



Universidad de San Carlos de Guatemala

Facultad de Ingeniería

Escuela de Ingeniería en Ciencias y Sistemas

**DESARROLLO E IMPLEMENTACIÓN DE PRUEBAS AUTOMÁTICAS PARA LA
PLATAFORMA DEL SISTEMA DE CONTROL DEL PROYECTO DE DESARROLLO DE
TRANSFERENCIA TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS
Y SISTEMAS DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

Jerson Eduardo Villatoro Nova

Asesorado por Ing. William Estuardo Escobar Argueta

Guatemala, junio de 2023

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**DESARROLLO E IMPLEMENTACIÓN DE PRUEBAS AUTOMÁTICAS PARA LA
PLATAFORMA DEL SISTEMA DE CONTROL DEL PROYECTO DE DESARROLLO DE
TRANSFERENCIA TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS
Y SISTEMAS DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

JERSON EDUARDO VILLATORO NOVA

ASESORADO POR EL ING. WILLIAM ESTUARDO ESCOBAR ARGUETA

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, JUNIO DE 2023

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANA	Inga. Aurelia Anabela Cordova Estrada
VOCAL I	Ing. José Francisco Gómez Rivera
VOCAL II	Ing. Mario Renato Escobedo Martinez
VOCAL III	Ing. José Milton De León Bran
VOCAL IV	Br. Kevin Vladimir Armando Cruz Lorente
VOCAL V	Br. Fernando José Paz González
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANA	Inga. Aurelia Anabela Cordova Estrada
EXAMINADORA	Inga. Floriza Felipa Ávila Pesquera de Medinilla
EXAMINADOR	Ing. Sergio Leonel Gómez Bravo
EXAMINADOR	Ing. Carlos Alfredo Azurdia Morales
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

HONORABLE TRIBUNAL EXAMINADOR

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

**DESARROLLO E IMPLEMENACIÓN DE PRUEBA AUTOMÁTICAS PARA LA PLATAFORMA
DEL SISTEMA DE CONTROL DEL PROYECTO DE DESARROLLO DE TRANSFERENCIA
TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS DE LA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha 05 de agosto de 2021.

A handwritten signature in black ink, enclosed within a large, loopy oval shape. The signature appears to read 'J. Villatoro'.

Jerson Eduardo Villatoro Nova



Guatemala 20 de enero de 2023

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

Estimado Ingeniero Argueta:

Atentamente por medio de la presente hago de su conocimiento que he revisado y aprobado el informe final de EPS titulado **“DESARROLLO E IMPLEMENTACIÓN DE PRUEBAS AUTOMÁTICAS PARA LA PLATAFORMA DEL SISTEMA DE CONTROL DE TRANSFERENCIA TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA”**. El informe ha sido elaborado por el estudiante: JERSON EDUARDO VILLATORO NOVA quien se identifica con código única de identificación 2942637562001 y registro académico 201442819, de la carrera de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala.

Sin otro particular me despido, atentamente.

Ing. William Estuardo Escobar Argueta
Asesor de EPS
Colegiado No. 11529
Escuela de Ingeniería en Ciencias y Sistemas

Universidad de San Carlos de
Guatemala



Facultad de Ingeniería
Unidad de EPS

Guatemala, 13 de marzo de 2023.
REF.EPS.DOC.155.03.2023.

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

Estimado Ingeniero Argueta Hernández:

Por este medio atentamente le informo que como Supervisora de la Práctica del Ejercicio Profesional Supervisado, (E.P.S) del estudiante universitario de la Carrera de Ingeniería en Ciencias y Sistemas, **Jerson Eduardo Villatoro Nova, Registro Académico 201442819 y CUI 2942 63756 2001** procedí a revisar el informe final, cuyo título es **DESARROLLO E IMPLEMENTACIÓN DE PRUEBAS AUTOMÁTICAS PARA LA PLATAFORMA DEL SISTEMA DE CONTROL DEL PROYECTO DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA.**

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

Sin otro particular, me es grato suscribirme.

Atentamente,

"Id y Enseñad a Todos"



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

FFAPdM/RA

Universidad de San Carlos de
Guatemala



Facultad de Ingeniería
Unidad de EPS

Guatemala, 21 de marzo de 2023.
REF.EPS.D.116.03.2023.

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

Estimado Ingeniero Alonzo:

Por este medio atentamente le envío el informe final correspondiente a la práctica del Ejercicio Profesional Supervisado, (E.P.S) titulado **DESARROLLO E IMPLEMENTACIÓN DE PRUEBAS AUTOMÁTICAS PARA LA PLATAFORMA DEL SISTEMA DE CONTROL DEL PROYECTO DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**, que fue desarrollado por el estudiante universitario **Jerson Eduardo Villatoro Nova, Registro Académico 201442819 y CUI 2942 63756 2001** quien fue debidamente asesorado por el Ing. William Estuardo Escobar Argueta 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"

The image shows a handwritten signature in blue ink over a circular official stamp. The stamp contains the text: 'Universidad de San Carlos de Guatemala', 'DIRECCIÓN', 'Unidad de Prácticas de Ingeniería y EPS', and 'Facultad de Ingeniería'.

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 28 de marzo de 2023

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

Respetable Ingeniero Alonzo:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación-EPS del estudiante **JERSON EDUARDO VILLATORO NOVA** carné 201442819 y CUI 2942 63756 2001, titulado: **"DESARROLLO E IMPLEMENTACIÓN DE PRUEBAS AUTOMÁTICAS PARA LA PLATAFORMA DEL SISTEMA DE CONTROL DEL PROYECTO DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA"** y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

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

Atentamente,



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

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA

LNG.DIRECTOR.132.EICCSS.2023

El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del Asesor, el visto bueno del Coordinador de área y la aprobación del área de lingüística del trabajo de graduación titulado: **DESARROLLO E IMPLEMENTACIÓN DE PRUEBAS AUTOMÁTICAS PARA LA PLATAFORMA DEL SISTEMA DE CONTROL DEL PROYECTO DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**, presentado por: **Jerson Eduardo Villatoro Nova**, procedo con el Aval del mismo, ya que cumple con los requisitos normados por la Facultad de Ingeniería.

"ID Y ENSEÑAD A TODOS"



Msc. Ing. Carlos Gustavo Alonzo
Ing. Carlos Gustavo Alonzo
Escuela de Ingeniería en Ciencias y Sistemas
Director

Escuela de Ingeniería en Ciencias y Sistemas

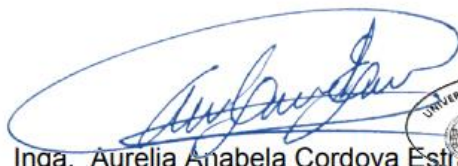
Guatemala, junio de 2023



Ingeniería Civil, Ingeniería Mecánica Industrial, Ingeniería Química, Ingeniería Mecánica Eléctrica, -Escuela de Ciencias, Regional de Ingeniería Sanitaria y Recursos Hidráulicos (ERIS), Maestría en Sistemas Mención construcción y Mención Ingeniería Vial. Carreras: Ingeniería Mecánica, Ingeniería Electrónica, Ingeniería en Ciencias y Sistemas, Licenciatura en Matemática, Licenciatura en Física. Centros: de Estudios Superiores de Energía y Minas (CESEM). Guatemala, Ciudad Universitaria, Zona 12, Guatemala, Centroamérica.

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: **DESARROLLO E IMPLEMENTACIÓN DE PRUEBAS AUTOMÁTICAS PARA LA PLATAFORMA DEL SISTEMA DE CONTROL DEL PROYECTO DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**, presentado por: **Jerson Eduardo Villatoro Nova**, después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, autoriza la impresión del mismo.

IMPRÍMASE:



Inga. Aurelia Anabela Cordova Estrada
Decana

Guatemala, junio de 2023
AACE/gaoc

ACTO QUE DEDICO A:

Dios	Por darme las bendiciones, fuerza y voluntad para cumplir mis objetivos.
Mis padres	Por su amor y ser mi pilar en el camino que ha decidido tomar.
Mi hermano	Por ser mi amigo y darme su apoyo.
Mis abuelos	Por su amor y apoyo incondicional.
Mis amigos	Por acompañarme a lo largo de mi camino.
Ingenieros	Por sus enseñanzas e invaluable experiencia.

AGRADECIMIENTOS A:

Universidad de San Carlos de Guatemala	Por ser mi casa de estudios y darme la formación académica y humana.
Mi familia	Por su amor incondicional y estar conmigo.
Mis amigos	Por acompañarme y apoyarme a lo largo de mi camino.
Mis catedráticos	Por brindarme sus valiosa enseñanza y experiencia.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES	V
LISTA DE SÍMBOLOS	VII
GLOSARIO	IX
RESUMEN	XI
OBJETIVOS.....	XIII
INTRODUCCIÓN	XV
1. FASE DE INVESTIGACIÓN	1
1.1. Antecedentes de la empresa	1
1.1.1. Reseña Histórica	1
1.1.2. Misión	2
1.1.3. Visión.....	3
1.1.4. Servicios que realiza.....	3
1.2. Descripción de las necesidades	4
1.2.1. Desarrollo de pruebas automáticas para las principales funciones de la Plataforma DTT	4
1.2.2. Continuidad de la administración y configuración de la arquitectura de pruebas	5
1.3. Priorización de las necesidades	6
2. FASE TÉCNICO PROFESIONAL	11
2.1. Descripción del proyecto	11
2.2. Investigación Preliminar para la solución del proyecto	12
2.2.1. Exploración de la Plataforma DTT	12
2.2.2. Ambiente de desarrollo	18

2.2.3.	Ambiente de automatización de QA	19
2.3.	Presentación de la solución al proyecto	20
2.3.1.	Descripción Técnica de la solución	21
2.3.2.	Descripción de productos	23
2.3.2.1.	Pruebas automatizadas para las principales funciones de la Plataforma DTT	23
2.3.2.2.	Continuación de la administración y configuración de la arquitectura de pruebas	24
2.4.	Costos del proyecto.....	24
2.4.1.	Recursos Humanos.....	24
2.4.2.	Recursos Materiales.....	25
2.4.3.	Presupuesto del proyecto.....	26
2.5.	Beneficios del proyecto	28
2.5.1.	Beneficios en el proceso de desarrollo.....	28
2.5.2.	Beneficios en el proceso control de calidad	29
3.	FASE ENSEÑANZA APRENDIZAJE	31
3.1.	Capacitación propuesta.....	31
3.1.1.	Introducción al ambiente de desarrollo y el ambiente de QA	31
3.1.2.	Práctica de Desarrollo de Pruebas.....	32
3.1.3.	Despliegue de Pruebas en Ambiente de QA.....	33
3.2.	Materiales Elaborados.....	33
3.2.1.	Manual de Usuario para Jenkins	33
3.2.2.	Plantilla para Control de Pruebas.....	34
3.2.3.	Guía de Desarrollo de Pruebas	35

4.	PRESENTACIÓN DE RESULTADOS	37
4.1.	Cuestionario a los interesados	37
4.2.	Inventario de prueba finalizadas.....	40
4.2.1.	Pruebas del rol administrador	40
4.2.2.	Pruebas del rol Catedrático	43
4.2.3.	Pruebas del rol Tutor Académico.....	44
4.2.4.	Pruebas del rol Estudiante.....	45
4.3.	Pruebas subidas al servidor de automatización en la nube.....	46
4.4.	Documentación de pruebas.....	51
	CONCLUSIONES	53
	RECOMENDACIONES	55
	REFERENCIAS	57
	APÉNDICES	59

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Pantalla principal de la Plataforma DTT	13
2.	Pantalla principal con rol Administrador	13
3.	Pantalla principal de Jenkins	20
4.	Primera parte, cuestionario al administrador de la Plataforma DTT	38
5.	Segunda parte, cuestionario al administrador de la Plataforma DTT	39
6.	Pantalla principal del servidor de automatización en la nube	46
7.	Carpeta de las pruebas del rol Administrador	47
8.	Pantalla de arranque de ejecución de prueba	48
9.	Pantalla de ejecución de prueba	49
10.	Reporte en la consola del servidor de automatización de pruebas	50
11.	Documentación de prueba	52

TABLAS

I.	Funcionalidades del rol Administrador	6
II.	Funcionalidades del rol catedrático	8
III.	Funcionalidades del rol de tutor académico	9
IV.	Funcionalidades del rol Estudiante	9
V.	Priorización de necesidades según rol	10
VI.	Funcionalidades identificadas del rol Administrador	14
VII.	Recursos Humanos....	25
VIII.	Recursos Materiales...	26
IX.	Costos del proyecto ...	27

X.	Funcionalidades del rol Administrador con pruebas automáticas implementadas	40
XI.	Funcionalidades del rol Catedrático con pruebas automáticas implementadas	43
XII.	Funcionalidades del rol Tutor Académico con pruebas automáticas implementadas	44
XIII.	Pruebas del rol Estudiante con pruebas automatizadas implementadas	45

LISTA DE SÍMBOLOS

Símbolo	Significado
Q	Quetzal
GB	Unidad de almacenamiento de información equivalente a 10^9 bytes
MB	Unidad de almacenamiento de información que equivale a 10^6 bytes.

GLOSARIO

Docker	Es una tecnología que permite crear imágenes y contenedores. Un contenedor es una unidad estándar de software que empaqueta código y todas sus dependencias para que la aplicación se ejecute de forma rápida y confiable de un entorno informático a otro. Una imagen de contenedor de Docker es un paquete de software ligero, independiente y ejecutable que incluye todo lo necesario para ejecutar una aplicación: código, tiempo de ejecución, herramientas del sistema, bibliotecas del sistema y configuraciones.
IntelliJIDE	Es un entorno de desarrollo integrado (IDE) para el desarrollo de programas informáticos.
Jenkins	Servidor de automatización de código abierto autónomo que se puede utilizar para automatizar todo tipo de tareas relacionadas con la creación, prueba y entrega o implementación de software.
Rest-Assured	DSL (DomainSpecificLanguages) de Java para simplificar las pruebas de servicios basados en REST contruidos sobre peticiones HTTP. Admite POST, GET, PULL, DELETE, OPTIONS, PATCH y HEAD y se puede utilizar para validar y verificar la respuesta de estas solicitudes. Se puede considerar como DSL

cualquier lenguaje que se especialice en modelar o resolver un conjunto de problemas. Este conjunto de problemas se denominan dominio de aplicación o empresarial.

Selenium

Es un proyecto con una gran variedad de herramientas y bibliotecas que permiten la automatización de navegadores web. Proporciona extensiones para emular la interacción del usuario con los navegadores, un servidor distribución para escalar la asignación del navegador y la infraestructura para implementaciones de la especificación W3C WebDriver que le permite escribir código intercambiable para los principales navegadores web.

TestNG

Es un marco de prueba en Junit y NUnit pero que presenta algunas funcionalidades nuevas que lo hacen más poderoso y fácil de usar, como anotaciones, grupos de ejecución, y otros. TestNG está diseñado para cubrir todas las categorías de pruebas: unitarias, funcionales, *end-to-end*, integración, entre otros.

RESUMEN

En la actualidad la Escuela de Ingeniería en Ciencias y Sistemas cuenta con un Sistema de Control del Proyecto de Desarrollo de Transferencia Tecnológica (DTT) el cual le facilita la gestión de los estudiantes, catedráticos, cursos que se imparten, asignación de cursos, ingresos de notas, entre otras funcionalidades que se han automatizado para llevar un mejor control de catedráticos y personal administrativo.

A lo largo de su tiempo de funcionamiento se han ido añadiendo funcionalidades por parte de diversos estudiantes, lo que ha ido añadiendo a su vez complejidad al sistema, en especial a al tema de Control de Calidad, que abarca las diferentes pruebas y validaciones que se realizan a la plataforma previo al despliegue en el ambiente de producción. Actualmente estas las pruebas se realizan de forma manual, esto a su vez genera un desgaste en los programadores y consume tiempo ya que es necesario validar que al agregar una nueva funcionalidad no se hayan visto afectadas las funcionalidades ya implementadas.

