

**AUTOMATIZACIÓN DE PRUEBAS DE SOFTWARE EN APLICATIVO
BANCARIO**

AUTOMATION SOFTWARE TESTING IN BANKING APPLICATION

LUIS CARLOS GALVIS GIRALDO

POLITÉCNICO COLOMBIANO JAIME ISAZA CADAVID

FACULTAD DE INGENIERÍAS

INGENIERÍA INFORMÁTICA

MEDELLÍN

2015

**AUTOMATIZACIÓN DE PRUEBAS DE SOFTWARE EN APLICATIVO
BANCARIO**

AUTOMATION SOFTWARE TESTING IN BANKING APPLICATION

LUIS CARLOS GALVIS GIRALDO

**TRABAJO DE GRADO PARA OPTAR AL TÍTULO DE INGENIERO
INFORMÁTICO**

ASESOR

RUBÉN DARÍO MUÑOZ, INGENIERO EN SISTEMAS

POLITÉCNICO COLOMBIANO JAIME ISAZA CADAVID

FACULTAD DE INGENIERÍAS

INGENIERÍA INFORMÁTICA

MEDELLÍN

2015

Nota de aceptación

Firma del jurado

Firma del jurado

Medellín, 1 de Diciembre de 2015

CONTENIDO

LISTA DE TABLAS.....	6
LISTA DE GRAFICOS	6
GLOSARIO	7
RESUMEN	9
ABSTRACT	10
INTRODUCCIÓN	11
1. PRESENTACIÓN DEL TRABAJO DE GRADO	12
1.1 PLANTEAMIENTO DEL PROBLEMA.....	12
1.2 JUSTIFICACIÓN	13
1.3 OBJETIVOS.....	14
1.3.1 Objetivo general.....	14
1.3.2 Objetivos específicos	14
1.4 ALCANCE	15
2. MARCOS ACORDES A LA MODALIDAD DE TRABAJO DE GRADO.....	16
2.1 MARCO TEORICO.....	16
2.1.1 Pruebas funcionales	16
2.1.2 Automatización de pruebas	17
2.1.3 Casos de prueba	18
2.1.4 Herramientas de Pruebas.....	19
2.1.5 Unified Functional Testing (UFT).....	20
2.1.6 Quality Center (QC).....	21
2.2 MARCO CONTEXTUAL	22
2.2.1 Politécnico Colombiano Jaime Isaza Cadavid	22
2.2.2 Tata Consultancy services	25
2.3 MARCO LEGAL.....	27

3.	ANTECEDENTES	28
4.	DESARROLLO DEL TRABAJO DE GRADO.....	31
4.1	DISEÑO DE PROCESOS AUTOMATIZADOS PARA LA SIMULACIÓN DE USUARIOS Y/O PROCESOS PROGRAMADOS.	31
4.1.1	Definición de la ruta crítica de los casos prueba que serán automatizados.....	31
4.1.2	Solicitud accesos y permisos requeridos para la ejecución de los casos prueba.	32
4.1.3	Validación de adherencia para las herramientas de automatización.....	33
4.1.4	Grabación de scripts.....	34
4.1.5	Depuración, edición y parametrización de scripts.	38
4.2	SIMULACIÓN DE USUARIOS O PROCESOS PROGRAMADOS DE FORMA AUTOMATIZADA SOBRE EL APLICATIVO BANCARIO.....	39
4.2.1	Solicitud y coordinación con el cliente para la adecuación de data para las pruebas. 39	
4.2.2	Realización de smoke test y validación del ambiente.....	39
4.2.3	Ejecución de robots.....	39
4.2.4	Validación de la correcta ejecución de los robots.	41
4.3	ANÁLISIS DEL COMPORTAMIENTO DEL APLICATIVO BANCARIO E IDENTIFICACIÓN DE FALLOS.	42
4.3.1	Análisis de los resultados recopilados en las ejecuciones.....	42
4.3.2	Validación con el equipo de trabajo sobre las consideraciones o fallos identificados.....	43
4.3.3	Reporte en el sistema o herramienta de pruebas los fallos evidenciados y remitiendo a quien corresponda para su solución.	43
4.3.4	Generación del informe sobre la eficiencia en la automatización de las pruebas. ..	44
4.4	CRONOGRAMA 2015.....	45
5.	CONCLUSIONES	46
6.	REFERENCIAS.....	47
7.	ANEXOS.....	48

LISTA DE TABLAS

<i>Tabla 1. Casos prueba automatizados.</i>	31
<i>Tabla 2. Accesos y permisos requeridos.</i>	33
<i>Tabla 3. Esfuerzo en diseño.</i>	35
<i>Tabla 4. Resultados en ejecución.</i>	41
<i>Tabla 5. Estados finales.</i>	42
<i>Tabla 6. Clasificación de Bugs.</i>	42
<i>Tabla.7. Tiempo de pruebas.</i>	45

LISTA DE GRAFICOS

<i>Figura 1. Misión Tata.</i>	26
<i>Figura 2. Oráculo automatizado.</i>	29
<i>Figura 3. Gherkin.</i>	30
<i>Figura 4. Barras casos prueba.</i>	32
<i>Figura 5. UFT.</i>	34
<i>Figura 6. Resultado de ejecución.</i>	40
<i>Figura 7. Quality Center.</i>	44
<i>Figura 8. Cronograma.</i>	45

GLOSARIO

- **Oráculo de prueba:** cualquier mecanismo encargado de comprobar las salidas y emitir en consecuencia un veredicto sobre la ejecución de la prueba que permita valorar si el programa se comporta de forma correcta.
- **Issues:** Un issue puede ser el arreglo de un fallo, una característica pedida, una tarea, un pedido de Documentación específico y todo tipo de solicitud al equipo de desarrollo.
- **Pruebas de Regresión:** Son pruebas, tras la modificación de un programa previamente probado, con el objeto de asegurar que no se han introducido o descubierto defectos en áreas que no hubieran sido objeto de modificación como resultado de los cambios realizados. Se realizan cuando el software o su entorno ha sido modificado.
- **Automatización de Pruebas:** Consiste en el uso de software especial para controlar la ejecución de pruebas y la comparación entre los resultados obtenidos y los resultados esperados.
- **Pruebas de software:** Son los procesos que permiten verificar y revelar la calidad de un producto software antes de su puesta en marcha. Básicamente, es una fase en el desarrollo de software que consiste en probar las aplicaciones construidas. En este sentido, se prueba y se detectan los errores que tiene el aplicativo.
- **Script o robot:** Son los datos y las instrucciones escritas con una sintaxis formal, almacenado en un archivo y usado por una herramienta de automatización de las pruebas.

- **Ciclo de Pruebas:** Las actividades de la prueba se realizan para una determinada versión del producto, sobre la cual se ejecutan las pruebas y se reportan los incidentes encontrados. Para cada versión del producto se realizan alguna o todas las tareas asociadas a las pruebas, a esto se le llama un ciclo de prueba.
- **Salida a producción:** Es la promoción de los objetos de un sistema de un ambiente de pruebas a un ambiente productivo. El ambiente productivo es aquel donde se realizan transacciones en tiempo real y las cuales tienen afectación directa sobre los datos sensibles del cliente.

RESUMEN

El proyecto consistió en el desarrollo de práctica profesional en el sector de TI, donde se aplicaron los conocimientos adquiridos durante la carrera de Ingeniería Informática. En este proceso se recopiló información, para posteriormente documentar el trabajo de grado y sustentarlo ante la universidad, demostrando así que se han adquirido los conocimientos y habilidades necesarios para el título de Ingeniero Informático.

La práctica empresarial se llevó a cabo en el área de automatización de pruebas de software, de la multinacional prestadora de servicios en TI Tata Consultancy Services sede Colombia, situada en la ciudad de Medellín.

Como proyecto se implementó la automatización de pruebas funcionales para un cliente bancario, al cual la empresa Tata Consultancy Services le presta actualmente el servicio de pruebas de software.

ABSTRACT

The project was done on the development of the organizational practices in the IT sector. The knowledge acquired during the Information Science Engineering was applied in this project and it was presented in the university as degree work(degree project), demonstrating that the required knowledge for an Information science engineer has been acquired.

