utilizar la aplicación de Android. El fin de esta sección es clarificar lo anterior para poder hacer un uso correcto de la aplicación.

Creación de hojas de calificación

Cuando se inicia la sesión, primero es necesario dirigirse al menú “Mis Cursos”, después de presionarlo, aparecerá una pantalla como la que sigue:

Continuación apéndice 4.

Figura 1. **Mis cursos**



Fuente: elaboración propia.

En la pantalla anterior se debe ingresar a la sesión “Gestión de Notas”, al seleccionar esta opción aparecerá una pantalla como la que sigue:

Continuación apéndice 4.

Figura 2. Actividades

The screenshot displays a web interface for course planning, divided into two main sections: 'Ponderación del Curso' and 'Planificación'.

Ponderación del Curso

Categoría	Nota	P.N.	P.C.
Proyectos	25.00	No	No
Tareas	25.00	No	No
Laboratorio	25.00	No	No
Examen Final	25.00	No	No
Total	100.00		

Ponderación de Laboratorio

Categoría	Nota	P.N.
Exámenes Cortos	50.00	No
Proyectos	50.00	No
Total	100.00	

Planificación

Actividades de Clase | Actividades de Laboratorio | Actividades sin Métrica | Actividades Asignadas

■ Notas arriba del promedio ■ Notas en el promedio ■ Notas abajo del promedio [Ver Detalle](#)

Exámenes Cortos (50.00 pts)

Nombre	Nota	Fecha Inicio	Fecha Final	Actividades Entregadas	Promedio de Notas	Acciones
prueba	50.00	2019-01-19	2019-01-19	1 / 6	75.0	

Proyectos (50.00 pts)

Nombre	Nota	Fecha Inicio	Fecha Final	Actividades Entregadas	Promedio de Notas	Acciones
proyecto final	50.00	2019-04-11	2019-04-30	0 / 6	0.0	

Fuente: elaboración propia.

En esta pantalla, la opción “Definir plantilla” de la sección de proyectos. Después de seleccionar esta opción aparecerá una pantalla como la siguiente:

Continuación apéndice 4.

Figura 3. Definición de plantilla

proyecto final

[← Atras](#)

Definición de categoría

Categoría:

Nota:

¿Desea que las actividades dentro de esta categoría sean promediadas?
Si No

[+ Agregar Categoría](#)

Gestión hoja de calificación

Estado: **Incompleto**

Descripción de la ponderación	Nota	P.N.	Acciones
Total	0.0		

[Finalizar](#)

Fuente: elaboración propia.

En esta pantalla se indican los valores que serán evaluados en la calificación del proyecto. Para cada sección creada, deberás ingresar los parámetros a evaluar. Esto se hace presionando el símbolo “+” ubicado en la fila de la sección creada.

Continuación apéndice 4.

Figura 4. Creación de parámetros

Definición de hoja de calificación.
proyecto final

[← Atras](#)

Definición de categoría

Categoría:

Nota:

¿Desea que las actividades dentro de esta categoría sean promediadas? Si No

[Agregar Categoría](#)

Gestión hoja de calificación

Estado: **Incompleto**

Descripción de la ponderación	Nota	P.N.	Acciones
Manual técnico	10	No	+ -
Manual de usuario	10	No	+ -
Funcionalidad	75	No	+ -
Presentación	5	No	+ -
Total	100.0		

[Finalizar](#)

Fuente: elaboración propia.

Ahora, se ingresa a todos los parámetros que creas necesarios y presionas el botón finalizar.

Continuación apéndice 4.

Figura 5. Guardar hoja

The screenshot shows a web application interface for saving a grading sheet. It is divided into two main sections:

- Definición de categoría:** This section contains two input fields for 'Categoría:' and 'Nota:'. Below these is a checkbox question: '¿Desea que las actividades dentro de esta categoría sean promediadas?' with a 'Si' checkbox and a radio button. A green button labeled 'Agregar Categoría' is positioned below the form.
- Gestión hoja de calificación:** This section shows the state 'Estado: Completo'. It contains a table with the following data:

Descripción de la ponderación	Nota	P.N.	Acciones
Manual técnico	10	No	[Icono] [+] [Icono]
- Especificaciones	10		[Icono] [Icono]
Manual de usuario	10	No	[Icono] [+] [Icono]
- Guía de uso	10		[Icono] [Icono]
Funcionalidad	75	No	[Icono] [+] [Icono]
- Conexión con servicio web	25		[Icono] [Icono]
- Guardado en base de datos	25		[Icono] [Icono]
- Sincronización de datos	25		[Icono] [Icono]
Presentación	5	No	[Icono] [+] [Icono]
- Interfaz	5		[Icono] [Icono]
Total	100.0		

A blue button labeled 'Finalizar' is located at the bottom right of the interface.

Fuente: elaboración propia.

Al presionar finalizar la hoja será guardada y ya se podrá utilizar para calificar proyectos en la aplicación de Android.

Fuente: elaboración propia.