En vista de esta necesidad ya se desarrolló una arquitectura de pruebas automáticas implementada como parte de un ejercicio profesional supervisado con duración de tres meses, la cual incluye la implementación de dos contenedores que se puedan configurar en computadoras que cumplan con los requerimientos mínimos. Estos contenedores realizarán el trabajo bajo la tecnología Docker y DockerFile (archivos de configuración), lo que permite que las soluciones a desarrollar puedan ser portátiles en diferentes equipos de computadoras.

OBJETIVOS

General

Desarrollar pruebas automatizadas sobre las funcionalidades asociadas a los diferentes roles: administrador, estudiantes, tutores académicos y catedrático, de la plataforma del Sistema de Control del Proyecto de Desarrollo de Transferencia Tecnológica (Plataforma DTT) de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la universidad de San Carlos de Guatemala con el fin de validar el funcionamiento correcto y reportar resultados.

Específicos

1. Desarrollar pruebas de regresión para las funcionalidades pertenecientes a los roles administrador, estudiante, tutor académico y catedrático de la plataforma del Sistema de Control del Proyecto de Desarrollo Tecnológico.
2. Definir las configuraciones necesarias en el contenedor de Jenkins para la ejecución de las pruebas de forma automática y notificación al administrador del estado de finalización de éstas.
3. Configurar las pruebas automatizadas en el contenedor de Jenkins para que reporten el resultado después de su ejecución.

INTRODUCCIÓN

La plataforma del Sistema de Control del Proyecto de Desarrollo de Transferencia Tecnológica (Plataforma DTT) es una herramienta desarrollada y usada por la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala con el propósito de agilizar y facilitar los distintos procesos de la Escuela relacionados con los distintos roles que existen en la misma: estudiantes, catedráticos, tutores académicos, tutores de infraestructura, administradores, entre otros.

Esta plataforma ha ido creciendo conforme se han ido añadiendo a ella diferentes módulos que aumentan su funcionalidad y a su vez la vuelven una herramienta más completa y compleja, que permite automatizar y agilizar diversos trámites como el manejo de los cursos, la gestión de los tutores académicos y sus prácticas finales, la oposición a proyectos, por mencionar algunos procesos de los que actualmente se gestionan con la ayuda de la Plataforma DTT.

Al ir creciendo de esta forma en complejidad, la gestión del control de calidad (QA) se ha vuelto un proceso más engorroso y tardado. Por lo que en actualmente se encuentra en desarrollo una arquitectura de automatización de pruebas que pretende agilizar este proceso y optimizar tiempo en el desarrollo de futuros módulos para la plataforma. Sin embargo, la implementación de esta arquitectura no contempla el desarrollo de pruebas para las funcionalidades que ya se encuentran implementadas en la plataforma.

En vista de esta necesidad, se plantea el desarrollo de pruebas de regresión, es decir, pruebas enfocadas en las funcionalidades que ya se encuentran integradas a la plataforma para los diferentes roles que participan en la misma. Así también, es necesario que estas pruebas sean automatizadas de forma que optimicen el tiempo de desarrollo y mantenimiento de la plataforma, así como el depuramiento de los errores de las funcionalidades ya existentes.

1. FASE DE INVESTIGACIÓN

Para la fase de investigación se ha reunido información relacionada con la institución en la cual se ejecutó el proyecto, la Escuela de Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala. Datos relacionados con su historia, visión, misión y los servicios que realiza en su respectivo campo que es la formación de ingenieros en ciencias y sistemas que cuenten con las competencias necesarias para hacer frente al mercado actual de ciencias de la información y la tecnología.

1.1. Antecedentes de la empresa

La Escuela de Ingeniería en Ciencias y Sistemas es el ente designado por la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, para la planeación, organización, dirección y control de los grados, postgrados y maestrías, cuyo tema central de estudio son las ciencias de la computación y los sistemas de información. Los programas académicos ofrecidos por la Escuela de Ingeniería en Ciencias y Sistemas poseen como motivo principal apoyar la solución de problemas del desarrollo integral del país a través del potencial que contiene la aplicación de técnicas propias del área de computación y de la visión de sistemas.

1.1.1. Reseña Histórica

La carrera de Ingeniería en Ciencias y Sistemas fue fundada en 1971 ante la necesidad de la sociedad de formar profesionales con las competencias necesarias en el área de tecnologías de la información que entraba en su auge.

A lo largo de su trayectoria la población de estudiantes ha crecido dentro de la Escuela de Ingeniería en Ciencias y Sistemas, lo que derivó en implementar un sistema de control para los diferentes trámites y cursos impartidos por dicha escuela. Es por ello por lo que surge el proyecto DTT.

El proyecto de la plataforma de Desarrollo de Transferencia Tecnológica (Plataforma DTT) surge a partir de la contribución de profesionales que participaron en la Reforma Curricular 2011-2012, que aportaron ideas con el fin de mejorar la calidad de la educación de la Escuela. El proyecto consistió en continuar lo que se inició en 2012, con un conjunto de conferencias con temas de gran interés para el estudiantado y para docentes de la carrera de Ingeniería en Ciencias y Sistemas. La metodología del proyecto DTT se enfocaba en la participación de los estudiantes que realizan sus prácticas finales como asistentes de cátedra en el proyecto DTT durante el período de junio a noviembre de 2012.

El proyecto DTT tiene como fin ser un vínculo entre los estudiantes de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala que realizan su Práctica Final con el mundo laboral, la investigación, el emprendimiento y el estado.

1.1.2. Misión

Desarrollar en el alumno las competencias que garantizan el éxito en la construcción del conocimiento a través de los diferentes estilos de aprendizaje y fomentar la investigación permanente para permitir una mejor calidad de vida a la comunidad. Teniendo en cuenta las opciones del mercado actual en el país (logísticas, administración, tecnología de la información, finanzas, contabilidad,

comercio, entre otros.), y también el mercado internacional, hacen una gran demanda global y competitividad en la actualidad (About us, s.f.).

1.1.3. Visión

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

1.1.4. Servicios que realiza

La Escuela de Ciencias y Sistemas es la encargada de realizar las gestiones y planificaciones de cualquier actividad relacionada con la formación académica de los estudiantes de la carrera de Ingeniería en Ciencias y Sistemas como de aquellos que se asigna a cursos que son organizados por la carrera. Tiene entre sus funciones el supervisar y evaluar las técnicas utilizadas en la enseñanza de los cursos a su carga, como de las actividades relacionadas con sus catedráticos, tutores académicos y estudiantes en general.

La escuela vela por mantener al nivel de los estándares del mercado actual relacionado con las diferentes tecnologías los programas académicos y su pénsum de estudios, a fin de cumplir con las expectativas que se tienen de un profesional del área de sistemas.

1.2. Descripción de las necesidades

En la actualidad la Escuela de Ingeniería en Ciencias y Sistemas cuenta con un Sistema de Control de Desarrollo de Transferencia Tecnológica (DTT) el cual facilita la gestión de los estudiantes, catedráticos, tutores académicos, cursos que se imparten, control de notas, entre otras funcionalidades que han agilizado diferentes trámites administrativos. Al ir creciendo y evolucionando se ha añadido funcionalidades que satisfacen nuevas necesidades las cuales se listan más adelante.

1.2.1. Desarrollo de pruebas automáticas para las principales funciones de la Plataforma DTT

Con el crecimiento de la plataforma y el desarrollo de nuevas funcionalidades surge la necesidad de mantener un control de calidad sobre la Plataforma de manera que se garantice el correcto funcionamiento tanto de los módulos más antiguos como de los que se agregan posteriormente. Para cubrir esta necesidad se ha implementado una arquitectura para el control de calidad de la plataforma DTT como parte de un proyecto de EPS realizado anterior a este. Dicho proyecto tuvo una duración de tres meses en los cuales se armó una arquitectura a la que se le debe dar continuidad.

Con esta arquitectura montada se necesita dar continuidad al desarrollo de pruebas automáticas haciendo uso de un ambiente de pruebas implementado en el ambiente de desarrollo. Para el desarrollo de estas pruebas se hace uso de IntelliJ IDEA como IDE de programación de las pruebas automáticas. Se configuró esta herramienta con las diferentes librerías por medio de Maven y se hizo uso de Jinja2, TestNG, RestAssured, entre otras dependencias necesarias para el funcionamiento de las pruebas automáticas.

Para el control del código de las pruebas desarrolladas se hizo uso de GitLab para poder gestionar el repositorio de versionamiento donde se almacena el código fuente del proyecto de pruebas automáticas. También cabe mencionar que en este ambiente se hace uso de los JenkinsFile, los cuales son archivos que usa la herramienta Jenkins para la ejecución automatizada de las pruebas.

Para identificar las funcionalidades se hizo una exploración de la plataforma y se listaron las diferentes funcionalidades de acuerdo con su respectivo rol.

1.2.2. Continuidad de la administración y configuración de la arquitectura de pruebas

Uno de los productos dejados por el proyecto de EPS anterior es un contenedor con la herramienta de orquestación Jenkins, el cual “es un servidor de automatización autónomo de código abierto que se puede utilizar para automatizar todo tipo de tareas relacionadas con la creación, prueba y entrega o implementación de software” (Jenkins User Documentation, s.f.). Una de las necesidades que cubre este proyecto es la continua configuración de este contenedor para la ejecución de las pruebas desarrolladas en el ambiente de desarrollo y la notificación de los resultados, por medio de reportes de cada prueba que se ejecute, pudiendo visualizar los escenarios que fueron probados o bien los que fallaron.

Para continuar con su desarrollo y mantenimiento desde otros equipos y sistemas operativos se hará uso de los DockerFile, los cuales son archivos que permiten definir contenedores, y estarán almacenados en un repositorio en GitLab para su versionamiento.

1.3. Priorización de las necesidades

Para la identificación y priorización de las necesidades que se identificaron primero las diferentes funcionalidades que tiene la plataforma en cada rol. Para la realización de esta tarea se hizo una exploración de la plataforma haciendo uso de los diferentes roles: administrador, catedrático, tutor académico y estudiante, haciendo uso de un entorno de pruebas de la Plataforma DTT.

Al haber listado las funcionalidades de cada rol se procedió a tabularlas como se muestra a continuación.

Tabla I. **Funcionalidades del rol Administrador**

No.	Funcionalidad
1.	Creación, consulta, actualización y actualización de usuarios
2.	Creación, consulta, actualización y actualización de membresías (roles)
3.	Creación, consulta, actualización y actualización eventos
4.	Gestión de membresías
5.	Envío de emails y gestión de archivos
6.	Creación, consulta, actualización y actualización de proyectos
7.	Gestión de proyectos (asignación y cambios en la asignación)
8.	Creación, consulta, actualización y actualización de reportes de proyecto

Continuación de la tabla I.

No.	Funcionalidad
9.	Gestión de catedráticos
10.	Creación, consulta, actualización y eliminación de estudiantes
11.	Gestión de fotos de estudiantes
12.	Gestión de roles (modificar/agregar rol en usuarios)
13.	Reporte de notas
14.	Reporte de asistencia
15.	Reporte de asistencia de hoy
16.	Reporte de huellas registradas
17.	Reporte de asignación de horarios
18.	Aprobación o rechazo de solicitud de falta
19.	Creación, consulta, actualización y eliminación días de asueto
20.	Reporte de resoluciones
21.	Aprobación o rechazo de resolución de permisos por falta
22.	Creación, consulta, actualización y actualización de días de asueto
23.	Creación, consulta, actualización y actualización de horario por día
24.	Asignación de horario a tutor
25.	Inicialización de semestre
26.	Creación y eliminación de excepciones de horario
27.	Configuración de capturas de huella
28.	CRUD de parámetros de penalización de asistencia

Continuación de la tabla I.

No.	Funcionalidad
29.	Configuración de parámetros del sistema
30.	Carga de horarios
31.	Reporte de horarios por semestre
32.	Reporte de practicantes finales
33.	Generación de certificado de período de período completo para practicantes finales

Fuente: elaboración propia.

Tabla II. **Funcionalidades del rol catedrático**

No.	Funcionalidad
1	CRUD de actividades del curso
2	Reporte de carga de notas
3	Reporte de gestión de revalidación de laboratorio
4	Reporte de gestión de equivalencias de laboratorio
5	Envío de mensaje a estudiantes
6	Calificación de reportes del tutor académico

Fuente: elaboración propia.

Tabla III. **Funcionalidades del rol de tutor académico**

No.	Funcionalidad
1	Carga de estudiantes
2	Creación, consulta, actualización y eliminación de actividades del laboratorio
3	Carga de notas
4	Enviar mensaje a los estudiantes asignados al curso
5	Enviar reporte de prácticas a calificar

Fuente: elaboración propia.

Tabla IV. **Funcionalidades del rol Estudiante**

	Rol Estudiante
1	Visualización de información de proyecto asignado para prácticas finales
2	Visualización de cursos asignados
3	Generar certificado de practica final
4	Visualización del detalle de notas de curso y laboratorio
5	Lectura de mensajes recibitos por el tutor o catedrático
6	Visualización de estadísticas de curso
7	Visualización del perfil
8	Visualización de reporte de clase

Fuente: elaboración propia.

Al haber identificados todas las funcionalidades se realizó una reunión con los interesados del proyecto, el ingeniero encargado de la plataforma, Ing. Miguel Marín y el estudiante Gustavo Alonzo quien estuvo a cargo de la primera fase de la implementación de la arquitectura. El objetivo de esta reunión fue el priorizar la lista de funcionalidades para proceder a realizar un calendario en función de dicha prioridad. A continuación, se muestra una tabla que resume el orden en el que se decidió priorizar los roles y por tanto sus respectivas funcionalidades.

Tabla V. **Priorización de necesidades según rol**

Rol	Prioridad	Justificación
Administrador	Alto	Es el rol con mayor cantidad de funcionalidades y es fundamental para el funcionamiento de la plataforma.
Catedrático	Alto	Este rol cuenta con muchas funciones relacionadas con la administración de los cursos e interactúa con los roles del tutor y el estudiante.
Tutor Académico	Medio	El tutor tiene mucha interacción con sus estudiantes dado su papel en el desarrollo de los laboratorios de los cursos a los que se encuentra asignado.
Estudiante	Bajo	Este rol solo cumple con funciones de observador en los cursos a los que se encuentra asignado.

Fuente: elaboración propia.

2. FASE TÉCNICO PROFESIONAL

A continuación, se muestran diferentes detalles acerca del proyecto que se propone como solución a la problemática planteada anteriormente sobre los costos, beneficios, la investigación previa entre otros aspectos.

2.1. Descripción del proyecto

El proyecto consiste en la continuación del desarrollo e implementación de la arquitectura que se detalla en el capítulo 1. Esto conlleva la administración, configuración y desarrollo de pruebas de regresión en el ambiente de desarrollo preparado con las herramientas necesarias, el cual tiene instalado Java y las herramientas para el desarrollo de las pruebas automatizadas que son: TestNG, Selenium, Jinja2, Maven e IntelliJIDE, este último se utilizará para el ambiente de desarrollo de pruebas, ya que cuenta con funcionalidades para agregar herramientas para el desarrollo de pruebas automatizadas (Prepare for testing, 2021). Para el manejo del código y control de versionamiento se hace uso de Git y GitLab para mantener el código en línea.

También se dio continuidad con la administración y configuración en el contenedor de automatización de pruebas, el cual se encuentra equipado con Jenkins y está destinado a la ejecución de las pruebas de manera automática de la Plataforma DTT de la Escuela de Ciencias y Sistemas. Al realizar las pruebas notifica, si así lo requiere el usuario que ejecute dicha prueba, con correo la finalización de esta e incluye un enlace que dirige al reporte de consola que genera la ejecución. En este contenedor se configuran los *Jobs* que automatizan la tarea de ejecución de las pruebas desarrolladas en el ambiente de desarrollo.

Las pruebas que se han desarrollado se encuentran enfocadas a los diferentes roles que tiene la Plataforma DTT y sus respectivas funciones. Los roles principales que tiene la plataforma y en los cuales este proyecto se enfocó fueron: administrador, estudiante, catedrático y tutor académico. Cada rol tiene diferentes funciones y permisos para realizar diferentes acciones e interactuar con los diferentes procesos de la Escuela de Ciencias y Sistemas.