The internship was done in the automation of testing of software in the multinational organization - Tata Consultancy Services situated in Medellin.

In the project the automation of functional testing was implemented to a Bank, which is a current client of Tata Consultancy Services.

INTRODUCCIÓN

Entre los meses de marzo y noviembre de 2015 se planteó, desarrolló y ejecutó como trabajo de grado la implementación de una automatización de pruebas de Software, esta se desarrolló bajo el esquema de práctica profesional en la empresa Tata Consultancy Services Colombia en la ciudad de Medellín quien le presta servicios de TI a un cliente bancario, entre el múltiple catálogo de servicios que la compañía le proporciona al cliente se encuentra el de automatización de pruebas de software donde se identificó la necesidad de implementar flujos automatizados de prueba en uno de sus aplicativos core que presenta tiempos altos en prueba cada vez que se adiciona un módulo o se implementa una orden de cambio.

La práctica se desarrolló con un equipo de expertos profesionales en el tema bajo la supervisión y asesoría del líder de automatización de la compañía, en esta experiencia fue necesario investigar sobre métodos y herramienta de automatización y aprender de los procesos propios de la compañía y el cliente, aportando los conocimientos adquiridos durante toda la carrera de formación.

La automatización de las pruebas se implementó en un aplicativo web cliente servidor el cual tiene implementado múltiples procesos y hace parte de los aplicativos core del cliente bancario en su manejo interno.

1. PRESENTACIÓN DEL TRABAJO DE GRADO

1.1 PLANTEAMIENTO DEL PROBLEMA

Actualmente la empresa TATA Consultancy Services Colombia, se encarga de realizar pruebas de Software sobre un aplicativo que es utilizado para realizar y controlar diferentes flujos de proceso (ejemplo solicitud de crédito de consumo, solicitud de viáticos, custodia de documentos, entre otros) en un cliente bancario.

Debido a las múltiples ejecuciones de ciclos de pruebas funcionales sobre este aplicativo, los tiempos invertidos en la etapa de ejecución, así como la eficiencia de los analistas de prueba, se han visto afectados, ocasionando una mayor inversión de tiempo, lo que impacta directamente en los costos y las oportunas salidas a producción, afectando la operación del cliente y dejando una sensación de inconformidad en este.

La automatización de pruebas de software consiste en simular y controlar la ejecución de pruebas con el uso de software especializado, permitiendo realizar una comparación entre los resultados obtenidos y los resultados esperados en forma automatizada. Además permite realizar pruebas en forma repetitiva en un proceso ya existente donde la ejecución manual es compleja a nivel operativo por parte de los analistas.

1.2 JUSTIFICACIÓN

Con base en anterior, se propuso implementar este tipo de pruebas en el aplicativo bancario, obteniendo eficiencia, reducción de tiempos, ausencia de fallo humano en la ejecución, distribución de tareas y las demás bondades que ofrece la automatización de pruebas de software. Esto permite realizar regresión de pruebas completa con un esfuerzo menor y permite a los analistas de pruebas realizar actividades no tan operativas, sino, más analíticas.

De esta forma, se solucionaría la problemática que actualmente presenta la empresa TATA Consultancy Services con su cliente bancario, logrando la satisfacción del mismo y aportando a que la empresa continúe prestando un buen servicio y posicionándose en el mercado local e internacional en servicios de TI.

1.3 OBJETIVOS

1.3.1 Objetivo general

- Implementar la automatización de pruebas de software, mediante el diseño y la ejecución en simulación de usuarios y/o procesos programados, para la identificación de defectos en un aplicativo bancario.

1.3.2 Objetivos específicos

- Diseñar procesos automatizados sobre los casos de prueba que hacen parte de la regresión de pruebas funcionales, para la simulación de usuarios y/o procesos programados.
- Simular usuarios y/o procesos programados de forma automatizada, con base en los diseños realizados, que lleve al análisis del comportamiento en el aplicativo bancario.
- Analizar el comportamiento del aplicativo bancario en las pruebas de regresión, evaluando las evidencias generadas en la ejecución de los casos pruebas, para la identificación de fallos en el aplicativo.

1.4 ALCANCE

El alcance se da sobre uno de los aplicativos core en el módulo de manejo de documentos en custodia en el cliente bancario de la empresa Tata Consultancy Services, en la automatización de los casos de prueba definidos como ruta crítica, los cuales son ejecutados en la regresión de las pruebas funcionales del software que consta de 384 casos de prueba.

- Analizar los casos de prueba candidatos a ser automatizados.
- Diseño de scripts
- Parametrización de scripts.
- Ejecutar procesos automatizados.
- Analizar resultados de ejecución.
- Registrar detalle de fallos presentados.

No hace parte del alcance:

- Definir ruta crítica.
- Diseñar paso a paso en el caso de prueba funcional.
- Adecuar ambiente de prueba.
- Crear usuarios.
- Informar o registrar la causa de fallos funcionales en el aplicativo.
- Solucionar fallos funcionales en el aplicativo.
- Otro tipo de prueba (Performance, seguridad, usabilidad...).

2. MARCOS ACORDES A LA MODALIDAD DE TRABAJO DE GRADO

2.1 MARCO TEORICO

2.1.1 Pruebas funcionales

También llamadas pruebas de funciones, con estas se busca verificar que el sistema haga lo que el usuario y/o cliente solicito en la especificación de requisitos. Dicho tipo de pruebas se enfoca en el análisis de las entradas y salidas del sistema, por lo cual no se valida el código, es decir, no se comprueba cómo internamente se generan las respuesta del aplicativo.

En las pruebas funcionales se tienen varios enfoques, entre ellos el de seguridad, donde se valida que los usuarios tengan acceso únicamente a las funciones y datos autorizados; otra perspectiva son las pruebas de componentes, donde se verifica la forma en que se integran los diferentes componentes del sistema entre sí. Adicionalmente se tiene el enfoque de pruebas integrales, donde se valida la integración del sistema con las demás aplicaciones que componen la solución.

Para la definición de las funcionalidades a probar se toman como insumo, los documentos de Casos de Uso y levantamiento de requisitos, ambos suelen ser generados y proporcionados por los cuales y contienen las necesidades del mismo, las cuales deberán ser garantizadas en las pruebas a ejecutar.

Las pruebas funcionales pueden ser ejecutadas de forma manual, donde el analista de pruebas valida el correcto funcionamiento del aplicativo, ingresando al mismo y siguiendo una serie de pasos los cuales están definidos en el diseño de casos prueba. Otra forma de ejecutar las pruebas de funcionalidades, es por medio, de la automatización, donde el analista utiliza una o

varias herramientas la cual se encarga de ejecutar, validar, comparar los resultados obtenidos versus los esperados.

2.1.2 Automatización de pruebas

El objetivo que se busca al ejecutar pruebas de forma automatizada, es disminuir el esfuerzo requerido para validar una misma funcionalidad o serie de funcionalidades en repetidas ocasiones; esto se logra por medio de la generación de scripts los cuales permiten ser ejecutados tantas veces como sea necesario, sin la supervisión constante de un analista.

Durante la ejecución de una prueba, es posible encontrar uno o varios defectos, los cuales hagan necesario la modificación del programa, requiriendo la re-ejecución de los casos que originalmente tenían el error. Pero debido a los cambios realizados, también es necesario la ejecución adicional de aquellas funcionalidades críticas para el negocio, con el fin de garantizar que estas no hayan sido afectadas por la solución al error. Esta re-ejecución es costosa y demorada, lo cual puede generar retrasos en el cronograma de pruebas o sobreesfuerzos, ocasionando desfases en el presupuesto. Para evitar que estos riesgos se materialicen, se utiliza la automatización de pruebas, la cual permite que se ejecute tantas veces como se requiera los casos de prueba automatizados de una forma rápida, disminuyendo los sobreesfuerzos por re-ejecución y los desfases en el proyecto.

La automatización también es de gran utilidad, cuando se requiere probar los mismos casos prueba en diferentes navegadores, es decir, cuando se están realizando validaciones a aplicaciones web e incluso para apps. Ya que la automatización permite la ejecución simultánea del caso en diferentes navegadores, lo cual reduce la cantidad de personas requeridas para esta verificación, ya que solo es necesario el automatizador quien se encarga de controlar la ejecución en los diferentes escenarios.

La automatización consiste en llevar los casos prueba a scripts automatizados, es decir, convertirlos a un lenguaje formal, los cuales pueden ser interpretados por las herramientas de automatización. En estos scripts se detallan los datos de entrada, los pasos o instrucciones, que deberá seguir la herramienta con el fin de obtener el resultado esperado. De acuerdo a las características de los casos de prueba a validar, es posible que varios de estos sean diseñados en un solo script.

Algunas de las herramientas utilizadas para el diseño de los scripts de automatización, requieren de técnicas de programación como librerías, macros, sub-programas, mientras que en otros sistemas, el diseño se logra por medio de la ejecución de los pasos utilizando dichos sistemas, los cuales graban los movimientos del analista, generando luego un script, el cual puede ser modificado y personalizado con los datos requeridos cada que se va a re-ejecutar.

2.1.3 Casos de prueba

Un caso de prueba, consiste en la validación de una conducta esperada de un aplicativo o software, estos pueden tener un resultado positivo o negativo, es decir, pueden representar un resultado que se espera obtener del software o aquel que se espera no obtener. Estos representan todas las posibles combinaciones de las funcionalidades del sistema, donde para una misma funcionalidad es posible diseñar uno o más casos de prueba.

Debido a que los posibles caminos de pruebas de un sistema son infinitos, usualmente se utiliza para la definición de cuales casos se van a diseñar y posteriormente probar, diferentes técnicas como árboles de decisiones, valores limites, clases de equivalencia, diseño basado en la experiencia.

Los casos prueba están compuestos por un conjunto de valores de entrada, precondiciones para la ejecución del caso, descripción del mismo para determinar cuál es su objetivo, resultado esperado y las post-condiciones de la ejecución.

2.1.4 Herramientas de Pruebas

Existen diferentes tipos de herramientas de pruebas, las cuales permiten la gestión de las pruebas desde distintos enfoques.

Algunas se encargan de la gestión por medio del manejo de la documentación general de las pruebas, desde los requisitos, pasando por el plan y casos de pruebas, hasta la generación de los reportes de error; estas permiten la generación de reportes de las pruebas, indicando porcentaje de avance, cantidad de defectos, desfases, información que apoya la toma de decisiones por parte de los coordinadores de las pruebas.

Otras herramientas de testing se encargan de apoyar la ejecución de pruebas de carga y rendimiento, las cuales proporcionan reportes detallados del comportamiento del sistema probado, donde se especifican los cuellos de botella encontrados. También estas herramientas permiten conocer el volumen de datos y la carga soportada por el aplicativo.

Herramientas para las pruebas de seguridad web, son aquellas que al ingresar los datos correspondientes al sistema, validan las vulnerabilidades del mismo, como puertos inseguros, inserción de código, entre otras posibles fallas de seguridad del software validado.

Pruebas en dispositivos móviles son cada vez más comunes y requeridas por las diferentes áreas de tecnología, pero estas pruebas tienen una complejidad, al requerir ser ejecutadas en todos los dispositivos y navegadores soportados por el software. Dada esta complejidad se hace necesario para dichas pruebas la utilización de herramientas específicas para este fin, las cuales se conectan a los dispositivos de prueba o simulan dicha conexión, con el fin

de ser ejecutado por el analista de pruebas o por la herramienta de automatización los casos de prueba diseñados.

En el mercado se tienen herramientas de testing tanto Open Sources, como comerciales, donde estas últimas por lo general ofrecen paquetes donde se combinan los diferentes enfoques de las pruebas, permitiendo sincronización entre las mismas.

2.1.5 Unified Functional Testing (UFT)

La Suite ALM de la empresa Hewlett Packard, es una de la suite comerciales de pruebas más completa. La cual está compuesta por una serie de aplicativos que permiten tanto la gestión de las pruebas por medio de Quality Center, como la ejecución automatizada por medio de UFT, pruebas de carga y rendimiento a través de Load Runner, validación de seguridad de aplicaciones web por medio de Quick Test Professional, realización de pruebas a dispositivos móviles por medio de HP Mobile Center, entre otras aplicaciones.

UFT es la herramienta de Hewlett Packard encargada de la automatización de pruebas. Por medio de UFT, se puede validar tanto la interfaz gráfica de usuario, como el comportamiento interno de la aplicación, es decir, la lógica empresarial. Adicionalmente permite la ejecución de los scripts en diferentes plataformas móviles. La principal característica de la automatización por medio de esta herramienta es la modularidad, es decir, se puede diseñar los script de forma modular, los cuales son más sencillos de modificar en caso de un control de cambios.

Otra de las características que diferencia a UFT es la posibilidad de generar scripts basados en los procesos de negocio, permitiendo de forma rápida y sencilla realizar pruebas End to End, es decir, probar no solo las funcionalidades de forma independiente, sino todas las partes que componen el proceso de negocio a validar.

Adicionalmente esta herramienta se integra con la aplicación Quality Center, lo cual permite tener toda la información en un solo lugar, generando así reportes de estado de la prueba tanto manual como automatizada.

2.1.6 Quality Center (QC)

Herramienta que permite la gestión de las pruebas, por medio de una serie de módulos que comprenden desde la gestión de los requisitos, hasta la gestión de los defectos.

QC proporciona al equipo de pruebas las herramientas necesarias para tener la trazabilidad de la cobertura de pruebas, permitiendo vincular los casos prueba a los requisitos que estos estén verificando, esta funcionalidad ayuda a validar si se requiere de más casos, con el fin de cubrir todos los requisitos que se encuentran dentro del alcance de la prueba.

Reutilización de los casos diseñados, es la funcionalidad de QC, que permite optimizar el tiempo en el diseño, ya que es posible utilizar los mismos casos en diferentes pruebas.

Quality Center permite realizar trazabilidad de los errores, con los casos de prueba y los requisitos.

Otra de las bondades de QC en la gestión de pruebas es la integración entre todas las herramientas de la Suite de Hewlett Packard, permitiendo que los casos prueba diseñados en una sean utilizados en las demás. Adicionalmente esta integración habilita la generación de reporte del estado de todos los tipos de prueba desde esta herramienta, generando informes consolidados.

Esta herramienta utiliza dos de los cuatro tipos de framework de automatización de pruebas de software, el Data Driven Framework y Keyword Data Framework.

2.2 MARCO CONTEXTUAL

2.2.1 Politécnico Colombiano Jaime Isaza Cadavid

El Politécnico Colombiano Jaime Isaza Cadavid es una institución universitaria de carácter público, adscrita al Gobierno Departamental de Antioquia y fundada en marzo de 1964 que ofrece educación superior en los niveles técnico, tecnológico y universitario mediante una oferta de programas académicos de pregrado y posgrado en distintas áreas de la ingeniería, la administración, las ciencias agrarias, la comunicación audiovisual, el deporte y la recreación. Su oferta académica incluye cursos de educación continuada y educación no formal.

El Politécnico Colombiano ofrece también servicios para el sector empresarial y para la comunidad a través de programas y grupos especiales de asesoría e investigación.

Misión. Somos una Institución de educación superior estatal de vocacionalidad tecnológica, que con su talento humano ofrece una formación integral con programas de calidad en pregrado y posgrado, apoyados en la gestión del conocimiento de base científica; promovemos acciones innovadoras desde la investigación y la proyección social, para contribuir al desarrollo económico, social y ambiental de Antioquia y Colombia.

Visión. En 2020, el Politécnico Colombiano Jaime Isaza Cadavid, será reconocido como una Institución de alta calidad académica, con énfasis en la formación y gestión tecnológica, la investigación aplicada y la proyección social, en beneficio del desarrollo económico, social y ambiental, con presencia en las regiones de Antioquia y el País; articulado a las dinámicas del sector productivo, a la política pública y al crecimiento de la cobertura en educación.