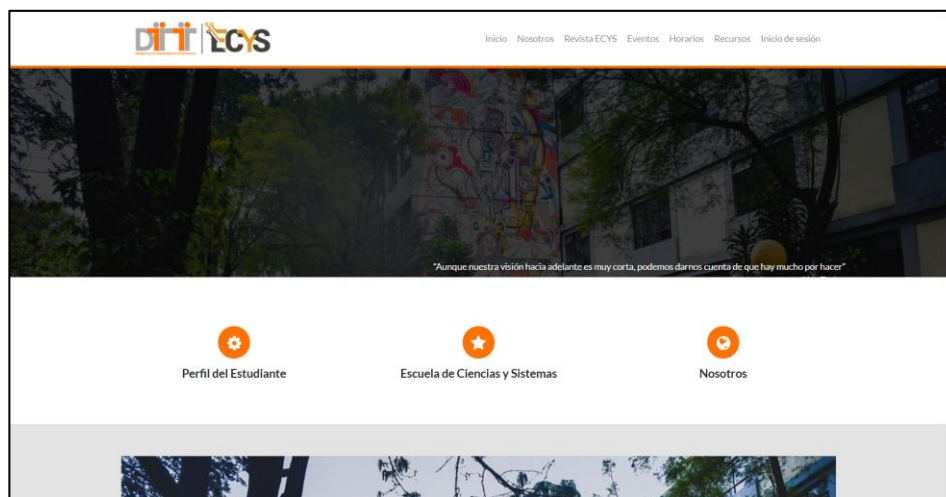
2.2. Investigación Preliminar para la solución del proyecto

La investigación previa a la realización del proyecto se divide en una fase de exploración de la plataforma y capacitación en el ambiente de desarrollo y en el ambiente de automatización.

2.2.1. Exploración de la Plataforma DTT

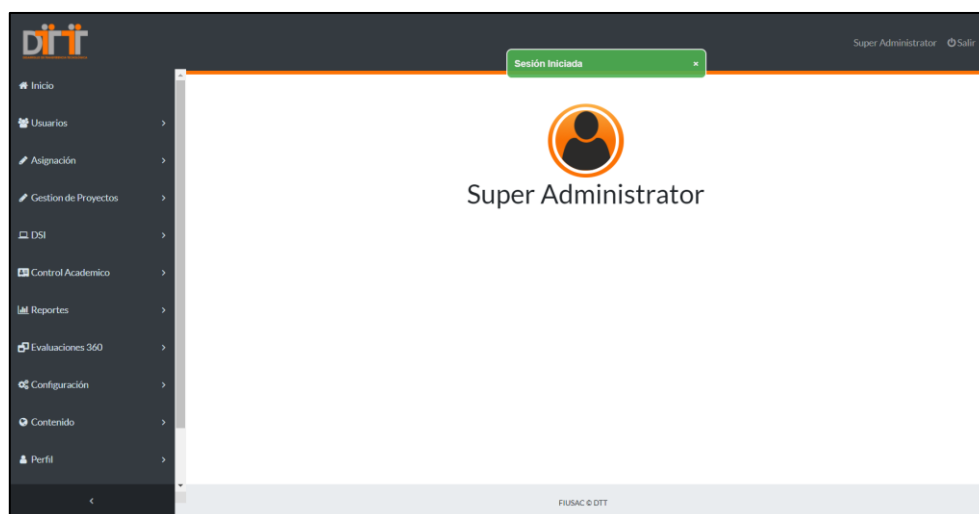
Para dar inicio con el proyecto se hizo una fase de exploración de la plataforma DTT en la cual haciendo uso del ambiente de QA se ingresó a dicha plataforma usando diferentes roles: Administrador, estudiante, catedrático, tutor académico. Se identificaron las funcionalidades de acuerdo con las vistas y al uso que cada uno de estos usuarios pudiera darle a la plataforma DTT.

Figura 1. **Pantalla principal de la Plataforma DTT**



Fuente: [Fotografía de Jerson Eduardo Villatoro Nova]. (Captura de pantalla, 2022). Colección Particular. Guatemala.

Figura 2. **Pantalla principal con rol Administrador**



Fuente: [Fotografía de Jerson Eduardo Villatoro Nova]. (Captura de pantalla, 2022). Colección particular. Guatemala.

Al ser identificadas las funcionalidades, se procedió a listarlas para luego ser presentadas al administrador de la plataforma con quien se detallaron y priorizaron aquellas que fueran urgentes de acuerdo con las necesidades de los usuarios y la frecuencia con la cual dichas funcionalidades son usadas durante el ciclo académico.

A continuación, se muestra la tabla de las primeras cincuenta funcionalidades del rol administrador que fueron identificadas. Al ser este el rol con más funciones en la plataforma se tomó la decisión de dar prioridad a las pruebas orientadas al mismo.

Tabla VI. **Funcionalidades identificadas del rol Administrador**

No.	Funcionalidad	Descripción
1	Visualización de todos los usuarios	Despliega tabla con la información de los usuarios al entrar en la página correspondiente
2	Crear usuario	Crea un nuevo usuario en el sistema
3	Buscar usuario	Realiza búsqueda de usuarios aplicando diferentes filtros (id, nombre de usuario, nombre, etc.)
4	Visualización de un usuario	Despliega la información correspondiente a un usuario en específico
5	Eliminar usuario	Elimina un usuario del sistema
6	Editar usuario	Realizar cambios en la información de un determinado usuario
7	Exportar información de usuarios	Exporta la información de uno o varios usuarios a un documento (pdf, csv, json, html, tsv)
8	Visualizar la membresía de usuario	muestra el rol al que está relacionado un determinado usuario

Continuación de la tabla VI.

No.	Funcionalidad	Descripción
9	Editar membresía de usuario	Edita el rol de un usuario, por lo tanto, las acciones que puede realizar dentro de la plataforma
10	Eliminar membresía de usuario	Elimina el rol al que se encuentra relacionado un usuario determinado
11	Visualización de Eventos de usuario	Muestra una tabla con todas las acciones y su respectiva información que ha realizado un determinado usuario
12	Visualización de detalles de evento	Muestra información sobre un evento determinado relacionado a un usuario en particular
13	Editar evento	Permite editar la información (marca de tiempo, IP cliente, id usuario, origen, descripción) un evento
14	Eliminar evento	Elimina un evento del registro de eventos
15	Agregar membresía a usuario	Agrega un rol a un usuario determinado dando los permisos correspondientes asociados a dicho rol
16	Agregar caso a usuario	Agregar un caso para dar seguimiento a un usuario determinado
17	Editar caso	Edita un caso asociado a un usuario determinado
18	Eliminar caso	Elimina un caso asociado a un usuario determinado
19	Agregar proyecto a usuario	Asocia un proyecto a un usuario, función que se usa para los proyectos de prácticas finales con los estudiantes
20	Visualizar proyectos de usuario	Muestra en una tabla todos los proyectos asociados a un estudiante determinado

Continuación de la tabla VI.

No.	Funcionalidad	Descripción
21	Editar proyecto de usuario	Permite modificar la información (proyecto, horas, periodos) sobre un proyecto asociado a un usuario
22	Eliminar proyecto	Elimina del registro el proyecto asociado a un usuario
23	Visualizar proyecto de usuario	Muestra la información un proyecto (id, proyecto, horas, periodos, entre otros.) asociado a un usuario
24	Crear reporte de proyecto	Crea un registro de reporte y lo asocia a un proyecto determinado
25	Visualizar reporte de proyecto	Muestra al administrador la información asociada a un reporte de proyecto
26	Editar reporte de proyecto	Permite cambiar la información asociada un reporte asociado a un proyecto
27	Eliminar reporte de proyecto	Elimina el registro de un reporte de proyecto
28	Visualizar todos los catedráticos	Despliega tabla con la información de los catedráticos en el sistema al entrar en la página correspondiente
29	Visualización de un catedrático	Despliega la información correspondiente a un catedrático en específico
30	Crear registro de catedrático	Crea un nuevo usuario en el sistema
31	Buscar catedrático	Realiza búsqueda de catedrático aplicando diferentes filtros (id, nombre de usuario, nombre, entre otros.)
32	Eliminar catedrático	Elimina un catedrático del sistema
33	Editar catedrático	Realizar cambios en la información de un determinado catedrático

Continuación de la tabla VI.

No.	Funcionalidad	Descripción
34	Visualizar todos los estudiantes	Despliega tabla con la información de los estudiantes en el sistema al entrar en la página correspondiente
35	Visualización de un estudiante	Despliega la información correspondiente a un estudiante en específico
36	Crear registro de estudiante	Crea un nuevo estudiante en el sistema
37	Buscar estudiante	Realiza búsqueda de estudiante aplicando diferentes filtros (id, nombre de usuario, nombre, etc.)
38	Eliminar estudiante	Elimina un registro de estudiante del sistema
39	Editar estudiante	Realizar cambios en la información de un determinado estudiante
40	Ver foto	Despliega una ventana con la foto del estudiante y las opciones para aceptarla o rechazarla
41	Aceptar o Rechazar foto	Permite al administrador determinar si la foto es válida o si el estudiante debe volver a subirla
42	Visualizar todos los roles	Despliega tabla con la información de los estudiantes en el sistema al entrar en la página correspondiente
43	Visualización de un rol	Despliega la información correspondiente a un estudiante en específico
44	Crear registro de rol	Crea un nuevo estudiante en el sistema
45	Buscar rol	Realiza búsqueda de estudiante aplicando diferentes filtros (id, nombre de usuario, nombre, etc.)
46	Eliminar rol	Elimina un registro de estudiante del sistema

Continuación de la tabla VI.

No.	Funcionalidad	Descripción
47	Editar rol	Realizar cambios en la información de un determinado estudiante
48	Visualizar usuarios asociados a rol	Se muestra todos los usuarios que se encuentran asociados a un rol determinado de forma tabulada
49	Calcular nota	Esta funcionalidad se encarga de calcular la nota de asistencia, tanto de inasistencias en DTT modulo DSI
50	Generar reporte de notas	Muestra en una tabla las notas de los diferentes tutores académicos en un período determinado

Fuente: elaboración propia.

Luego de presentar esta lista de funcionalidades del rol administrador como de los demás roles al administrador de la plataforma DTT se procedió a ordenar en función de su urgencia y a partir de ello se elaboró el cronograma del proyecto.

2.2.2. Ambiente de desarrollo

Habiendo explorado la plataforma e identificado las funcionalidades se procedió a una serie de capacitaciones con el anterior encargado del proyecto, las cuales estaban relacionadas a familiarizar a los usuarios con el ambiente de desarrollo.

Para reducir la curva de aprendizaje, el encargado del proyecto previo a la realización de este proyecto realizó una serie de sesiones en las cuales se tomó participación. El fin de estas reuniones era familiarizar a los desarrolladores

en el ambiente de desarrollo de las pruebas. Se expuso el uso de IntelliJ IDEA para la codificación de las pruebas de regresión.

Con el apoyo de estas sesiones y el acercamiento directo al código escrito por el compañero se identificó el patrón de diseño que se ocupó durante la realización de este proyecto. Este patrón consiste básicamente en una clase para la definición de los elementos web de la plataforma, otra clase para la definición de las acciones que se realizaran sobre estos elementos y al final una clase para definir el orden lógico de las acciones y a su vez detectar indicadores para verificar el éxito o fracaso de una prueba.

2.2.3. Ambiente de automatización de QA

El ambiente de automatización de pruebas consta de un contenedor de Jenkins con el cuál se automatizan las pruebas haciendo uso de sus *Jobs*. Como primer acercamiento a esta herramienta al igual que para el ambiente de desarrollo el encargado del proyecto anterior realizó una sesión para explicar de manera general el sistema y cómo se automatizaban las pruebas.

Antes de dar inicio al proyecto, durante de la fase de transición del proyecto anterior al actual se realizó una reunión en conjunto con el administrador de la Plataforma DTT con el fin de detallar el proceso de creación de los *Jobs* como también el proceso para ejecutarlos y ver su reporte de consola.

Figura 3. **Pantalla principal de Jenkins**

S	W	Nombre	Último Éxito	Último Fallo	Última Duración
		AQA-Login	N/D	N/D	N/D
		AQA-RolAcademic	N/D	N/D	N/D
		AQA-RolStudent	N/D	N/D	N/D
		AQA-RolTeacher	N/D	N/D	N/D
		QA-RolSuperAdministrator	N/D	N/D	N/D

Fuente: [Fotografía de Jerson Eduardo Villatoro Nova]. (Captura de pantalla, 2022). Colección particular. Guatemala.

Como parte de esta capacitación y para evaluar el nivel de comprensión de las herramientas y los diferentes ambientes se realizó el desarrollo de una prueba completa enfocada en alguna de las funcionalidades del rol administrador. La prueba elegida fue la de creación, edición, visualización y eliminación de usuarios de la plataforma DTT. Para ello se tomaron en cuenta todos los pasos que se han detallado en la sesión de capacitación orientada en el ambiente de desarrollo y se automatizó haciendo uso de Jenkins. Para corroborar su correcta implementación se realizó una sesión con el encargado del proyecto en función quien resolvió dudas e hizo comentarios para mejora de futuras pruebas.

2.3. **Presentación de la solución al proyecto**

La solución que se propone para este proyecto consiste en la continuación del desarrollo de pruebas automatizadas y la administración de los contenedores de QA que fueron delegados de la primera fase de la implementación de la arquitectura de QA.

2.3.1. Descripción Técnica de la solución

Paras alojar los contenedores que se hacen mención se hizo uso de una computadora con las siguientes características:

- 4 GB de Memora RAM
- 40 GB de Memoria de disco curo
- Sistema Operativo Linux (Ubuntu)
- Salida a Internet interfaz IPv4

Este contenedor es usado para desplegar el entorno para la ejecución, notificación y reportería de pruebas automatizadas para la Plataforma DTT de la Escuela de Ingeniería en Ciencias y Sistemas.

Para continuar con la administración y desarrollo de este contenedor fue necesario el dominio de las siguientes herramientas:

- Java para el desarrollo de las pruebas y las herramientas TestNG, Selenium, Jinja2, Maven, entre otras.
- Conocimiento para la configuración de Jenkins como herramienta de integración con los *plugins* necesarios para su administración.
- Configuración de Jenkins para su ejecución por medio de archivos JenkinsFile

Para el ambiente de desarrollo que tiene como fin ser el entorno de desarrollo de trabajo y el desarrollo de pruebas automáticas de la plataforma DTT de la Escuela de Ciencias y Sistemas de la Facultad de Ingeniería. Para dar continuidad a la administración y desarrollo de este ambiente es necesario el dominio de las siguientes herramientas:

- Java y las herramientas para realizar las pruebas automáticas: TestNG, Selenium, Jinja2, Maven, Python, entre otras.
- Configuración de IntelliJ Idea como herramienta de desarrollo y ejecución de las pruebas automáticas.
- GitLab como herramienta de versionamiento
- Integración de IntelliJ Idea con GitLab para clonar el código de las pruebas automáticas a desarrollar.

Tanto en el ambiente de desarrollo como en el de automatización de pruebas se hace uso de Selenium, una herramienta que proporciona extensiones para emular la interacción del usuario con los navegadores. “Es un *web driver* que proporciona una interfaz para escribir conjuntos de instrucciones que se pueden ejecutar indistintamente en muchos navegadores” (The Selenium Browser Automation Project, s.f.).

En cuanto a la priorización de las pruebas automáticas a desarrollar, estas se enfocan en las funcionalidades más utilizadas actualmente y validadas por la Escuela para su planificación en este proyecto. Para identificar estas funcionalidades es necesario un examen de la plataforma que permita explorar las diferentes funcionalidades desde los diferentes roles y llegar a un consenso

con el encargado de la plataforma para determinar aquellas que son consideradas críticas y aportan valor al correcto funcionamiento de la plataforma.