Principios. El Politécnico Colombiano Jaime Isaza Cadavid, en cumplimiento de sus objetivos adopta como principios los contenidos en la Ley 30 de 1992 y en aquellas que la

adicionen, modifiquen o sustituyan. Coherente con la filosofía y en desarrollo de su autonomía, basa su gestión en los siguientes principios:

- **Responsabilidad social:** instituye la responsabilidad social para el cumplimiento de su Misión y Visión, teniendo en cuenta que responde ante la sociedad mediante sus órganos de gobierno.
- **Excelencia académica:** encamina su labor hacia la consecución de niveles de excelencia, para lo cual no escatimará esfuerzos que lo conduzcan a obtener logros cada vez mayores en los procesos académicos.
- **Innovación:** dada su vocación técnica y tecnológica, la Institución apoya y fomenta actividades conducentes a la innovación, en los campos que tengan que ver con el ejercicio de la docencia, la investigación y la extensión, con el fin de contribuir de manera eficiente y constante al desarrollo local, regional y del país.
- **Equidad:** se compromete a llevar a cabo sus actuaciones con justicia, buscando el beneficio educativo de todos.
- **Universalidad:** orienta sus procesos de docencia, extensión, proyección social e investigación, hacia la búsqueda de diversidad de campos del conocimiento y hacia el impulso del saber, mediante las relaciones entre campos especializados de la ciencia y la tecnología.
- **Solidaridad:** impulsa las relaciones interpersonales basadas en la dignidad humana, estrategias de crecimiento y de sensibilidad social, para el beneficio común.
- **Sentido de ciudadanía:** expresado mediante la creación de espacios de convivencia que faciliten la colaboración y el apoyo, mediante la consolidación en un ambiente de

respeto y apertura en las relaciones interpersonales, que aporten al desarrollo de la ética y al compromiso ciudadano.

- **Convivencia:** al acoger la condición social del hombre, la Institución establece como uno de sus principios básicos el de la convivencia de sus participantes, mediante el respeto mutuo y el tratamiento constructivo de la divergencia de ideas y el acatamiento a los principios de la dignidad humana.
- **Transparencia:** uno de los fundamentos de la acción institucional es la transparencia, entendida como la rectitud y coherencia en el obrar y la disposición permanente de hacer públicos todos sus actos.
- **Participación:** en su labor de formar ciudadanos, promueve actitudes críticas y fomenta la participación ciudadana, estimula el trabajo en equipo, la cooperación y ofrece respuestas a los retos que impone la democracia.
- **Uso de las tecnologías de la comunicación (TIC):** las nuevas tecnologías de la información y comunicación, son aquellas herramientas computacionales e informáticas que procesan, almacenan, sintetizan, recuperan y presentan información representada de la más variada forma y constituyen nuevos soportes y canales para transmitir, compartir y socializar el conocimiento y por ello se convierten en medios e instrumentos importantes en la Institución o para mejorar la gestión administrativa y académica para dinamizar los procesos de enseñanza aprendizaje y para llegar a nuevos públicos, ampliando el radio de acción social de la Institución.
- **Internacionalización:** mediante este proceso, la Institución viabiliza la globalización de la enseñanza y el aprendizaje universitario. Es la forma como se estrechan los vínculos y niveles de integración con las diferentes instituciones de educación

superior, en el ámbito internacional para dinamizar el intercambio científico, técnico, tecnológico y cultural de: directivos, profesores y estudiantes así como el aprendizaje de los contenidos curriculares donde el conocimiento respectivo se reproduzca sin importar su ubicación espacial.

- **Medio ambiente:** tiene en cuenta en su actuar al entorno, que afecta y condiciona especialmente las circunstancias de vida de las personas o la sociedad, su conjunto es entendido de que el medio ambiente, comprende el conjunto de valores naturales, sociales y culturales existentes, en un lugar y un momento determinado, que influyen en la vida del hombre y en las generaciones venideras. Es decir, no se trata sólo del espacio en el que se desarrolla la vida, sino que también abarca elementos tan intangibles, como la cultura y por ello la institución, debe contribuir a la formación de los estudiantes en la concepción científica del mundo y la comprensión de los problemas del medio ambiente, del desarrollo sostenible y de la necesidad de la educación ambiental y su vínculo con la sociedad.
- **Bienestar:** en cumplimiento de sus objetivos, adopta como principio el mejoramiento de la calidad de vida y el desarrollo integral de todos los miembros de la Comunidad Politécnica, igualmente brinda bienestar y contribuye a la formación integral del ser.

2.2.2 Tata Consultancy services

El GRUPO TATA, fundado en 1.868 por JAMSETJI TATA, es el mayor holding empresarial de India y uno de los más grandes conglomerados industriales de Asia, con alrededor de 300.000 empleados e ingresos anuales cercanos a USD\$ 55.000 millones, representando un 3.2% del producto Interno Bruto de India. El GRUPO TATA comprende 98 empresas, entre las más destacadas se encuentran: TATA CONSULTANCY SERVICES, TATA MOTORS, TATA

POWER, TATA STEEL, TATA TELECOMMUNICATIONS Y TATA TEA, cubriendo varios sectores.

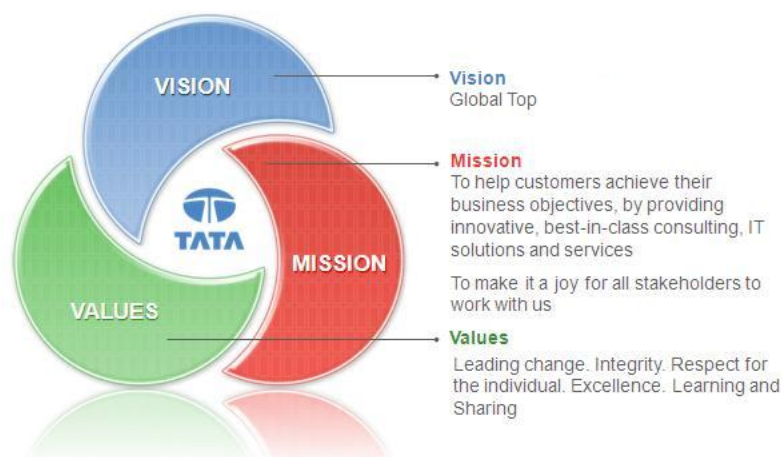


Figura 1. Misión Tata.

Tata Consultancy Services sucursal Colombia comenzó sus operaciones en el año 2005, incrementando progresivamente su oferta, brindando soluciones avanzadas de TI, servicios de mantenimiento y desarrollo de sistemas basados en diversas tecnologías, servicios de Tercerización de Procesos de Negocios (BPO), consultoría de procesos, consultoría TI e implementaciones de productos de TI. Actualmente TCS Colombia se especializa en la atención de proyectos en su mayoría bancarios, atendiendo a un grupo selecto de clientes como Bancolombia, Colpatría, Davivienda, City, Metro, Comfenalco Antioquia, Empresa Telefónica de Bogotá (ETB), Goodyear, Telefónica-Telecom, Teléfonos de México (Telmex) entre otros.

2.3 MARCO LEGAL

Todo el material creado, tratado y referenciado en el presente trabajo se encuentra protegido por leyes nacionales e internacionales entre las cuales se encuentran:

- **Decreto 1360 de 1989:** Por el cual se reglamenta la inscripción del soporte lógico (software) en el Registro Nacional del Derecho de Autor
- **Artículo 30 del acuerdo de Cartagena:** Régimen común sobre derecho de autor y derechos conexos.
- **Ley 23 de 1982:** Sobre derechos de autor.
- **Ley 178 de 1994:** Por medio de la cual se aprueba el "Convenio de París para la Protección de la Propiedad Industrial".
- **Ley 603 de 2000:** En la cual todas las empresas deben reportar en sus Informes Anuales de Gestión el cumplimiento de las normas de propiedad intelectual y derechos de autor.

En el marco del código legal y de conducta de Tata Consultancy Services se encuentran:

- Clausula 3: Competencia
- Clausula 12: Representación pública de la compañía y el grupo.
- Clausula 17: Conducta Ética.
- Clausula 18: Cumplimiento de las regulaciones.
- Clausula 19: Empleo simultaneo.
- Clausula 20: Conflicto de intereses.
- Clausula 21: Transacción de valores e información confidencial.
- Clausula 22: Protección de los bienes de la compañía.
- Clausula 24: Integridad de los datos proporcionados.

3. ANTECEDENTES

En Colombia la automatización de pruebas de software es relativamente nueva y las empresas apenas están adquiriendo conciencia de la importancia de automatizar las pruebas en sus nuevos aplicativos.

En nuestro país existen pocas personas capacitadas y con el conocimiento necesario para implementar la automatización de pruebas, además en las universidades poco o nada es el conocimiento en este tema que adquieren los profesionales, puede ser por que no se han dado cuenta de su importancia o simplemente no se cuenta con los profesores calificados en el tema.

Actualmente en Colombia las empresas que prestan servicio en la automatización de pruebas son:

- Tata Consultancy Services
- Globant
- Choucair testing

Actualmente Tata Consultancy Services, no ha realizado un proyecto de automatización de pruebas sobre este aplicativo bancario en particular, por lo que no se cuenta con métodos o procesos que puedan ser reutilizados en el presente proyecto.

El cliente bancario no presentó un proyecto previo realizado por un proveedor diferente sobre el cual se deba continuar el trabajo. Tampoco presentó un repositorio con una base de conocimiento en la automatización de pruebas sobre dicho aplicativo.

Al realizar una búsqueda de proyectos previos que pudieran aportar en el desarrollo del proyecto, encontré:

- Kim-Park, D. S., Riva, C. D. L., & Tuya, J. (2011). Aplicación de un oráculo de prueba automatizado a la evaluación de salidas de programas basados en XML.

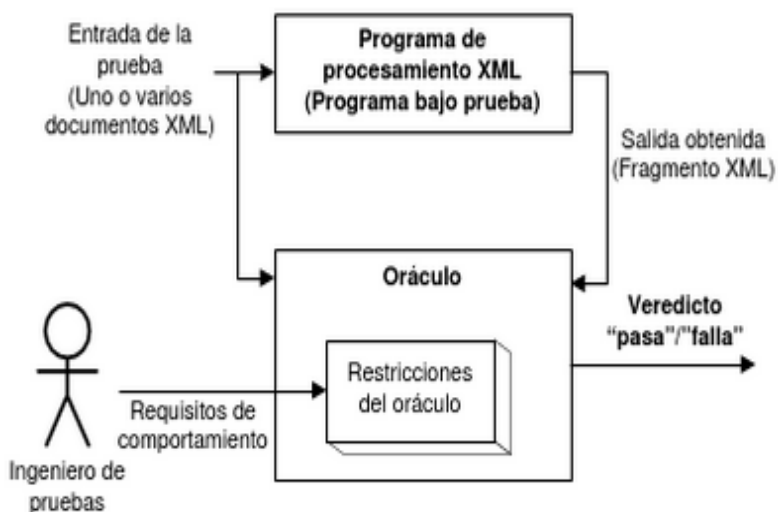


Figura 2. Oráculo automatizado.

Una ventaja importante es que se podría usar en las pruebas de mensajes XML, los cuales usan los componentes de Integración entre los diferentes sistemas del banco, sin embargo este solo es aplicable para mensajería basada en XML.

- Delgado Dapena, Martha Dunia, López Trujillo, Yucely, Chávez Valiente, Indira.

Propuesta de utilización del razonamiento basado en casos para la recuperación de procedimientos de prueba funcionales.

Desarrollar pruebas basados en procedimientos de pruebas ya realizados y validados.

Permite la reutilización y el diseño de CP en etapas tempranas evitando reproceso y

sobrecostos, sin embargo por ser una propuesta nueva es muy probable que todo el equipo no este sincronizado con este.

- Pantaleo, G. (2011). Calidad en el desarrollo de software.

Un libro que trata puntos de suprema importancia al momento de automatizar pruebas de software.

Obstáculos para automatizar las pruebas.

Que no debería automatizarse.

Que debería automatizarse.

Estrategias para comenzar la automatización.

- Lo nuevo o tendencia en la automatización de pruebas.

```
Scenario: Some determinable business situation
  Given some precondition
  And some other precondition
  When some action by the actor
  And some other action
  And yet another action
  Then some testable outcome is achieved
  And something else we can check happens too
```

Figura 3. Gherkin.

Gherkin una sintaxis simple que se complementa con las herramientas Cucumber y Selenium. Es orientado a metodologías ágiles, compatibilidad, diseño de casos prueba basado en historias de usuario, simplicidad y multilenguaje; su uso es reciente por lo que hay poca experiencia y bajo manejo del concepto por los involucrados en el proyecto.

4. DESARROLLO DEL TRABAJO DE GRADO

4.1 DISEÑO DE PROCESOS AUTOMATIZADOS PARA LA SIMULACIÓN DE USUARIOS Y/O PROCESOS PROGRAMADOS.

4.1.1 Definición de la ruta crítica de los casos prueba que serán automatizados.

Se debe tener identificado cuales son los casos de prueba que harán parte del procesos de automatización. Teniendo en cuenta que el alcance del proyecto se da sobre la ruta crítica, la cual fue definida previamente por el equipo de pruebas funcionales de TCS y el equipo de certificación y usuario del cliente bancario, se realizó un análisis a los casos de prueba definidos como ruta crítica para identificar cuáles son candidatos para automatización y cuales no lo son, entre los criterios sobre los cuales se determinó esta caracterización esta la cardinalidad esperada, nivel de complejidad, criticidad para el negocio, sistemas impactados y el criterio del automatizador adquirido con la experiencia.

Con base en lo anterior se definieron los siguientes casos de prueba para ser automatizados:

Tabla 1. Casos prueba automatizados.

Descripción	Cantidad	Porcentaje
Casos prueba funcionales	384	100%
Casos prueba Automatización	357	93%
Casos prueba NO Automatizados	27	7%

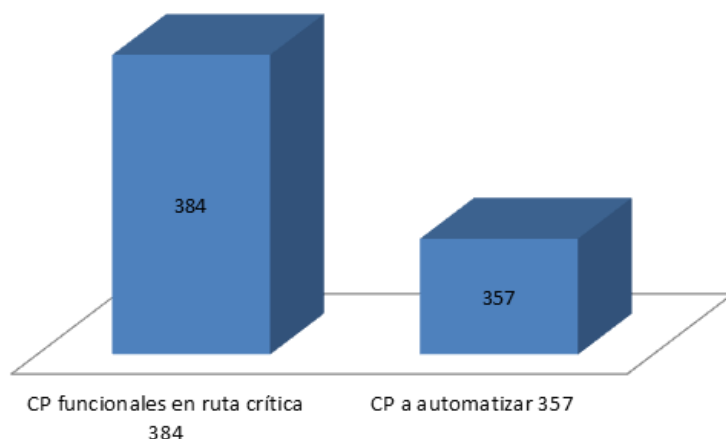


Figura 4. Barras casos prueba.

4.1.2 Solicitud accesos y permisos requeridos para la ejecución de los casos prueba.

La seguridad de la información es un tema extremadamente importante para TCS y se deben aplicar varios lineamientos en el manejo de cuentas y autorizaciones otorgados por los cliente. El cliente bancario tiene su área especializada en seguridad informática con sus propios lineamientos y protocolos de seguridad.

Previo al inicio de del diseño se realizó un inventario de la cantidad se usuarios y permisos requeridos en los casos de prueba, los cuales fueron aprobados, creados e implementados por las respectivas personas y áreas responsables.

Tabla2. Accesos y permisos requeridos.

Solicitud	Autorizador	Cantidad	Duración
Usuario en aplicativo	Gerencia de certificación cliente	3	6 meses
Usuario administrador	Gerencia de certificación cliente	1	6 meses
Autorización RW en repositorio 1	Gerencia de certificación Admin dominio	2	6 meses
Autorización RW en repositorio 2	Gerencia de certificación	2	6 meses
Autorización R en repositorio 3	Gerencia de certificación	2	6 meses
Autorización ruta compartida	Gerente proyecto cliente	2	6 meses
Autorización LOG 1	Gerencia de certificación	2	6 meses
Autorización LOG 2	Gerencia de certificación	2	6 meses
Autorizar MAC en RED	Gerencia de certificación Admin RED	1	6 meses
Usuario en QC	Gerencia de certificación	1	6 meses
Usuario BD local Access	Líder proyecto TCS	1	NA

4.1.3 Validación de adherencia para las herramientas de automatización.

Se grabó un flujo de proceso no diseñado con el objetivo de validar que no se presenten errores con la herramienta a nivel de tecnología usada y parametrización, la prueba de adherencia se realizó sobre los navegadores web internet Explorer y Chrome desde UFT 12.02, registrándose un comportamiento similar en ambos navegadores sin error.