2.3.2. Descripción de productos

A continuación, se describe de manera concreta los productos que se hacen entrega como parte de la solución planteada con base en el análisis previo de la plataforma DTT y la arquitectura de pruebas previamente implementada.

2.3.2.1. Pruebas automatizadas para las principales funciones de la Plataforma DTT

Se dio continuidad al desarrollo de pruebas automáticas haciendo uso del ambiente de pruebas implementado en el ambiente de desarrollo. Para la implementación de estas pruebas se utilizará con IntelliJ Idea como IDE de programación y ejecución de las pruebas automáticas. Configurando esta herramienta con las diferentes librerías a utilizar por medio de Maven, así como se configura una herramienta para el desarrollo de pruebas los *frameworks* de Python, Jinja2, TestNG, RestAssured, entre otras dependencias necesarias para el funcionamiento de las pruebas automáticas.

Para poder gestionar el código de las pruebas desarrolladas se trabajará con GitLab para poder gestionar el repositorio de versionamiento donde se almacenará el código fuente del proyecto de pruebas automáticas. También se debe mencionar que en este contenedor se harán uso de los JenkinsFile, los cuales son archivos utilizados por la herramienta Jenkins para la ejecución de las pruebas automáticas, siempre sincronizados a su correspondiente repositorio de versionamiento.

2.3.2.2. Continuación de la administración y configuración de la arquitectura de pruebas

En el ambiente de automatización se debe continuar con la administración y configuración de Jenkins que permitirá ejecutar las pruebas desarrolladas en el ambiente de desarrollo y notificará los resultados, por medio de reportes, de cada prueba automática que se ejecute, pudiendo visualizar los escenarios que fueron aprobados o bien los que fallaron.

Para continuar con su desarrollo y mantenimiento desde otros equipos y sistema operativo se hará uso de los DockerFile, los cuales son archivos que permiten definir contenedores, y estarán almacenados en un repositorio en GitLab para su versionamiento (Docker Overview, s.f.).

2.4. Costos del proyecto

Para la elaboración de un presupuesto primero se hizo un detalle de los recursos humanos y materiales que se ocuparon durante el transcurso del proyecto.

2.4.1. Recursos Humanos

A continuación, se detallan los principales actores que se requieren para la ejecución del proyecto y el tiempo de estos profesionales es cuantificado en el presupuesto del proyecto.

Tabla VII. **Recursos Humanos**

Recursos	Descripción
Asesores	Un (01) asesor de EPS de la Escuela de Ingeniería en Ciencias y Sistemas – Ing. William Estuardo Escobar Argueta.
	Un (01) Asesor Supervisor de la Institución (Escuela de Ingeniería en Ciencias y Sistemas) – Ing. Miguel Marín De León.
Desarrollo e implementación	Un (01) estudiante con Pensum Cerrado de la carrera de Ingeniería en Ciencias y Sistemas, para el análisis, diseño y desarrollo de la solución tecnológica del proyecto – Estudiante Jerson Eduardo Villatoro Nova.
	Personal que participa en el desarrollo del Proyecto de la Plataforma DTT de la Escuela de Ingeniería en Ciencias y Sistemas y sean asignados para la capacitación del proyecto a implementar. Por lo menos (02) personas

Fuente: elaboración propia.

2.4.2. Recursos Materiales

Para el desarrollo e implementación de la aplicación se hizo uso de los siguientes materiales e insumos, los cuales también son cuantificados en el presupuesto del proyecto.

Tabla VIII. **Recursos Materiales**

Recursos	Descripción
Mobiliario y Equipo	Una (01) computadora (Laptop) configurada con las herramientas requeridas para el desarrollo del proyecto. Cabe mencionar que las herramientas a instalar son de licenciamiento libre, el cual no genera un costo para el proyecto.
	Una (01) impresora multifuncional para la impresión de los diferentes manuales o documentos requeridos durante el desarrollo del proyecto.
	Un (01) escritorio de oficina y una (01) silla de oficina para el uso de la persona a desarrollar el proyecto.
	Dos (02) máquinas virtuales en la nube de <i>Digital Ocean</i> que alojan Jenkins, para la automatización de las pruebas, y la Plataforma DTT de pruebas.
Inmueble	Localidad (oficina de trabajo) donde se resguardarán los insumos, el mobiliario y equipo que se requiere para el desarrollo del proyecto, contando también con los servicios básicos de agua, luz, extracción de basura, servicio telefónico e internet.
Útiles de Oficina	Papel bond (hojas en blanco), engrapadora, grapas, sacabocado, gancho de folder, folder y demás material de oficina para los diferentes documentos requeridos en el desarrollo del proyecto.

Fuente: elaboración propia.

2.4.3. Presupuesto del proyecto

A continuación, se desglosa una estimación de los costos del proyecto que se ha desarrollado durante un período de tiempo de (06) meses. Se hace la división entre gastos iniciales, gastos durante la ejecución y se agrega un apartado para los gastos no planificados.

Tabla IX. Costos del proyecto

Gastos de Inicio			
Recursos	Cantidad	Costo Unitario (Q)	Subtotal (Q)
Mobiliario y Equipo			
Laptop	01	7,000.00	7,000.00
Impresora	01	1,300.00	1,300.00
Escritorio de oficina	01	600.00	600.00
Silla de oficina	01	300.00	300.00
Subtotal (Q)			9,200.00
Gatos Mensuales durante le Ejecución del Proyecto – 06 meses -			
Recurso	Costo Mensual (Q)	Tiempo/Meses	Subtotal (Q)
Consumibles			
Agua	75.00	06	450.00
Energía eléctrica	200.00	06	1,200.00
Telefonía e Internet	350.00	06	2,100.00
Útiles de oficina	50.00	06	300.00
Servidor en la nube para alojar Jenkins	117.90	06	707.40
Servidor en la nube para alojar la Plataforma DTT de pruebas	196.50	06	1,179.00
Subtotal (Q)			5,936.40
Recurso Humano	*Horas Trabajo	Costo / Hora (Q)	Subtotal (Q)
Asesor EPS	48	200.00	9,600.00
Estudiante EPS	516	70.00	36,120.00
Subtotal (Q)			45,720.00
Gastos no planificados			
Detalle	Gastos Calculados (Q)	Estimación	Subtotal (Q)
Se estima un porcentaje en gastos imprevistos o no planificados.	60,856.40	5%	3,042.82
Subtotal (Q)			3,042.82
Total			Q 63,899.22
Noventa mil novecientos ochenta y nueve quetzales con veintidós centavos.			

Fuente: elaboración propia.

2.5. Beneficios del proyecto

La realización de este proyecto enfocado al control de calidad de la Plataforma DTT trajo consigo una serie de beneficios para los diferentes interesados en esta. Para un mejor despliegue de estos se dividen en dos categorías para un mejor desglose. Primero se encuentran los beneficios en el proceso de desarrollo y luego los beneficios a nivel de procesos administrativos.

Los beneficios en el proceso de desarrollo son aquellos que se encuentran relacionados con el desarrollo y crecimiento de la plataforma a nivel de programación, es decir, la programación de futuros módulos y le mejora o actualización de los ya implementados. Los beneficios a nivel de procesos administrativos son los que se encuentran relacionados con las mejoras a los diferentes procesos que tiene a su cargo la escuela y que a su vez involucran el uso de la Plataforma DTT.

2.5.1. Beneficios en el proceso de desarrollo

Se buscó reducir los tiempos para validar las principales funcionalidades de la Plataforma DTT esperando que estos se reduzcan a menos de la mitad del tiempo que ocupaba previo a la implementación de la arquitectura de QA. Anteriormente para la validación de las funcionalidades se realizaban pruebas a mano sobre las pantallas que se necesitaban comprobar, esto repercutía en el tiempo que se tomaba un desarrollador en entregar está nueva funcionalidad, validando además que las otras que se encuentran relacionadas sigan funcionando como se espera.

Además, para la realización de estas pruebas, dependiendo del rol que evaluaban, se necesitan de los permisos del administrador de la plataforma, lo

que también entorpecía el proceso al hacer necesario que tanto el equipo de desarrollo como el coordinador de la plataforma tuvieran que coordinar para realizar las pruebas.

Por tanto, con la automatización de estas pruebas actualmente se puede hacer uso de la arquitectura de QA para evaluar las pruebas de manera automatizada al tener acceso al contenedor de Jenkins que alberga las pruebas para que sean ejecutadas sobre la Plataforma DTT. Finalmente, toda esta implementación permite a largo plazo facilitar el ciclo de desarrollo de software del proyecto de la Plataforma DTT, lo que a su vez permitirá realizar implementaciones más periódicas y la reducción de riesgo de introducir errores en las funcionalidades ya implementadas.

2.5.2. Beneficios en el proceso control de calidad

Otro beneficio que vale la pena mencionar es la mejora del control de calidad en los productos o las nuevas funcionalidades a implementar, permitiendo que el desarrollo se enfoque más en la funcionalidad en sí y no invertir demasiado tiempo en realizar pruebas para validar que las otras funcionalidades se encuentren funcionando correctamente al implementar una nueva.

Es importante mencionar que este proyecto se enfocó en el desarrollo de pruebas para las funcionalidades que ya encuentran implementadas en la Plataforma DTT, pero que por carencia de una arquitectura de pruebas y una metodología de pruebas no se realizaron pruebas automatizadas en el proceso de desarrollo. Al tener ambas cosas en funcionamiento se capacitó a los nuevos desarrolladores para que durante su proceso de construcción de los nuevos módulos se tuviera en cuenta el desarrollo de las pruebas que corresponden a sus productos.

También se trabajó en un modelo de documentación, el cual se adjunta al anexo de este informe, una plantilla de documentación de pruebas con el fin de documentar las pruebas automatizadas lo más detallado posible para futuras referencias y cambios que se necesiten realizar en estas.

3. FASE ENSEÑANZA APRENDIZAJE

Se propone un plan de capacitación para el uso de las pruebas, el ambiente de desarrollo y los diferentes materiales de apoyo elaborados con el fin de dar continuidad a la construcción de la arquitectura de QA.

3.1. Capacitación propuesta

Para la capacitación de los interesados, tanto desarrolladores de la Plataforma DTT como del administrador de esta, se ha propuesto el dividir la misma en al menos tres segmentos, cada cual puede tomarse como una sesión de entre 45 a 60 minutos, para explicar los detalles del proyecto. Las sesiones mínimas propuestas se detallan a continuación.

3.1.1. Introducción al ambiente de desarrollo y el ambiente de QA

En esta sesión de capacitación se hace explicación de cómo se encuentran montados los ambientes que se ocupan para la elaboración de pruebas como para su respectivo despliegue. El ambiente de desarrollo se encuentra montado de manera local haciendo uso de Docker para levantar el contenedor de Jenkins y otro contenedor con la Plataforma DTT.

También se expondrán de manera detallada las herramientas que se ocuparán para el desarrollo y las diferentes partes que conforman el patrón que se ha ocupado para el desarrollo de pruebas previas. Cabe aclarar que el patrón solo es una sugerencia para mantener una estructura en el desarrollo del

proyecto, pero este se puede adaptar a las necesidades que se encuentren en los módulos a los que se les desarrollará pruebas posteriormente.

Como material de apoyo se dará a los desarrolladores y al administrador de la plataforma, una guía de desarrollo de pruebas como referencia. La misma se detalla en el apartado de Material Elaborado de este informe.

3.1.2. Práctica de Desarrollo de Pruebas

Durante esta sesión de la capacitación se tiene como objetivo la elaboración, a manera de ejemplo, de una prueba a nivel de programación. Se pretende explicar a detalle las clases que se sugieren utilizar como base para su desarrollo. En general, en el desarrollo de pruebas se realizan dos clases para iniciar la construcción de la prueba. Dichas clases son una clase de definición y una clase de acciones. La primera identifica en la interfaz gráfica los elementos HTML con los que se va a interactuar y la segunda se encarga de definir las acciones que se realizarán sobre ellos.

Luego de detallar esas clases y sus funciones, se procede a desarrollar la clase que se encarga de la lógica de la prueba, es decir que toma las acciones y las realiza en el orden lógico que se requiera. Para esta parte, se debe indicar a los desarrolladores que es importante los reportes en consola que se dejan a manera de muestra para indicar el progreso de la prueba.

En resumen, el fin de esta sesión es explicar a los interesados el proceso de desarrollo, a nivel de programación, acompañado de un ejemplo y la guía de desarrollo que se deja para los futuros desarrolladores de la Plataforma DTT.

3.1.3. Despliegue de Pruebas en Ambiente de QA

Como sesión final se recomienda la explicación del proceso de despliegue de la prueba elaborada en la sesión anterior. Para ellos primero se hace una exposición general del Jenkins en el cual se han desplegado las pruebas durante el desarrollo del proyecto. Se explican los detalles del archivo que se usa para el despliegue y los pasos a seguir para crear y ejecutar un *job* en Jenkins.

Es importante ahondar en el orden que se ha llevado al momento de subir una prueba al ambiente de QA, con el fin de mantener el proyecto lo más entendible. Las pruebas se han agrupado en función de los menús principales de cada rol para la facilidad de su ubicación.

3.2. Materiales Elaborados

En el transcurso de la ejecución del proyecto se han generado tres documentos para uso del administrador de la plataforma como para los interesados o futuros desarrolladores de la Plataforma DTT. A continuación, se detallan dichos documentos.

3.2.1. Manual de Usuario para Jenkins

Como material de apoyo principalmente para el administrador de la Plataforma DTT y para los desarrolladores del área de QA. El fin de este documento es la explicación paso a paso del uso de la plataforma Jenkins como usuario final del producto.

En este documento se detallan los pasos necesarios para usar el contenedor de Jenkins. Va desde el levantar el contenedor usando la consola, el

visionado de las pruebas subidas a Jenkins, su ejecución y visualización de su reporte en consola. A manera de continuar con el desarrollo de la plataforma de QA se dejó una copia editable de dicho manual para futuros cambios en la organización de las pruebas o bien en el caso de una actualización del orquestador Jenkins.

3.2.2. Plantilla para Control de Pruebas

Como parte de la documentación entregada se trabajó en una plantilla que llevará la información respectiva de cada prueba desarrollada. Esta plantilla se llenó con la información de las pruebas que se realizaron durante el proyecto y se dejó como parte de los productos entregados.

En el contenido de la plantilla figuran los siguientes datos:

- Detalles del desarrollo como el título de la prueba, correlativo, nombre del desarrollador y correo de contacto.
- Detalles de la funcionalidad conformados por una descripción breve de la funcionalidad, enlace a la pantalla a la cual se la realiza la prueba, consideraciones u observaciones sobre dicha prueba, dentro de este último punto se contemplan detalles como descripción de archivos de entrada, compatibilidad con navegadores o cualquier detalle relacionado.
- Descripción paso a paso de la prueba. Este apartado muestra una lista de los pasos que debe mostrar la prueba en el reporte de consola en caso de que se haberse ejecutado de manera exitosa.

- Historial de Cambios. Este apartado de la plantilla se incluye para detallar los cambios que se realicen sobre el código de la prueba realizados por los usuarios del proyecto.

3.2.3. Guía de Desarrollo de Pruebas

Esta guía fue elaborada con el fin de dejar un documento que explicará paso a paso cómo elaborar una prueba. Los pasos que se explican en ella tienen el fin de mantener el patrón de diseño que se ha ocupado desde el inicio del proyecto. Las fases que se explican en ella son las siguientes:

- Identificación de la funcionalidad objetivo.
- Identificación de los elementos con los cuales habrá interacción.
- Encontrar identificadores de éxito en la prueba, es decir elementos que indiquen que la prueba se desarrolla con éxito.
- Definición de las interacciones.
- Definición del orden lógico en el cual se realizarán las acciones de la prueba.

Como parte final de la guía se incluyen una serie de sugerencias y comentarios respecto al desarrollo de pruebas que se deben tomar en cuenta para que quien haga uso del proyecto posteriormente pueda ahorrar tiempo y reducir la curva de aprendizaje de las herramientas que usará.

4. PRESENTACIÓN DE RESULTADOS

4.1. Cuestionario a los interesados

Con el fin de obtener retroalimentación para la continuación del proyecto en próximas iteraciones y a su vez evaluar los resultados obtenidos en el transcurso de esta se ha realizado una encuesta al administrador de la Plataforma DTT, ing. Miguel Marín De León.

Figura 4. **Primera parte, cuestionario al administrador de la Plataforma DTT**



CUESTIONARIO AL ADMINISTRADOR DE LA PLATAFORMA DTT

El siguiente cuestionario tiene como objetivo el evaluar el desarrollo del EPS, sus productos y recibir retroalimentación para la mejora continua de la arquitectura de QA implementada.

- Las pruebas realizadas en la ejecución de este proyecto son sobre funcionalidades ya implementadas, ¿cómo considera que impactará en la implementación de futuras funcionalidades?

R/ Si se realizaron sobre funcionalidades existentes, y el impacto que se tendrá es el tiempo de realizar las pruebas ya que antes se tomaba días para realizar las pruebas de forma manual haciendo uso de usuarios dentro del sistema y en ocasiones aparecían errores de funcionalidad hasta el próximo semestre.
- Podría dar un estimado de a cuántas personas se beneficiarán con el desarrollo continuo de la arquitectura de QA.

R/ Diez por año próximamente en los proyectos de desarrollo en DTT.
- ¿Qué recomendación haría para dar continuidad al crecimiento y mantenimiento de la arquitectura de QA trabajada en este proyecto?

R/ Actualmente se implementó un equipo de desarrollo de pruebas automatizadas, por semestre, dentro del proyecto de prácticas, para que este proyecto sea continuo en los nuevos módulos que se adhieren.
- Desde el punto de vista del Administrador de la Plataforma DTT, ¿cómo mejora el proceso de implementación de nuevas funcionalidades respecto de como se hacía antes de implementar la arquitectura de QA?

R/ Como administrador me da certeza rápida que, a la hora de incorporar una nueva funcionalidad, no se dejen problemas de funcionamiento en módulos ya existentes con pruebas automatizadas.

Fuente: elaboración propia.

Figura 5. **Segunda parte, cuestionario al administrador de la Plataforma DTT**



Antes había que probarlo durante el semestre y si había error, se corregía cuando se detectaba el error.

5. ¿En la siguiente iteración del proyecto qué módulos o roles considera que se deben abarcar y por qué?

R/ Dentro de DTT, ya se cuentan con nuevos módulos a los cuales ya se está proyectando realizar pruebas automatizadas, como lo son:
Módulo de firma Digital de documentos de prácticas.
Y un nuevo Módulo documentos de Graduación entre Lingüística y Dirección ECYS
Estos módulos son críticos que estén funcionando cuando se agreguen nuevas funcionalidades.

6. ¿En general, tiene algún comentario u observación acerca del impacto que tendrán los productos dejados tras la ejecución de este proyecto?

R/ Uno de los impactos es:

- a) Verificación de errores en tiempos rápidos.
- b) Corrida de pruebas de forma rápida y automática, sin intervención de usuarios físicos.
- c) Mejor garantía que las funcionalidades actuales no sean afectadas por las nuevas, en forma inmediata, para que sea corregida a tiempo por el responsable del proyecto que afecte algún módulo existente(antiguo).

Fuente: elaboración propia.

Las respuestas reflejan que es el impacto que se tiene sobre la plataforma la implementación y continuo desarrollo de la plataforma de QA es de gran importancia para el continuo crecimiento de la plataforma. Como se puede leer en la pregunta 4, la manera en que los errores se detectaban y solucionaban no es la óptima. Por lo que el contar con un sistema que automatice la detección de errores permitirá avanzar en la implementación de nuevas funcionalidades.

También se debe resaltar que en la pregunta 6, se hace mención a que se proyecta empezar con el desarrollo de pruebas automáticas a funcionalidades nuevas, que son el módulo de firma digital y el módulo de documentos de graduación.

4.2. Inventario de prueba finalizadas

Se tabularon las pruebas finalizadas a lo largo de la ejecución del proyecto para dejar un registro que pueda ser consultado por quienes den continuidad a la construcción y mantenimiento de la arquitectura de QA. Se ha dividido en función de los roles para mayor orden.

En las tablas se muestra si la prueba desarrollada funciona con el navegador Chrome, si funciona con el navegador Firefox y en la columna de comentario se aclara cualquier condición a tomar en cuenta para su ejecución. Las columnas C y F indican si la prueba funciona correctamente en el navegador Chrome y Firefox respectivamente.

4.2.1. Pruebas del rol administrador

Estás pruebas abarcan las funcionalidades que propia del administrador en las diferentes pantallas a las que tiene acceso.

Tabla X. **Funcionalidades del rol Administrador con pruebas automáticas implementadas**

No.	Nombre	FC	FF	Observaciones
1	CRUD de usuarios	SI	SI	
2	CRUD de membresías (roles)	SI	SI	

Continuación de la tabla X.

No.	Nombre	FC	FF	Observaciones
3	Cargar archivos	SI	SI	Usa un archivo de entrada para probar la carga
4	Gestión de membresías	SI	SI	Desde la misma pantalla de usuario asigna un rol
5	Envío de emails	SI	SI	
6	CRUD de proyectos	SI	SI	
7	Asignación de proyectos (asignación y cambios en la asignación)	SI	SI	
8	CRUD de reportes de proyecto	SI	SI	
9	CRUD de catedráticos	SI	SI	
10	CRUD de estudiantes	SI	SI	
11	Gestión de fotos de estudiantes	Sí	No	- No funciona en Firefox - En la misma pantalla de estudiante se valida la fotografía
12	Reporte de notas	SI	SI	
13	Reporte de asistencia	SI	SI	
14	Reporte de asistencia de hoy	SI	SI	
15	Reporte de huellas registradas	SI	SI	
16	Reporte de asignación de horarios	SI	SI	
17	Reporte de Asistencia Virtual Hoy	SI	SI	
18	CRUD días de asueto	NO	SI	La mayoría de las veces no funciona con navegador Chrome
19	Reporte de resoluciones	SI	SI	

Continuación de la tabla X.

No.	Nombre	FC	FF	Observaciones
20	Aprobación o rechazo de resolución de permisos por falta	SI	NO	Se hará desde la pantalla del reporte No funciona en navegador Firefox
21	Creación de horario por día	NO	SI	No funciona con navegador Chrome
22	Asignación de horario a tutor	SI	SI	
23	Inicialización de semestre	SI	SI	Solo valida la correcta carga de la página
24	Creación y eliminación de excepciones de horario	SI	SI	
25	Configuración de capturas de huella	NO	SI	No funciona con navegador Chrome
26	CRUD de parámetros de penalización de asistencia	NO	SI	No funciona con navegador Chrome
27	Configuración de parámetros del sistema	SI	SI	
28	Carga de horarios	SI	SI	
29	Reporte de horarios por semestre	SI	SI	
30	Reporte de practicantes finales (resumen)	SI	SI	
31	Generación de certificado de período de período completo para practicantes finales	SI	NO	No se pudo detectar que se abre el reporte No funciona en Firefox

Fuente: elaboración propia.

4.2.2. Pruebas del rol Catedrático

Las pruebas para el rol Catedrático evalúan el correcto funcionamiento de las funcionalidades más usadas por los usuarios que cuentan con dicho rol. Entre ellas está la creación de ponderación de notas de un curso, calificar reportes de tutores académicos y la visualización de reportes.

Tabla XI. **Funcionalidades del rol Catedrático con pruebas automáticas implementadas**

COD	Nombre	FC	FF	Observaciones
1	Creación de Ponderación	SI	SI	Se debe usar con un usuario que tenga cargado al menos un curso
2	Reporte de Carga de Notas	SI	SI	Se debe usar con un usuario que tenga cargado al menos un curso
3	Reporte de Revalidación de Laboratorio	SI	SI	Se debe usar con un usuario que tenga cargado al menos un curso
4	Reporte de Revalidación de Gestión de Equivalencia	SI	SI	Se debe usar con un usuario que tenga cargado al menos un curso
5	Envío de mensajes a estudiantes	SI	NO	-Se debe usar con un usuario que tenga cargado al menos un curso -Presenta errores al ejecutarse en Firefox
6	Calificación de reportes a tutor académico	SI	SI	

Fuente: elaboración propia.

4.2.3. Pruebas del rol Tutor Académico

Las pruebas del rol de Tutor Académico cubren varias funcionalidades que son usadas durante el desempeño de sus funciones. Al cumplir como encargados de laboratorio de curso en muchos casos las acciones que pueden realizar sobre dicho laboratorio son parecidas a las que puede realizar un catedrático sobre el curso.

Tabla XII. **Funcionalidades del rol Tutor Académico con pruebas automáticas implementadas**

No.	Nombre	FC	FF	Observaciones
1	Carga Estudiantes	SI	SI	-Se debe usar un usuario con proyecto asignado -Si el listado ya ha sido cargado la prueba no fallará, pero se imprimirán los errores de la aplicación
2	Crear actividades de laboratorio	SI	SI	-Se debe usar un usuario con proyecto asignado
3	Carga de Notas Masiva por Actividad	SI	SI	-Se debe usar un usuario con proyecto asignado. -Debe haber una actividad a la cual se le puedan editar notas
4	Envío de mensajes	SI	SI	-Se debe usar un usuario con proyecto asignado
5	Enviar reporte a calificación	SI	SI	-Por los elementos web que se usan el controlador no puede interactuar por lo que solo se valida que se puede acceder a la edición

Fuente: elaboración propia.

4.2.4. Pruebas del rol Estudiante

Estás pruebas están orientadas a las funcionalidades que ocupa el usuario con rol Estudiante, el cual en su mayoría están orientas a visualizar información en diferentes pantallas de la Plataforma DTT.

Tabla XIII. **Pruebas del rol Estudiante con pruebas automatizadas implementadas**

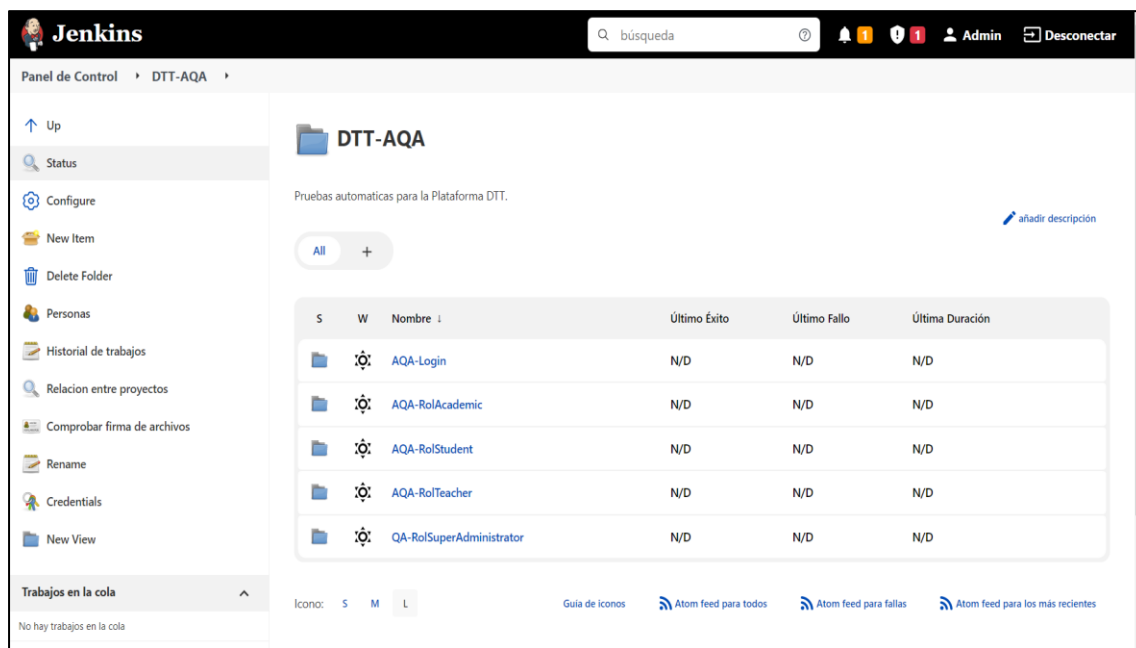
No.	Nombre	FC	FF	Observaciones
1	Visualización de información de proyecto asignado para prácticas finales	SI	SI	-Se debe usar un usuario con un proyecto asignado
2	Visualización de cursos asignados	SI	SI	
3	Generar Certificado de Práctica final	SI	SI	-Debe realizarse con un usuario sin proyectos activos
4	Visualización del detalle de notas de curso y laboratorio	SI	SI	
5	Lectura de mensaje recibidos por el tutor académico o catedrático	SI	SI	
6	Visualización de Estadísticas de Curso	SI	SI	
7	Visualización de los datos del perfil	SI	NO	-No funciona en navegador Firefox
8	Visualización del reporte de clase	SI	SI	-Se debe realizar con un estudiante asignado a un curso con laboratorio y clase con ponderación creada
9	Visualización del reporte de laboratorio	SI	SI	Se debe realizar con un estudiante asignado a un curso con laboratorio y clase con ponderación creada

Fuente: elaboración propia.

4.3. Pruebas subidas al servidor de automatización en la nube

Todas las pruebas listadas anteriormente han sido subidas a un servidor de automatización que se encuentra alojado en la nube bajo la administración del encargado de la Plataforma DTT. Dicho servidor cuenta con las características óptimas para ejecutar las pruebas en cualquier momento.

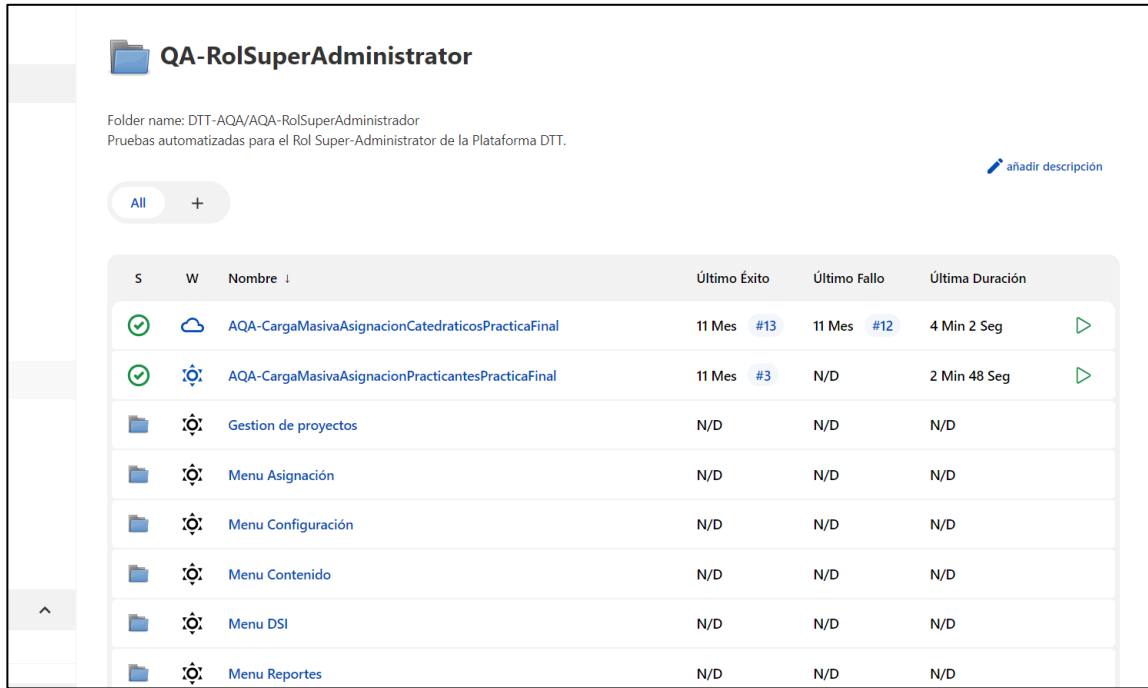
Figura 6. Pantalla principal del servidor de automatización en la nube



Fuente: [Fotografía de Jerson Eduardo Villatoro Nova]. (Captura de pantalla, 2022). Colección particular. Guatemala.