4.1.4 Grabación de scripts.

En la grabación de los scripts se utilizó la herramienta UFT (UNIFIED FUNCTIONAL TESTING SOFTWARE) dispuesta por el cliente bancario quien dispone de las licencias, actualmente es una de las herramientas de automatización más completas y robustas que existen en el mercado.

La grabación de los script sobre el aplicativo bancario se realizó 100% web usando como navegador internet Explorer, permitiendo automatizar prueba a nivel de aplicación. El siguiente diagrama describe las bondades que ofrece esta poderosa herramienta.

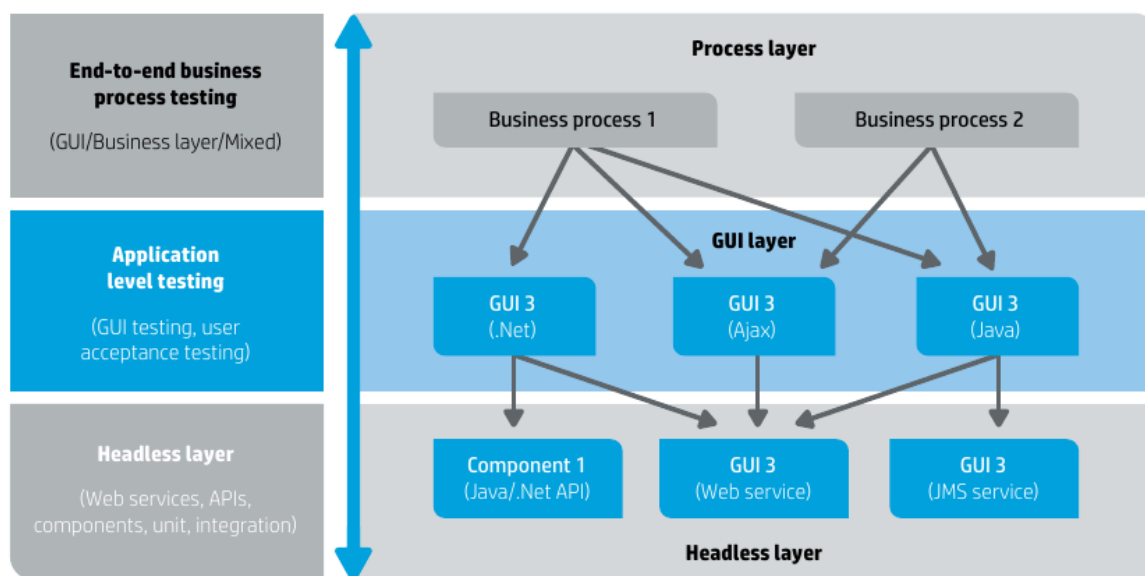


Figura 5.UFT.

Actualmente Tata Consultancy Services Colombia dispone de un framework de automatización propio desarrollado por el área, el cual a través de macros permite exteriorizar datos y usar pasos y palabras de acción predefinidas que al ser exportadas a una base de datos permitiendo integración con la herramienta UFT de Hewlett-packard, obteniéndose así una

arquitectura de automatización híbrida que permite una mayor estabilidad, reusabilidad y reducción de esfuerzo en los flujos de procesos automatizados.

Sobre la arquitectura definida se diseñó el 100% de los casos prueba seleccionados para ser automatizados, la automatización de los 357 flujos de proceso requirió un esfuerzo en diseño de 498 horas, dando un promedio de 1.4 horas por caso de prueba automatizado. Este bajo costo en esfuerzo se obtuvo gracias a la arquitectura de automatización implementada y a la gran experiencia del equipo de automatización.

Tabla3. Esfuerzo en diseño.

Esfuerzo diseño	Cantidad	Porcentaje
1 - 1.5 horas	157	44%
1.5 – 2 horas	149	42%
2 – 2.5 horas	35	10%
Más de 2.5 horas	16	4%

A continuación se presenta un ejemplo de script de un flujo automatizado, grabado y editado con la herramienta Silkperformer(Versión de prueba), el cual consiste en el logeo a la base de datos de la biblioteca virtual del Politécnico Colombiano Jaime Isaza Cadavid. El proceso completo de grabación, edición y ejecución del siguiente script se encuentra en el video adjunto del presente documento:

```
// -----
// Recorded 27/11/2015 by Silk Performer Recorder v15.0.1.6041
// -----
// Rule set "ASP.NET ViewState"
// * Rule "Parse and Replace __VIEWSTATE"
// * Rule "Parse and Replace __EVENTVALIDATION"
```

```

// -----
@codepage(1252)
benchmark SilkPerformerRecorder
use "Kernel.bdh"
use "BrowserAPI.bdh"
dcluser
user
VUser
transactions
TInit      : begin;
TMain      : 1;
var
dclrand
dcltrans
transaction TInit
begin
end TInit;
transaction TMain
var
wnd1 : number;
wnd2 : number;
begin
    BrowserStart(BROWSER_MODE_DEFAULT, 1040, 447);
    BrowserNavigate("http://www.politecnicojic.edu.co/index.php?option=com_content&view=article&i
d=250&Itemid=225",
    "Navigate_http://www.politecnicojic.edu.co/index.php?option=com_content&view=article&id=250&
Itemid=225 (#1)");
    wnd1 := BrowserGetActiveWindow("wnd1");
    wait 10.0;
    ThinkTime(14.6);
    BrowserClick("//IMG[@alt='Bases de datos']", BUTTON_Left, "Click, IMG (#1)");
    ThinkTime(4.7);
    wait 10.0;
    BrowserClick("//INPUT[@name='cedula']", BUTTON_Left);
    ThinkTime(6.3);
    wait 10.0;
    BrowserSetText("//INPUT[@name='cedula']", "8100971000");
    ThinkTime(2.5);
    wait 10.0;
    BrowserClick("//INPUT[@name='clave']", BUTTON_Left);
    ThinkTime(7.8);
    wait 10.0;
    BrowserSetPassword("//INPUT[@name='clave']",
    Decrypt3DES("CD85feOVfLK8A1armTKxSDg="));
    ThinkTime(2.1);
    wait 10.0;
    BrowserListBoxSelect("//SELECT[@name='tipousu']", "Seleccione uno");
    ThinkTime(3.5);
    wait 10.0;
    BrowserListBoxSelect("//SELECT[@name='tipousu']", "Estudiante pregrado");
    ThinkTime(11.8);
    wait 10.0;
    BrowserClick("//INPUT[@name='Submit']", BUTTON_Left, "Click, INPUT, name=Submit (#1)");
    ThinkTime(11.2);
    wait 10.0;

```

```

BrowserClick("//A[@textContents='lcontec']", BUTTON_Left, "Click, BODY (#1)");
wait 10.0;
wnd2 := BrowserGetActiveWindow("wnd2");
print("Fin de script");
end TMain;

```

A continuación se describen las funciones utilizadas en el anterior script:

- `BrowserStart(BROWSER_MODE_DEFAULT, 1040, 447);` // Se inicializa el navegador nativo de la herramienta.
- `BrowserNavigate("http://www.politecnicojic.edu.co/index.php?option=com_content&view=article&id=250&Itemid=225", "Navigate_http://www.politecnicojic.edu.co/index.php?option=com_content&view=article&id=250&Itemid=225 (#1)");` // Se carga la página de la biblioteca virtual del politécnico Colombiano Jaime Isaza Cadavid.
- `wnd1 := BrowserGetActiveWindow("wnd1");` //Se indica la ventana activa en el navegador.
- `wait 10.0;` // El script se detiene por diez segundos.
- `ThinkTime(14.6);` // Función que simula retardos de usuario según una distribución pre programada donde el promedio es 14.6 segundos.
- `BrowserClick("//IMG[@alt='Bases de datos']", BUTTON_Left, "Click, IMG (#1)");` // Clic sobre la imagen de base de datos.