Para mantener un orden dentro del servidor las pruebas se han separado de acuerdo con los roles que se han abarcado durante el proyecto, a excepción de la prueba de Login que se encuentra en una carpeta aparte ya que está es la base para la ejecución de todas las demás pruebas.

Figura 7. Carpeta de las pruebas del rol Administrador



QA-RolSuperAdministrator

Folder name: DTT-AQA/AQA-RolSuperAdministrator
Pruebas automatizadas para el Rol Super-Administrator de la Plataforma DTT.

[añadir descripción](#)

All +

S	W	Nombre ↓	Último Éxito	Último Fallo	Última Duración
✓	☁	AQA-CargaMasivaAsignacionCatedraticosPracticaFinal	11 Mes #13	11 Mes #12	4 Min 2 Seg
✓	⚙	AQA-CargaMasivaAsignacionPracticantesPracticaFinal	11 Mes #3	N/D	2 Min 48 Seg
📁	⚙	Gestion de proyectos	N/D	N/D	N/D
📁	⚙	Menu Asignación	N/D	N/D	N/D
📁	⚙	Menu Configuración	N/D	N/D	N/D
📁	⚙	Menu Contenido	N/D	N/D	N/D
📁	⚙	Menu DSI	N/D	N/D	N/D
📁	⚙	Menu Reportes	N/D	N/D	N/D

Fuente: [Fotografía de Jerson Eduardo Villatoro Nova]. (Captura de pantalla, 2022). Colección particular. Guatemala.

Las pruebas dentro de cada rol fueron agrupadas por menús según estos se fueron abarcando a lo largo del proyecto. En la pantalla del rol Administrador se puede observar que se realizaron pruebas para los siguientes menús: gestión de proyectos, asignación, configuración, contenido, DSI y reportes.

Figura 8. Pantalla de arranque de ejecución de prueba

Pipeline AQA-CRUD_ReportesPracticantes

Esta ejecución requiere parámetros adicionales:

ENVIRONMENT
qa ▾

BROWSER
CHROME ▾

DTT_URL
Url de la Plataforma DTT para validar
https://104.248.57.198/

CREDENTIALS_ID
d813a722-8407-412b-b50b-b5ef506ad0bb ▾

☐ **ENVIAR_EMAIL**
Selecciona para enviar email al finalizar la prueba

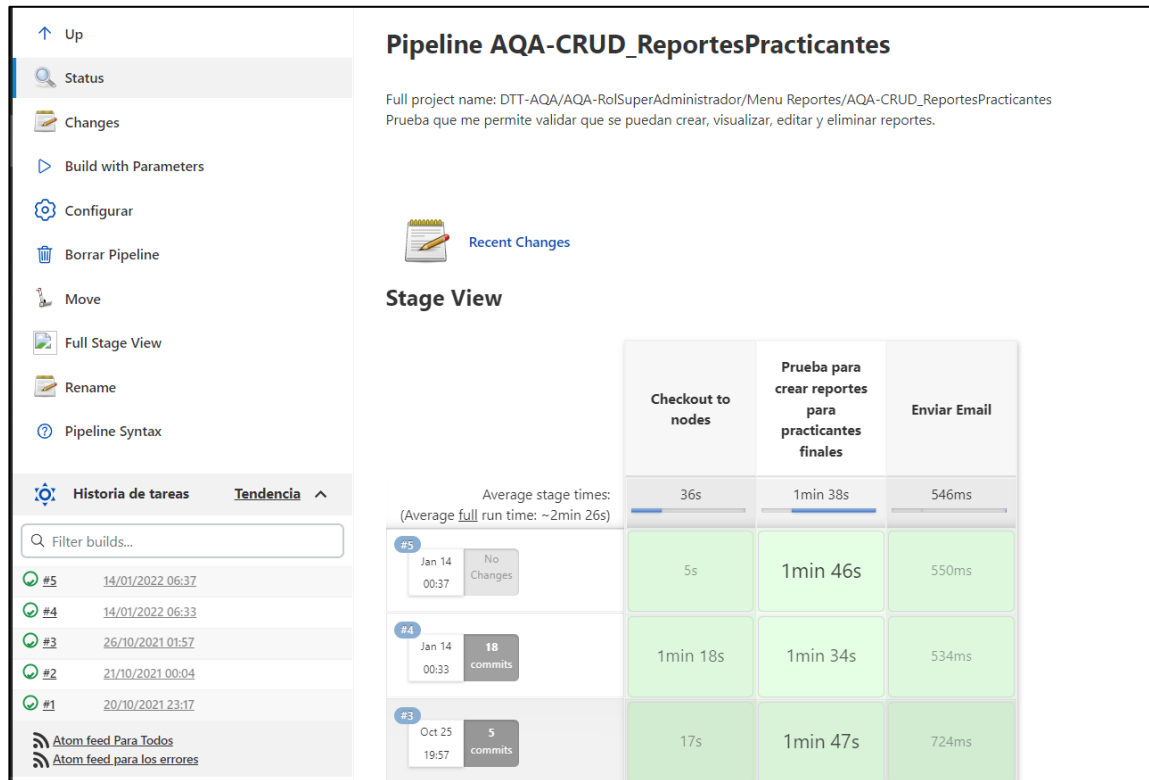
EMAIL_TO
Email para notificación al finalizar la prueba

Ejecución

Fuente: [Fotografía de Jerson Eduardo Villatoro Nova]. (Captura de pantalla, 2022). Colección particular. Guatemala.

Para la ejecución de las pruebas se espera que el usuario ingrese parámetros como el ambiente, que en el caso de todas las pruebas desarrolladas durante este proyecto son en el ambiente de QA, el navegador que se quiere usar para la prueba, la URL para la Plataforma DTT, que en el caso de las pruebas desarrolladas se usa el URL del DTT de QA que se encuentra en la nube y por último si se desea recibir un correo al finalizar la prueba con un enlace para ver los resultados y la dirección de correo electrónico a la cual se mandará dicho enlace.

Figura 9. Pantalla de ejecución de prueba



Fuente: [Fotografía de Jerson Eduardo Villatoro Nova]. (Captura de pantalla, 2022). Colección particular. Guatemala.

Figura 10. **Reporte en la consola del servidor de automatización de pruebas**

```
=====
##### Test: Prueba: CRUD de Reportes para practicantes en menú Reportes
##### Env: qa
##### Browser: FIREFOX / SetHeadless: true
##### DttUrl: https://104.248.57.198/
=====
=>> Inicio de la prueba de CRUD de Reportes de Practicantes
=>> Accediendo a la página de Reportes de Practicantes del rol Administrador
=>> Selecciona la opción Reportes en el menú principal
=>> Selecciona la opción Reportes de Practicantes en el menú Reportes
=>> Selecciona la opción Reportes y Restricciones en el menú Reportes de Practicantes
=>> Recuperando el título de la página de Reportes
=>> Reportes.GetTitle: Restricciones de reporte
=>> Click en el botón Añadir Registro
=>> Llenando formulario para creación de Reporte para practicantes
=>> Seleccionando el checkbox Activo
=>> Seleccionando el checkbox Es Final
=>> Ingresando fecha de finalización del reporte
=>> Ingresando fecha de inicio del reportes
=>> Click al botón continuar para guardar el Reporte
=>> Eliminando reporte creado
=>> Click a la etiqueta Fecha Inicio para ordenar
=>> Click al botón eliminar
====>>>Done: Prueba de CRUD de Reportes de Practicantes finalizada con éxito
=>> DriverBaseClass.closeDriver
=>> Closing the driver...
1642142345102 Marionette INFO Stopped listening on port 46653
JavaScript error: resource://gre/modules/osfile/osfile_async_front.jsm, line 426: Error: OS.File has been shut down. Rejecting post to stat
=>> Driver closed
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 63.851 sec - in TestSuite

Results :

Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
```

Fuente: [Fotografía de Jerson Eduardo Villatoro Nova]. (Captura de pantalla, 2022). Colección particular. Guatemala.

Para cada prueba desarrollada, se imprime un reporte en consola que muestra paso a paso el avance de la prueba, de manera tal que, si hay un error en el transcurso de su ejecución, el usuario pueda identificar el paso en el que ha fallado. También se ha dejado documentado cada una de la prueba para que haya una referencia de una ejecución terminada con éxito.

4.4. Documentación de pruebas

Cada una de las pruebas desarrolladas es acompañada por un documento que registra datos que se consideran importantes para futuras referencias. Entre ellos un correlativo, el nombre, la funcionalidad y una descripción de esta, los pasos que muestra la prueba al ser ejecutada, observaciones y comentarios del desarrollados y por último un historial de desarrolladores que han modificado el código fuente de la prueba en el cual se incluye un correo de contacto.

Figura 11. Documentación de prueba






CONTROL DE AUTOMATIZACIÓN DE PRUEBAS

1. Desarrollo

Código: A-04

Título	Gestión de Membresías
Desarrollador	Jerson Villatoro
Correo	jerduar@gmail.com
Enlace	https://104.248.57.198/cpfecys/admin/users

2. Funcionalidad

Descripción de la Funcionalidad

Agrega o quita membresías a un usuario en específico desde el rol de administrador.

Consideraciones u observaciones

Esta funcionalidad se valida desde la misma pantalla de usuarios.

3. Pasos de la Prueba

1. Select Usuarios option in the principal menu
2. Select Usuarios option from Usuarios Menú
3. Click sobre la opción de membresías del primer elemento de la tabla
4. Click the Añadir Registro Button in the User Page
5. Seleccionando membresías para el usuario
6. Click the Añadir Registro Button in the User Page
7. Click en el botón eliminar

4. Historial de Cambios

No.	Nombre	Correo	Fecha
1	Jerson Villatoro	jerduar@gmail.com	01/02/2022

Fuente: elaboración propia.

CONCLUSIONES

1. El desarrollo de pruebas automatizadas para las funcionalidades ya implementadas en la Plataforma DTT en los roles del Administrador, Catedrático, Tutor académico y Estudiante apoya a los desarrolladores de la plataforma a implementar nuevos módulos asegurando que las funcionalidades con las que se encuentren relacionadas continúen funcionando según lo esperado, lo que al ser la Plataforma una herramienta en constante uso y crecimiento es un aspecto que se debe tener en cuenta en los procesos de desarrollo de nuevas funcionalidades.
2. Al usar Jenkins como servidor de automatización combinado con la tecnología de la nube, se facilita al Administrador y a cualquier interesado en el desarrollo de nuevos módulos para la Plataforma DTT una herramienta que se encuentra disponible en cualquier momento, dentro de este servidor queda registrado todas las pruebas que se han ejecutado sobre diferentes vistas y funcionalidades de la plataforma, y muestra al usuario de una manera legible su progreso, además se configurado las pruebas para que muestren paso a paso las interacciones que se realizan con los elementos de la plataforma de forma que se pueda detectar cualquier falla en la funcionalidad a evaluar.
3. Se consideró que quienes hagan uso de este servidor de automatización y las pruebas que en él se encuentran pueda recibir de una forma más personal información con los resultados de la pruebas ejecutadas, por lo que entre los parámetros de ejecución de las pruebas se deja lugar para recibir un correo electrónico con el cual el usuario puede obtener un enlace

para visualizar la prueba ejecutada paso a paso y confirmar si la misma se ha realizado con éxito o ha fallado en el proceso de ejecución de tal manera que sea posible determinar en dónde y la o las posibles razones que causaron dicho fallo.

4. El continuo trabajo en la arquitectura de pruebas, tanto el desarrollo de pruebas como la configuración de automatización en el servidor de automatización en la nube es un factor primordial que beneficia de manera directa el proceso de desarrollo de nuevos módulos en la Plataforma DTT, facilita el mantenimiento general de la plataforma y el aseguramiento de la calidad en todas las funcionalidades ya implementadas, al final asegura al administrador y a los equipos de desarrollo que los cambios que se realicen no afecten la consistencia y el correcto funcionamiento de la plataforma.

RECOMENDACIONES

1. Finalizar el desarrollo de pruebas automatizadas para las funcionalidades faltantes, ya que debido a la gran cantidad de necesidades que cubre la Plataforma DTT, sobre todo el rol del Administrador, no permitió que se logaran calendarizar todas las pruebas en el tiempo que abarcó este proyecto, por lo que es importante que se asigne una continuación a este trabajo que permita finalizar todas las funcionalidades faltantes que por el factor de tiempo y priorización quedaron fuera de esta iteración.
2. Analizar la formación de un equipo permanente para continuar con el trabajo de mantenimiento de la arquitecta de QA, el desarrollo de nuevas pruebas para las nuevas funcionalidades y su respectiva automatización. Este equipo se recomienda que sea asignado únicamente para estas tareas, ya que si esta tarea es asignada a equipos de desarrollo puede restar tiempo de desarrollo y también se corre el riesgo de desarrollar pruebas sesgadas, al ser los mismos desarrolladores de pruebas como de módulos.
3. Implementar el uso de la plantilla de documentación de pruebas y su continua mejora para atender las necesidades del proyecto DTT manteniendo un estándar en el registro de las pruebas y un historial de los desarrolladores que se involucren con el código y configuración de las pruebas automatizadas.
4. Continuar utilizando el patrón de diseño de las pruebas automatizadas que se dejan en los entregables del proyecto. Dicho patrón fue seguido

durante la ejecución de este proyecto, ya que resultaba conveniente para mantener un código ordenado y legible.

REFERENCIAS

1. About us (2021). *Plataforma DTT-ECYS*. Recuperado de https://dttecys.org/about_us.
2. Docker Overview (s.f.). *Docker docs*. Recuperado de <https://docs.docker.com/get-started/overview/>.
3. Prepare for testing (2021). *IntelliJ IDEA Documentation*. Recuperado de <https://www.jetbrains.com/help/idea/2022.1/testing.html>.
4. Jenkins User Documentation (s.f.). *User Documentation Home*. Recuperado de <https://www.jenkins.io/doc/>.
5. The Selenium Browser Automation Project (s.f.). *Selenium Documentation*. Recuperado de <https://www.selenium.dev/documentation/>.

APÉNDICES

Se muestran a continuación varios documentos que han sido generados para el uso del proyecto.

Apéndice 1. Cronograma de actividades de configuración de ambiente de desarrollo

No.	Actividades	Cantidad días	FechaInicio	Fecha Fin
1	Configuración de arquitectura de pruebas	7		
1.01	Configurar repositorios de versionamiento para proyecto de pruebas y archivos de configuración de infraestructura a implementar	1	05/08/2021	05/08/2021
1.02	Instalación de Docker y configuración de contenedor en el que se encuentra el entorno de desarrollo de pruebas	2	06/08/2021	09/08/2021
1.03	Instalación y configuración del contenedor en el que se encuentra la herramienta de automatización de pruebas, Jenkins	2	10/08/2021	11/08/2021
1.04	Realizar pruebas de funcionamiento de infraestructura con pruebas básicas	1	12/08/2021	12/08/2021
1.05	Realizar pruebas de funcionamiento de infraestructura conectándose a la plataforma DTT	1	13/08/2021	13/08/2021

Fuente: elaboración propia.

Apéndice 2. Cronograma de pruebas para el rol administrador

2	Desarrollo de pruebas del rol Administrador	80		
2.01	Prueba: CRUD de usuarios	4	17/08/2021	20/08/2021
2.02	Prueba: CRUD de membresías (roles)	3	23/08/2021	25/08/2021
2.03	Prueba: CRUD de eventos	3	26/08/2021	30/08/2021
2.04	Prueba: gestión de membresías	2	31/08/2021	01/09/2021
2.05	Envío de emails y gestión de archivos	3	02/09/2021	06/09/2021
2.06	Prueba: CRUD de proyectos	4	07/09/2021	10/09/2021
2.07	Prueba: gestión de proyectos (asignación y cambios en la asignación)	2	13/09/2021	14/09/2021
2.08	Prueba: CRUD de reportes de proyecto	3	15/09/2021	17/09/2021
2.09	Prueba: Gestión de catedráticos	4	20/09/2021	23/09/2021
2.10	Prueba: CRUD de estudiantes	4	24/09/2021	29/09/2021
2.11	Prueba: gestión de fotos de estudiantes	2	30/09/2021	01/10/2021
2.12	Prueba: gestión de roles (modificar/agregar rol en usuarios)	3	04/10/2021	06/10/2021
2.13	Prueba: Reporte de notas	2	07/10/2021	08/10/2021

Fuente: elaboración propia.