- `BrowserSetText("//INPUT[@name='cedula']", "8100971000");` // Escribir número de cedula en el campo de texto cedula.
- `BrowserSetPassword("//INPUT[@name='clave']", Decrypt3DES("CD85feOVfLK8A1armTKxSDg="));` // Escribir password en el campo clave.
- `BrowserListBoxSelect("//SELECT[@name='tipousu']", "Estudiante pregrado");` // Seleccionar la opción de estudiante pregrado en la lista de texto.

4.1.5 Depuración, edición y parametrización de scripts.

Se realiza ajuste en los datos paramétricos requeridos por el esquema de automatización, los cuales son:

- Id script
- Id case
- Case name
- Test Case Description
- Name Suite
- Cycle
- Orientation of the case
- Id Step
- Step description
- Expected result
- Value

4.2 SIMULACIÓN DE USUARIOS O PROCESOS PROGRAMADOS DE FORMA AUTOMATIZADA SOBRE EL APLICATIVO BANCARIO.

4.2.1 Solicitud y coordinación con el cliente para la adecuación de data para las pruebas.

Se coordina con el cliente bancario la generación de la data necesaria en el ambiente de prueba para los casos de prueba automatizados, en esta actividad se encuentran involucrados el equipo de pruebas funcionales por parte de TCS, el equipo de certificación del cliente bancario, el equipo proveedor de desarrollo del aplicativo y el usuario.

En esta adecuación fue necesario enmascarar datos de ambiente productivo al ambiente de prueba, dado que es necesario garantizar la integridad en información consistente para evitar fallos por información corrupta en los procesos automatizados.

4.2.2 Realización de smoke test y validación del ambiente.

Se hace prueba de smoke test, la cual consiste en una pequeña ejecución para validar que todo el esquema de arquitectura de automatización, el ambiente de pruebas y el aplicativo se encuentran funcionales y estables y así tener un Go de inicio para la ejecución de los procesos automatizados. El resultado no exitoso de esta prueba de smoke genera un estado de stop en el proceso de pruebas lo que no permitiría iniciar con la ejecución de los casos de prueba automatizados.

La prueba de smoke se realizó correctamente, por lo que la ejecución de los casos prueba se inició correctamente según lo planeado.

4.2.3 Ejecución de robots.

La siguiente imagen muestra el resultado de ejecución para una suite de 47 casos de prueba ejecutados, 1 caso fallido en 59 minutos de ejecución.

Automated Test Results SET_PREVENTIVO001

Casos Ejecutados	Casos Exitosos	Casos Fallidos	Tiempo Promedio Caso (Seg)	Tiempo Ejecución (Mins)
47	46	1	71.468085106383	55.9833333333333

· [Funcionalidad FLUJOS](#)

Tata Consultancy Services All Rights Reserved

Figura 6. Resultado de ejecución.

La siguiente tabla describe los resultados en los diferentes bloques de ejecución realizados, estos se programaron de acuerdo a diferentes grupos de funcionalidades del aplicativo bancario, la tabla describe los bloques de ejecución éxitos, dado que algunos bloques se ejecutaron de forma no exitosa en varias ocasiones, estos fallos en los procesos automatizados se debieron a que el ambiente de prueba es un ambiente compartido con otros aplicativos y adicionalmente con otro tipos de prueba como pruebas de seguridad, pruebas de rendimiento y pruebas de usabilidad, en varias ocasiones la ejecución de este otro tipo de pruebas genero alteraciones en las parametrizaciones y data adecuada para los caso de prueba automatizados, lo que genero fallo en los mismos.

De lo anterior quedo como lecciones aprendidas para TCS y para el cliente bancario que es importante coordinar y programar ventanas de tiempo para la ejecución, parametrización y adecuación de la data para cada tipo de prueba o en lo posible disponer de un ambiente para cada tipo de prueba.

Tabla4. Resultados en ejecución.

Casos Ejecutados	Casos Exitosos	Casos fallidos	Tiempo promedio Seg	Tiempo Ejecución Mins
CP001 – CP047	46	1	71.4	55.9
CP047 – CP066	20	2	40.8	14.9
CP067 – CP104	38	0	80.5	51
CP104 – CP125	22	0	65.3	23.9
CP126 – CP152	27	1	103.2	48.1
CP153 – CP176	24	0	60.8	24.3
CP177 – CP200	24	5	133.2	64.3
CP201 - CP257	57	0	88.1	83.6
CP258 – CP305	48	3	77.0	65.4
CP306 - CP357	52	0	69.9	60.5

4.2.4 Validación de la correcta ejecución de los robots.

Al finalizar la ejecución de cada procesos automatizado se debe revisar si los errores o advertencias presentados son generados por fallos funcionales del aplicativo o se deben a fallos propios del esquema de automatización, si se identifica que los fallos presentados se deben al esquema de automatización se deben corregir y repetir la ejecución.

4.3 ANÁLISIS DEL COMPORTAMIENTO DEL APLICATIVO BANCARIO E IDENTIFICACIÓN DE FALLOS.

4.3.1 Análisis de los resultados recopilados en las ejecuciones.

La ejecución de un flujo de proceso automatizado o caso prueba automatizado puede finalizar con diversos resultados o estados según la definición o condición que se tiene programado como el resultado esperado. El analista de pruebas debe revisar los estados finales del caso prueba y clasificarlo.

Tabla5.Estados finales.

PASS	WARNING	FAIL
Caso de prueba exitoso.	El analista de prueba determina si el caso de prueba fue exitoso o fallido.	Caso de prueba fallido.

Se realizó un análisis a los 12 casos de prueba fallidos durante las ejecuciones clasificándolos de la siguiente manera:

Tabla6.Clasificación de Bugs.

Caso de prueba	Naturaleza	Prioridad	Genera Stoper	Reproducibile
CP022	Desarrollo	Muy Alta	Si	Si
CP053	Datos	Baja	No	Si
CP054	Datos	Baja	No	Si
CP150	Desarrollo	Alta	No	Si
CP179	Ambiente	Media	No	Si
CP180	Ambiente	Media	No	Si
CP181	Ambiente	Media	No	Si
CP190	Datos	Media	No	Si
CP191	Datos	Muy Alta	Si	Si
CP273	Desarrollo	Alta	No	Si
CP280	Desarrollo	Alta	No	Si
CP291	Desarrollo	Alta	No	Si

4.3.2 Validación con el equipo de trabajo sobre las consideraciones o fallos identificados.

Todos los fallos presentados en la ejecución de las pruebas se validaron con los equipos del proyecto involucrados, el equipo de pruebas funcionales, el equipo de certificación del lado cliente, el equipo de desarrollo del lado cliente y proveedor, en algunos fallos donde su naturaleza fue de ambiente con el área de infraestructura.

En el desarrollo del testing es de suma importancia validar los fallos y clasificarlos correctamente, ya que de lo contrario se pueden generar falsos positivos o retrasos en la solución de los mismos lo que afecta directamente los resultados del proyecto.

4.3.3 Reporte en el sistema o herramienta de pruebas los fallos evidenciados y remitiendo a quien corresponda para su solución.

El área de certificación en el cliente bancario dispone de una herramienta que le permite administrar y manejar el proceso de pruebas, permitiéndole llevar un seguimiento y control al proceso de certificación en los diferentes proyectos de TI y generar los reportes necesarios para indicadores y toma de decisiones.

Todo el proceso de prueba, diseño de los casos prueba, ejecución de los casos prueba, reporte de fallos con sus respectivas evidencias son registrados en la herramienta Quality Center de Hewlett-packard dispuesta por el cliente bancario.

La siguiente imagen contiene un ejemplo de un proyecto ficticio creado en la herramienta.

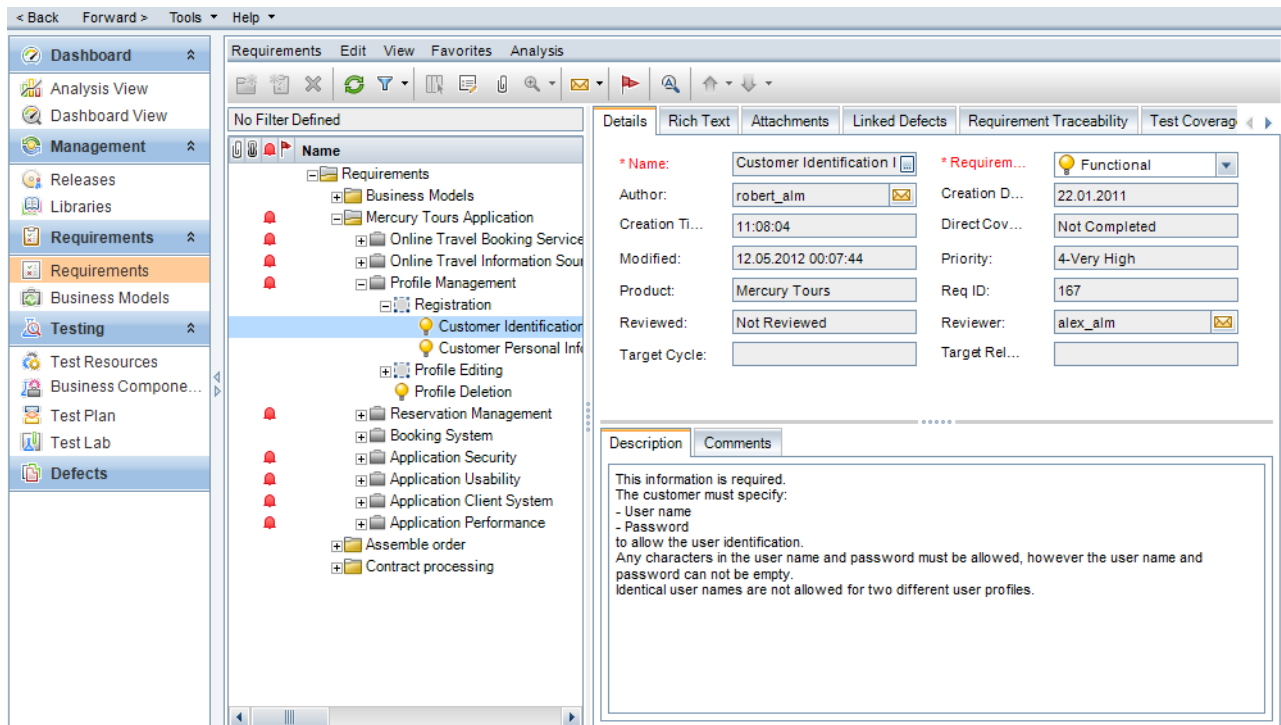


Figura 7. Quality Center.

4.3.4 Generación del informe sobre la eficiencia en la automatización de las pruebas.

Se realizó una presentación con el equipo de trabajo y el área de certificación del cliente bancario, donde se expusieron los resultados en las pruebas de regresión del primer ciclo de prueba con una automatización del 93% y una ejecución manual del 7%, donde las conclusiones más importantes fueron:

- Se obtuvo un 62% de reducción de horas de ejecución automatizada para los 357 CP, con respecto a la ejecución manual.
- Se obtuvo un 52% de reducción de horas de ejecución del set de regresión incluyendo los CP automatizados y manuales.

Tabla.7.Tiempo de pruebas.

Regresión	Horas
Ejecución manual 384 CP	132
Ejecución automatizada 357 CP	42
Ejecución manual 27 CP No automatizados	22
Ejecución Automatizada + manual	64

4.4 CRONOGRAMA 2015

Etapas	Actividades	May	Jun	Jul	Ago	Sep	Oct	Nov
1. Diseño de procesos automatizados para la simulación de usuarios y/o procesos programados.	1.1 Definición de la ruta critica de los casos prueba que serán automatizados.		■					
	1.2 Solicitud accesos y permisos requeridos para la ejecución de los casos prueba.		■					
	1.3 Validación de adherencia para las herramientas de automatización.		■					
	1.4 Grabación de scripts.				■			
	1.5 Depuración, edición y parametrización de scripts.				■			
2. Simulación de usuarios o procesos programados de forma automatizada sobre el aplicativo bancario.	2.1 Solicitud y coordinación con el cliente para la adecuación de data para las pruebas.					■		
	2.2 Realización de smoktest y validación del ambiente.					■		
	2.3 Ejecución de robots.					■		
	2.4 Validación de la correcta ejecución de los robots					■		
3. Análisis del comportamiento del aplicativo bancario e identificación de fallos.	3.1 Análisis de los resultados recopilados en las ejecuciones.					■		
	3.2 Validación con el equipo de trabajo sobre las consideraciones o fallos identificados.						■	
	3.3 Reporte en el sistema o herramienta de pruebas los fallos evidenciados y remitiendo a quien corresponda para su solución.						■	
	3.4 Generación de un informe consolidado sobre los errores reportados y los caso prueba ejecutados.							■
	3.5 Solicitud y recolección de los informes de avance generados por el equipo de pruebas funcionales en los primeros ciclos de pruebas.	■						
	3.6 Consolidación de la información de las ejecuciones manuales.		■					■
	3.7 Generación del informe sobre la eficiencia en la automatización de las pruebas.							■

Figura 8.Cronograma.

5. CONCLUSIONES

Con la automatización implementada en las pruebas de software se logró disminuir los tiempos altos invertidos en la ejecución de las pruebas funcionales, bajando el impacto en costos y facilitando que se logre cumplir oportunamente con los cronogramas de salidas a producción, generando una sensación de satisfacción en el cliente bancario y aportando a que la empresa continúe posicionándose como una de las más importantes en TI.

Una vez más se demostró la eficiencia y la ganancia que se obtiene al implementarse la automatización en pruebas de software, en el presente caso se logró disminuir los tiempos de ejecución de 384 casos de prueba que se realizaban de forma manual en 132 horas a ser ejecutadas con un 93% de automatización en 64 horas, lo que representó una reducción del 62% en los tiempos de ejecución.

El tester funcional actual ha visto y está consciente de la necesidad de implementar la automatización en sus pruebas funcionales, esto dada la tendencia actual de desarrollar proyectos bajo una metodología ágil que requiere ciclos de prueba cortos, haciendo necesaria la automatización.

Es necesario que la universidad en Colombia se prepare y prepare en la implementación de proyectos automatizados de pruebas, dada la tendencia mundial y la alta demanda creciente en el mercado con estos conocimientos de personas y empresas capaces de implementar este tipo de pruebas.

6. REFERENCIAS

1. Kim-Park, D. S., Riva, C. D. L., & Tuya, J. (2011). Aplicación de un oráculo de prueba automatizado a la evaluación de salidas de programas basados en XML. *Revista Española de Innovación, Calidad e Ingeniería del Software*, 7(1): 6-21. España.
2. Dapena, D., Dunia, M., Trujillo L., & Chávez, V. (2010). Propuesta de utilización del razonamiento basado en casos para la recuperación de procedimientos de prueba funcionales. *Revistas electrónicas UN > Ingeniería e Investigación*. Colombia.
3. Pantaleo, G. (2011). *Calidad en el desarrollo de software*. Alfaomega Grupo Editor. México.
4. International Software Testing Qualifications Board. (2014). <http://www.istqb.org/>
5. Cresswell, J. (2015). <https://github.com/cucumber/cucumber/wiki/Gherkin>
6. Cucumber Ltd. (2014). <https://cukes.info/>
7. Selenium. (2015). <http://docs.seleniumhq.org/>
8. Ministerio del interior. (2015). <http://www.derechodeautor.gov.co/>
9. TCS. (2015). <https://www.ultimatix.net>
10. Hewlett Packard (2014). HP Unified Functional Testing Software.
<http://www8.hp.com/us/en/software-solutions/unified-functional-automated-testing/>
11. Hewlett Packard (2014). HP Quality Center. <http://www8.hp.com/us/en/software-solutions/quality-center-quality-management/>

7. ANEXOS

El presente documento contiene anexo un video donde se presenta el proceso de grabación, edición, ejecución exitosa de un caso de prueba y ejecución fallida de un caso de prueba, este es un ejemplo de la forma como se automatiza un proceso de usuario, el cual consistió en un login sobre la base de datos en la biblioteca virtual del Politécnico Colombiano Jaime Isaza Cadavid.