Apéndice 3. Segunda parte del cronograma del rol Administrador

2.14	Prueba: Reporte de asistencia	2	11/10/2021	12/10/2021
2.15	Prueba: Reporte de asistencia de hoy	2	13/10/2021	14/10/2021
2.16	Prueba: Reporte de huellas registradas	2	15/10/2021	18/10/2021
2.17	Prueba: Reporte de asignación de horarios	2	19/10/2021	21/10/2021
2.18	Prueba: Aprobación o rechazo de solicitud de falta	2	22/10/2021	25/10/2021
2.19	Prueba: CRUD días de asueto	2	26/10/2021	27/10/2021
2.20	Prueba: Reporte de resoluciones	2	28/10/2021	29/10/2021
2.21	Prueba: Aprobación o rechazo de resolución de permisos por falta	2	02/11/2021	03/11/2021
2.22	Prueba: CRUD días de asueto	2	04/11/2021	05/11/2021
2.23	Prueba: CRUD de horario por día	2	08/11/2021	09/11/2021
2.24	Prueba: Asignación de horario a tutor	2	10/11/2021	11/11/2021
2.25	Prueba: Inicialización de semestre	3	12/11/2021	16/11/2021
2.26	Prueba: Creación y eliminación de excepciones de horario	2	17/11/2021	18/11/2021
2.27	Prueba: Configuración de capturas de huella	2	19/11/2021	22/11/2021
2.28	Prueba: CRUD de parámetros de penalización de asistencia	2	23/11/2021	24/11/2021
2.29	Prueba: Configuración de parámetros del sistema	2	25/11/2021	26/11/2021
2.30	Prueba: Carga de horarios	3	29/11/2021	01/12/2021
2.31	Prueba: Reporte de horarios por semestre	2	02/12/2021	03/12/2021
2.32	Prueba: Reporte de practicantes finales	2	06/12/2021	07/12/2021
2.33	Prueba: Generación de certificado de período de período completo para practicantes finales	1	08/12/2021	08/12/2021

Fuente: elaboración propia.

Apéndice 4. Calendario de pruebas para los roles Catedrático y Tutor Académico

3	Desarrollo de pruebas del rol Teacher	14		
3.01	Prueba: CRUD de actividades del curso	3	09/12/2021	13/12/2021
3.02	Prueba: Reporte de carga de notas	3	14/12/2021	16/12/2021
3.03	Prueba: Reporte de gestión de revalidación de laboratorio	2	17/12/2021	20/12/2021
3.04	Prueba: Reporte de gestión de equivalencias de laboratorio	2	21/12/2021	22/12/2021
3.05	Prueba: Envío de mensaje a estudiantes	2	23/12/2021	27/12/2021
3.06	Prueba: Calificación de reportes del tutor académico	2	28/11/2021	29/12/2021
4	Desarrollo de pruebas del rol Academic	13		
4.01	Prueba: Carga de estudiantes	3	30/12/2021	04/01/2022
4.02	Prueba: CRUD de actividades del laboratorio	3	05/01/2022	07/01/2022
4.03	Prueba: Carga de notas	3	10/01/2022	12/01/2022
4.04	Prueba: Enviar mensaje a los estudiantes asignados al curso	2	13/01/2022	14/01/2022
4.05	Prueba: Enviar reporte de prácticas a calificar	2	17/01/2022	18/01/2022

Fuente: elaboración propia.

Apéndice 5. Calendario de pruebas del rol Estudiante

5	Desarrollo de pruebas del rol Student	15		
5.01	Prueba: Visualización de información de proyecto asignado para prácticas finales	2	19/01/2022	20/01/2022
5.02	Prueba: Visualización de cursos asignados	2	21/01/2022	24/01/2022
5.03	Prueba: Generar certificado de practica final	2	25/01/2022	26/01/2022
5.04	Prueba: Visualización del detalle de notas de curso y laboratorio	2	27/01/2022	28/01/2022
5.05	Prueba: Lectura de mensajes recibitos por el tutor o catedrático	1	01/02/2022	01/02/2022
5.06	Prueba: Visualización de estadísticas de curso	2	02/02/2022	03/02/2022
5.07	Prueba: Visualización del perfil	2	04/02/2022	07/02/2022
5.08	Prueba: Visualización de reporte de clase	1	08/02/2022	08/02/2022
5.09	Prueba: Visualización de reporte de laboratorio	1	09/02/2022	09/02/2022

Fuente: elaboración propia.

Apéndice 6. Formato de documentación de pruebas

 			
CONTROL DE AUTOMATIZACIÓN DE PRUEBAS			
1. Desarrollo			
Código:			
Título			
Desarrollador	Jerson Villatoro		
Correo	jerduar@gmail.com		
Enlace			
2. Funcionalidad			
Descripción de la Funcionalidad			
Consideraciones u observaciones			
3. Pasos de la Prueba			
4. Historial de Cambios			
No.	Nombre	Correo	Fecha
1	D		

Fuente: elaboración propia.

Apéndice 7. Presupuesto del proyecto

Gastos de Inicio			
Recursos	Cantidad	Costo Unitario (Q)	Subtotal (Q)
Mobiliario y Equipo			
Laptop	01	7,000.00	7,000.00
Impresora	01	1,300.00	1,300.00
Escritorio de oficina	01	600.00	600.00
Silla de oficina	01	300.00	300.00
Subtotal (Q)			9,200.00
Gatos Mensuales durante le Ejecución del Proyecto – 06 meses -			
Recurso	Costo Mensual (Q)	Tiempo/Meses	Subtotal (Q)
Consumibles			
Agua	75.00	06	450.00
Energía eléctrica	200.00	06	1,200.00
Telefonía e Internet	350.00	06	2,100.00
Útiles de oficina	50.00	06	300.00
Servidor en la nube para alojar Jenkins	117.90	06	707.40
Servidor en la nube para alojar la Plataforma DTT de pruebas	196.50	06	1,179.00
Subtotal (Q)			5,936.40
Recurso Humano	*Horas Trabajo	Costo / Hora (Q)	Subtotal (Q)
Asesor EPS	48	200.00	9,600.00
Estudiante EPS	516	70.00	36,120.00
Subtotal (Q)			45,720.00
Gastos no planificados			
Detalle	Gastos Calculados (Q)	Estimación	Subtotal (Q)
Se estima un porcentaje en gastos imprevistos o no planificados.	60,856.40	5%	3,042.82
Subtotal (Q)			3,042.82
Total			Q 63,899.22

Fuente: elaboración propia.

Apéndice 8. Manual de Usuario de Jenkins



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



MANUAL DE USUARIO: USO DE JENKINS

Autor: Jerson Eduardo Villatoro Nova
Carné: 2014-42819
Correo: jerduar@gmail.com

Guatemala, 08 de octubre de 2021

Continuación del apéndice 8.



INTRODUCCIÓN

Como parte del proyecto del proyecto de EPS titulado **DESARROLLO E IMPLEMENTACIÓN DE PRUEBAS AUTOMÁTICAS PARA LA PLATAFORMA DEL SISTEMA DE CONTROL DEL PROYECTO DE DESARROLLO DE TRANSFERENCIA TECNOLÓGICA (DTT) DE LA ESCUELA DE INGENIERÍA EN CIENCIAS Y SISTEMAS DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA** se presenta un manual para el uso de la máquina virtual que se encuentra en la nube que detalla como iniciar y acceder a Jenkins así como el ejecutar alguna de las pruebas que se encuentran publicadas en el servidor.

Requerimientos mínimos

- Conexión estable a internet
- La máquina virtual que contiene Jenkins debe tener dos (02) núcleos y cuatro (04) GB de memoria RAM para que pueda ejecutar las pruebas
- Si se hará uso de un sistema operativo Windows se recomienda instalar PuTTY o Git Bash para la conexión a la consola de la máquina virtual.

Proceso para Iniciar Jenkins

1. Conectar con el servidor en la nube, para ellos se necesitará el usuario, la contraseña, el puerto y la IP proporcionada con el administrador. Para hacerlos desde PuTTY solo se debe ejecutar el programa e ingresar la IP, el puerto y el tipo de conexión SSH.

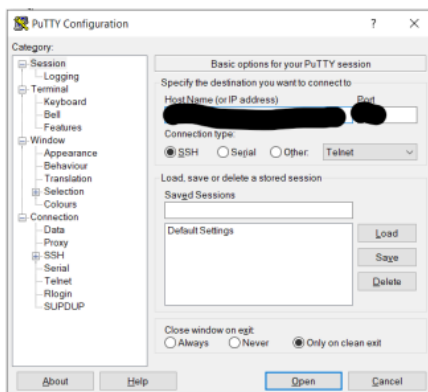


Imagen 1. Pantalla principal de PuTTY

Continuación del apéndice 8.



Luego PuTTY pedirá el usuario y contraseña.



Imagen 2. Solicitud de usuario y contraseña en PuTTY

Si se va a entrar usando GitBash o desde Linux se puede hacer uso del siguiente comando:

```
ssh <usuario>@<IP> -p <puerto>
```

Y luego ingresar se pedirá la contraseña.

2. Iniciar el contenedor de Jenkins. Ahora que se ha accedido al servidor se debe levantar el contenedor de Jenkins, para lo cual se hará uso del siguiente comando

```
Docker start <id contenedor>
```

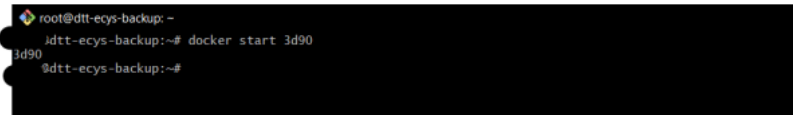


Imagen 3. Iniciando el contenedor de Jenkins

En el caso de no conocer el identificador del contenedor se puede usar el comando

```
docker ps -a
```



Imagen 4. Obteniendo identificador del contenedor

3. Verificar que está corriendo el contenedor.

Para verificar si se encuentra corriendo el contenedor podemos usar el comando:

Continuación del apéndice 8.



docker ps

Y verificamos que el estatus es *Up*

```
root@dt-ecys-backup:~# docker ps
CONTAINER ID   NAMES      IMAGE          COMMAND                  CREATED        STATUS
3d90a0b75bdf   ca0a4e83077a  "/sbin/tini -- /usr/..."  4 months ago   Up 3 days
```

Imagen 5. Verificación de estado del contenedor

O también podemos acceder desde el navegador a la dirección IP del servidor en el puerto configurado para Jenkins y debe mostrar la página de inicio de sesión.



Welcome to Jenkins!

Sign in

☐ Keep me signed in

Imagen 6. Pantalla de inicio de sesión de Jenkins

Continuación del apéndice 8.



Ejecución de Pruebas

Para el correr las pruebas es necesario configurara el servidor con los requerimientos mínimos que ya se han especificado en cuento a recursos. A continuación, se detallan los pasos para ejecutar una prueba.

1. Iniciar sesión

Al iniciar sesión se mostrará la carpeta raíz del proyecto y el panel de control de Jenkins

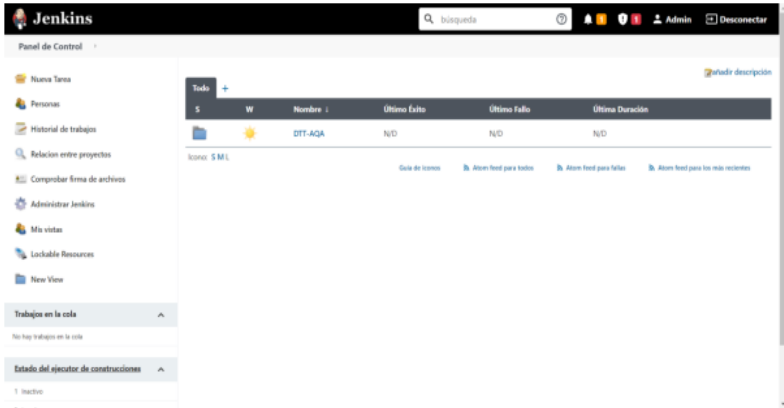


Imagen 7. Panel de control de Jenkins

Se puede navegar a partir de esta pantalla



Raíz del proyecto

A partir de esta carpeta se puede acceder a las pruebas que se encuentran ordenadas por roles y menús al darle click.

Continuación del apéndice 8.



DTT-AQA

Pruebas automaticas para la Plataforma DTT.

añadir descripción

S	W	Nombre	Último Éxito	Último Fallo	Última Duración
		AQA-Login	N/D	N/D	N/D
		AQA-RolAcademic	N/D	N/D	N/D
		AQA-RolStudent	N/D	N/D	N/D
		QA-RolSuperAdministrator	N/D	N/D	N/D

Icono: S M L

Guía de iconos

Atom feed para todos

Atom feed para fallos

Atom feed para los más recientes

Imagen 8. Carpetas por roles.

QA-RolSuperAdministrator

Folder name: DTT-AQA/AQA-RolSuperAdministrator
Pruebas automatizadas para el Rol Super-Administrator de la Plataforma DTT.

añadir descripción

S	W	Nombre	Último Éxito	Último Fallo	Última Duración
		AQA-CargaMasivaAsignacionCatedraticosPracticaFinal	6 Mes 29 dias - #13	4 Mes 9 dias - #14	4 Min 2 Seg
		AQA-CargaMasivaAsignacionPracticantesPracticaFinal	4 Mes 9 dias - #4	N/D	1 Min 55 Seg
		Gestión de Proyectos	N/D	N/D	N/D
		Menu Contenido	N/D	N/D	N/D
		Menu Usuarios	N/D	N/D	N/D

Icono: S M L

Guía de iconos

Atom feed para todos

Atom feed para fallos

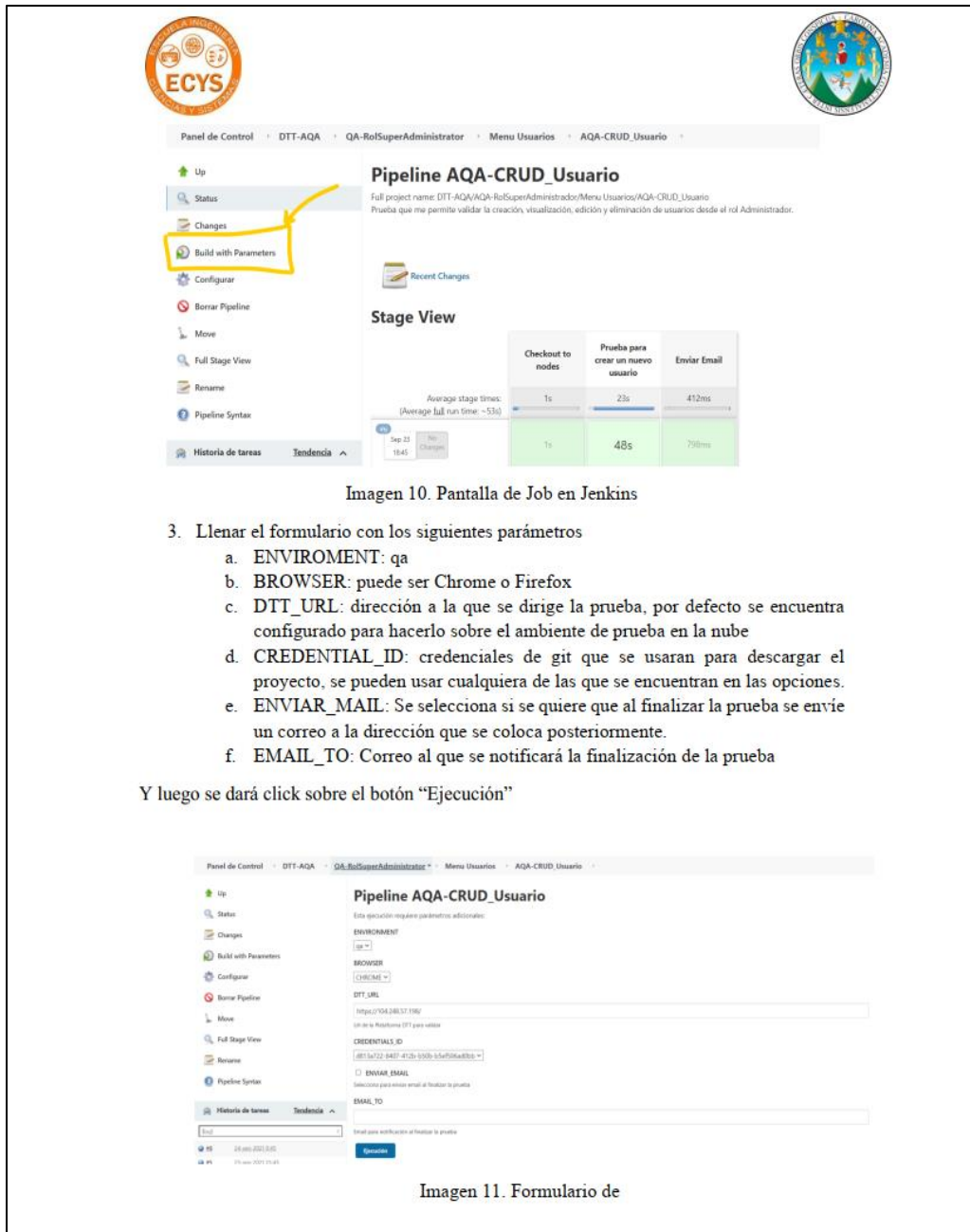
Atom feed para los más recientes

Imagen 9. Carpetas por menús del rol administrador

2. Seleccionar prueba a ejecutar

Para este ejemplo se usará la prueba creación de usuarios del rol administrador. Al acceder a la prueba se dará click sobre ejecución con parámetros.

Continuación del apéndice 8.



Continuación del apéndice 8.



4. Visualización de la ejecución de la prueba

Al ejecutar se regresará a la página principal de la prueba y se podrá ver el proceso en tiempo real.

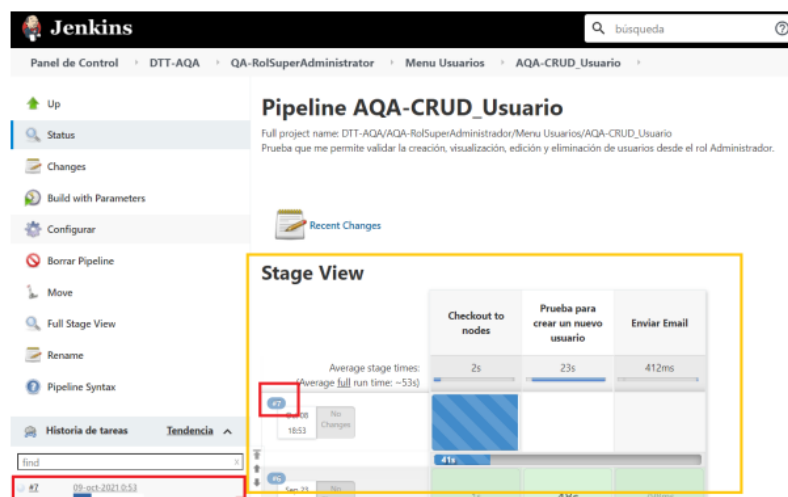


Imagen 12. Visualización del proceso del Job

Si se da click sobre las áreas rodeadas en rojo se desplegará la siguiente pantalla.

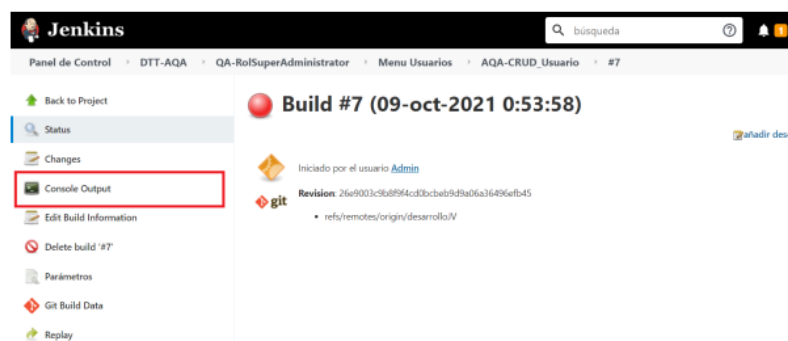
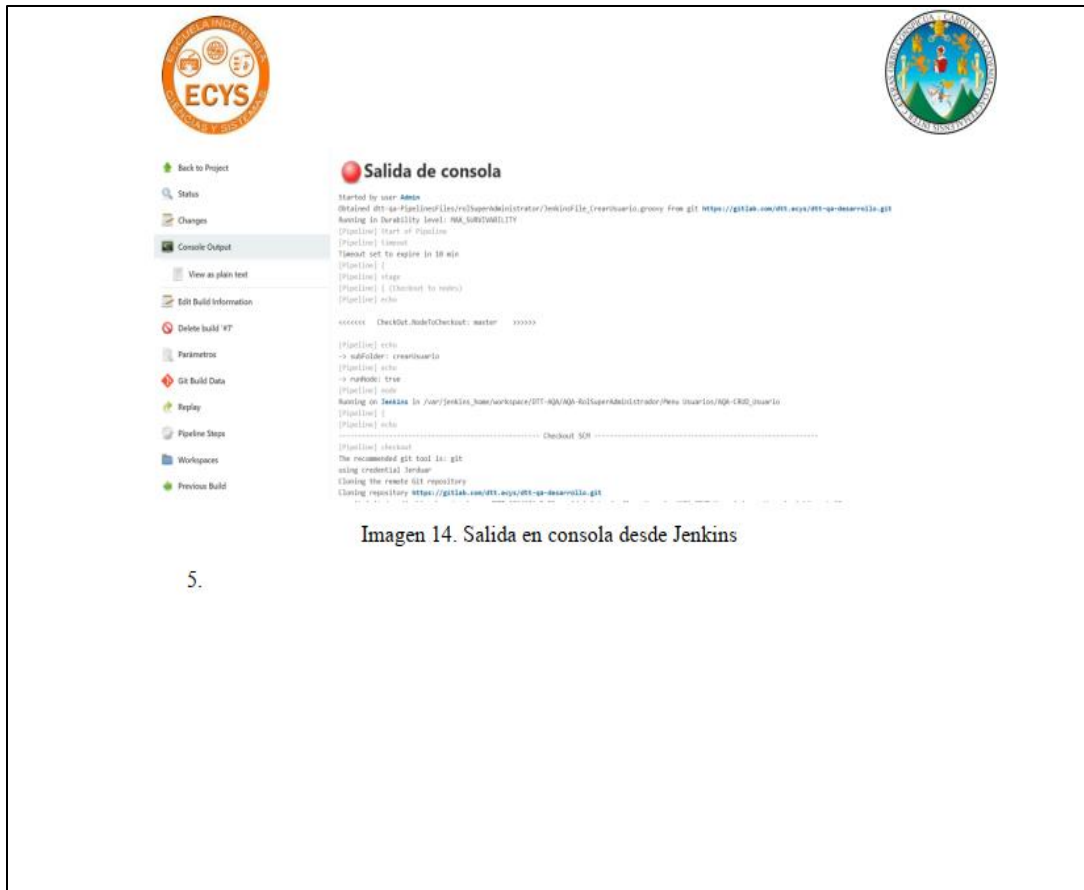


Imagen 13. Pantalla con información de la ejecución

Al dar click sobre *Console Output* se nos mostrará la salida en consola de la ejecución.

Continuación del apéndice 8.



Fuente: elaboración propia.

Apéndice 9. Manual de Desarrollo de Pruebas



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



GUÍA DE DESARROLLO DE PRUEBAS

Jerson Eduardo Villatoro Nova
Registro Estudiantil: 201442819
Correo: jerduar@gmail.com

Guatemala, 18 de marzo de 2022

Continuación apéndice 9.



Guía de desarrollo de pruebas

Para el desarrollo de pruebas automatizadas para la Plataforma DTT se recomienda seguir una serie de pasos y patrones que permitirán a largo plazo mantener la estructura en el código y dar seguimiento a los desarrolladores. Para la codificación se recomienda seguir los siguientes pasos:

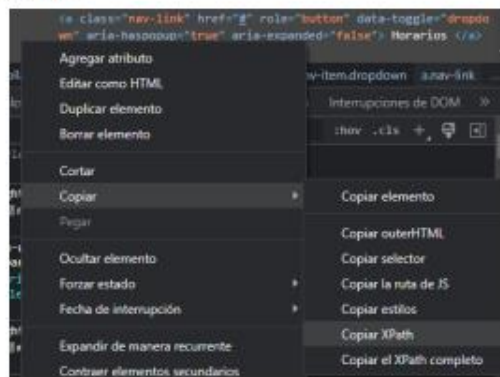
1. Definición de elementos web
2. Definición de acciones
3. Ordenamiento de acciones
4. Instancia de la prueba

1. Definición de elementos

Para cada prueba se recomienda tener una clase nombrada de la siguiente manera: `<nombre_prueba>_definitions.java` en la cual se definan todos los elementos con los cuales se interactuará durante la ejecución de la prueba. Estos elementos se definen de la siguiente forma:

```
@FindBy(how = How.XPATH, using = "//header/div[1]/div[1]/nav[1]/div[2]/ul[1]/li[7]/div[1]/a[3]")  
protected WebElement ulLoginOptionMenu;
```

Se recomienda un nombre descriptivo. Cada elemento es ubicado por el programa por medio de su XPATH el cual se extrae de la página web al hacer click derecho sobre el elemento y luego entrar a inspeccionar. Luego sobre el código HTML del elemento dar click derecho, copiar y copiar XPATH.



[illegible]

Para cada prueba se recomienda tener una clase nombrada de la siguiente manera: `<nombre_prueba>_actions.java` en la cual se definan todas las acciones que se realizarán sobre los elementos definidos en la clase de definiciones. Con acciones, se quiere decir, por ejemplo, dar click, escribir sobre un campo de texto, seleccionar una opción de un menú, etc.

[illegible]

77

Continuación del apéndice 9.



```
public void selectPageSeleccionarServicio() throws Exception {
    try {
        manejarFactura con "select servicio de (servicio) donde tipo de servicio sea";
        ejecutarConsulta(con, "select servicio de (servicio) donde tipo de servicio sea");
        seleccionarPagina();
    } catch (Exception ex) {
        ex.printStackTrace();
        throw new Exception("Error al seleccionar servicio: " + ex.getMessage());
    }
}

public void selectPageSeleccionarServicioPorCodigo() throws Exception {
    try {
        manejarFactura con "select servicio de (servicio) donde tipo de servicio sea";
        ejecutarConsulta(con, "select servicio de (servicio) donde tipo de servicio sea");
        seleccionarPagina();
    } catch (Exception ex) {
        ex.printStackTrace();
        throw new Exception("Error al seleccionar servicio: " + ex.getMessage());
    }
}

public void selectPageSeleccionarServicioPorCodigo() throws Exception {
    try {
        manejarFactura con "select servicio de (servicio) donde tipo de servicio sea";
        ejecutarConsulta(con, "select servicio de (servicio) donde tipo de servicio sea");
        seleccionarPagina();
    } catch (Exception ex) {
        ex.printStackTrace();
        throw new Exception("Error al seleccionar servicio: " + ex.getMessage());
    }
}
```

Al final, esta clase una lista de funciones que reflejan las acciones necesarias para realizar la prueba, paso a paso.

3. Ordenamiento de acciones

Ya que se ha definido las acciones se define una clase con el nombre <nombre prueba> logicTest en la cual se procede a colocar todas las acciones agrupadas y ordenadas para su ejecución.

```
public void seleccionarServicio() throws Exception {
    try {
        seleccionarServicio con "select servicio de (servicio) donde tipo de servicio sea";
        ejecutarConsulta(con, "select servicio de (servicio) donde tipo de servicio sea");
        seleccionarPagina();
    } catch (Exception ex) {
        ex.printStackTrace();
        throw new Exception("Error al seleccionar servicio: " + ex.getMessage());
    }
}

public void seleccionarServicioPorCodigo() throws Exception {
    try {
        seleccionarServicio con "select servicio de (servicio) donde tipo de servicio sea";
        ejecutarConsulta(con, "select servicio de (servicio) donde tipo de servicio sea");
        seleccionarPagina();
    } catch (Exception ex) {
        ex.printStackTrace();
        throw new Exception("Error al seleccionar servicio: " + ex.getMessage());
    }
}

public void seleccionarServicioPorCodigo() throws Exception {
    try {
        seleccionarServicio con "select servicio de (servicio) donde tipo de servicio sea";
        ejecutarConsulta(con, "select servicio de (servicio) donde tipo de servicio sea");
        seleccionarPagina();
    } catch (Exception ex) {
        ex.printStackTrace();
        throw new Exception("Error al seleccionar servicio: " + ex.getMessage());
    }
}
```

Continuación del apéndice 9.



Cada función de esta clase contiene las acciones ordenadas para su ejecución. En su constructor se recibe de parámetro el driver y se instancia la propiedad que contiene a las acciones.

```
import
    ProyectoAction, ProyectoAction;
import
    WebDriver;

public class ProyectoLogicTest {
    private WebDriver driver;
    private ProyectoAction action;

    public ProyectoLogicTest(WebDriver driver) throws Exception {
        this.driver = driver;
        this.action = new ProyectoAction(driver);
        execute();
        assertEquals("El usuario no se pudo loguear", action.getMessage());
    }
}
```

4. Instancia de la prueba

Primero se recomienda crear una clase con el nombre de la siguiente forma: <nombre_prueba>.java. En el constructor de esta clase se instanciará la clase logicTest y se llamará a la o las funciones que ejecutan las acciones en orden lógico.

```
import java.util.*;
import org.openqa.selenium.*;
import org.openqa.selenium.chrome.*;
import org.openqa.selenium.support.ui.*;
import org.testng.annotations.*;
import org.testng.asserts.*;

public class ProyectoLogicTest extends Login {
    private WebDriver driver;
    private ProyectoAction action;

    public ProyectoLogicTest(WebDriver driver) throws Exception {
        this.driver = driver;
        this.action = new ProyectoAction(driver);
        execute();
        assertEquals("El usuario no se pudo loguear", action.getMessage());
    }
}
```

Todas las clases de este tipo deben heredar de la clase Login, ya que todas las funciones a probar se encuentran una vez el usuario haya ingresado con sus credenciales al sistema DTT. Lo que quiere decir que primero deben realizarse las acciones correspondientes al login.

Continuación del apéndice 9.



Para finalizar la clase de la prueba debe instanciarse en la clase factory correspondiente al rol al que pertenece.

```
public class RolSuperAdministrator_Factory {  
  
    @Factory()  
    @Parameters({"env", "browser", "title"})  
    public Object[] rolSuperAdministratorFactory(String env, String browser, String title) throws Exception {  
        try {  
            messageOfAction( new "Running RolSuperAdministrator_Factory");  
            setConfiguration(env, browser, title);  
            String username = projectConfiguration.getUserDTT_rolSuperAdmin();  
            String password = projectConfiguration.getPassDTT_rolSuperAdmin();  
            return new Object[]{  
                new CargaMasivaAsignacionCatedraticosPorCSVPracticaFinal(username, password),  
                new CargaMasivaAsignacionPracticantesPorCSVPracticaFinal(username, password),  
                new CRUDUsuarios(username, password),  
                new CRUDCatedraticos(username, password),  
                new CRUDEstudiantes(username, password),  
                new CRUDRoles(username, password),  
                new CRUDProyectos(username, password),  
                new CRUDArchivos(username, password)  
            };  
        }  
    }  
}
```

Fuente: [Fotografía de Jerson Eduardo Villatoro Nova]. (Captura de pantalla, 2022). Colección particular. Guatemala.