

**Automatización del Proceso de Cargue de Información en la Aplicación Web
VARYS Mediante Cypress para Mejorar la Eficiencia Operativa en una
Organización en Bogotá**

Humberto José Ballesteros Mir

Tutor: Freskman Danilo Silva Marín

Programa de Ingeniería de Sistemas

Escuela de Ciencias Básicas Tecnología e Ingeniería

Universidad Nacional Abierta y a Distancia

Febrero de 2026

Tabla de Contenidos

Agradecimientos	8
Resumen.....	9
Abstract	11
Introducción	12
Líneas y Grupos de Interés Investigativo.....	14
Planteamiento del Problema	16
Árbol Causa – Efecto del Problema.....	20
Justificación	22
Delimitación del Proyecto:	25
Objetivos.....	27
Objetivo General	27
Objetivos Específicos.....	27
Marco Referencial.....	28
Marco Teórico:.....	28
Automatización como Mejora de Procesos.....	28
Importancia de Automatizar tareas Repetitivas	29
Cypress como Herramienta de Automatización	29
Pruebas Automatizadas y Eficiencia Operativa	30
Transformación Digital en el Entorno Laboral.....	30
Marco Conceptual:.....	31
Procesos Repetitivos:	31

	3
Cypress:.....	31
Script:	32
Aplicación VARYS:	32
Pruebas Funcionales Automatizadas:	32
Optimización:.....	32
Cierre de Bloque:	33
Marco Legal:	33
Legislación Colombiana	33
Ley 1581 de 2012 Protección de Datos Personales	33
Ley 527 de 1999 Comercio Electrónico y firma digital	33
Ley 1341 de 2009 Sociedad de la Información	34
Decreto 1377 de 2013 Reglamentación de la Ley de Protección de Datos	34
Regulaciones Internacionales.....	35
Reglamento General de Protección de Datos (GDPR)	35
ISOIEC 27001:2022 Seguridad de la Información.....	36
Convenio de Budapest sobre Ciberdelincuencia.....	36
Cumplimiento Legal de la Aplicación	36
Protección de Datos Personales (Ley 1581 de 2012 Colombia).....	37
Habeas Data (Artículo 15 de la Constitución Política de Colombia)	37
Responsabilidad en el Uso de Software (Ley 603 de 2000)	38
Cumplimiento Legal de Protección de Datos:	39
Diseño de Scripts con Buenas Prácticas:	39

	4
Control de Versiones:	39
Validación Constante de Resultados:.....	39
Documentación del Proceso:.....	39
Revisión Ética y Técnica:	40
Marco Tecnológico:	40
Metodología	44
Metodología de Investigación.....	44
Tipo de Investigación.....	44
Enfoque de Investigación.....	45
Diseño de la Investigación	45
Hipótesis de Investigación	46
Matriz de Trazabilidad Objetivo	46
Técnicas de Recolección de Información	48
Observación Estructurada	48
Cronometraje.....	49
Entrevistas Semiestructuradas	49
Revisión Documental.....	49
Instrumentos de recolección	49
Metodología de Desarrollo	51
Fases del Proyecto.....	52
Diagnóstico y Análisis de Requerimientos	52
Planificación y Diseño de la Solución	53

	5
Desarrollo del Script Automatizado.....	53
Pruebas, Validación y Medición.....	53
Evaluación de Resultados	54
Documentación y Cierre	54
Muestra y Población del Proyecto	56
Instrumento de Medición y Recolección de los Datos.....	58
Análisis y Diagnóstico del Proceso Investigativo.....	60
Metodología de Desarrollo de Software	65
Análisis de Requerimientos	69
Cronograma de Actividades.....	74
Recursos Necesarios para la Implementación.....	79
Diseño de la Solución	81
Análisis del Proceso Actual (AS-IS).....	81
Arquitectura de la Solución Automatizada (TO-BE).....	83
Metodología de Desarrollo Adoptada.....	84
Validación del Diseño.....	84
Análisis de Resultados	86
Conclusiones	91
Referencias.....	93
Anexos	100

Lista de Tablas

Tabla 1	Relación de intereses investigativos, líneas y grupos de investigación.....	14
Tabla 2	Diagnóstico inicial del proceso manual en la aplicación web VARYS.....	17
Tabla 3	Matriz de trazabilidad Objetivo.	47
Tabla 4	Cronograma de actividades a lo largo del ciclo de vida del proyecto.	74
Tabla 5	Presupuesto necesario para la implementación de la solución.	79
Tabla 6	Tabla de registros creados de forma manual y con la automatización.	88
Tabla 7	Tabla de promedio por ejecuciones de registros.....	88
Tabla 8	Comparación de errores entre el proceso manual y el automatizado.	89

Lista de Figuras

Figura 1	Árbol de problema.....	21
Figura 2	Diagrama de flujo del proceso manual (AS-IS).	82
Figura 3	Arquitectura de la solución automatizada.	83
Figura 4	Ejecución de cypress del aplicativo web.	100
Figura 5	Aplicación completando la información del formulario.	100
Figura 6	Prueba exitosa con Cypress tiempo de ejecución.....	101
Figura 7	Muestra de ejecución en cloud de cypress.	101
Figura 8	Visualización de los resultados en cloud cypress.....	102
Figura 9	Ejecución fallida en cloud cypress.	102
Figura 10	Ingreso de la información de forma manual con tiempo incluido.....	103
Figura 11	Validación de los tiempos en ejecución manual.....	104
Figura 12	Duración del registro de forma automatizada con Cypress.....	104
Figura 13	Formato de Observación Estructurada del Proceso Manual.....	105
Figura 14	Hoja de Registro de Cronometraje.	106
Figura 15	Guía de Entrevista Semiestructurada.....	106
Figura 16	Encuesta Estructurada de Percepción.....	107
Figura 17	Escala SUS (System Usability Scale).....	108

Agradecimientos

Quiero expresar mi más sincero agradecimiento a todas las personas que hicieron posible el desarrollo de este proyecto. En primer lugar, agradezco a Dios por brindarme la fortaleza, la disciplina y la perseverancia necesarias para culminar esta etapa académica. De manera especial, agradezco a mi director de proyecto Freskman Danilo Silva Marín por su orientación, acompañamiento constante y aportes académicos, los cuales fueron fundamentales para estructurar y desarrollar esta investigación con rigurosidad y claridad metodológica. También extendo mi agradecimiento a los docentes del programa, quienes a lo largo de mi formación académica compartieron sus conocimientos y experiencias, contribuyendo significativamente a mi crecimiento profesional y personal.

A los usuarios y colaboradores de la aplicación VARYS, gracias por su disposición y apoyo durante el proceso de diagnóstico y validación, ya que su participación permitió comprender el problema real y evaluar la efectividad de la solución propuesta.

Finalmente, agradezco profundamente a mi familia por su apoyo incondicional, comprensión y motivación constante durante este proceso. Su confianza fue un pilar fundamental para alcanzar esta meta. Este logro no solo representa el cierre de una etapa académica, sino también el inicio de nuevos retos profesionales, fortalecidos por el aprendizaje y la experiencia adquirida en el desarrollo de este proyecto.

Resumen

Este proyecto presenta el desarrollo e implementación de un script de automatización mediante la herramienta Cypress, con el propósito de optimizar los procedimientos repetitivos de cargue de información en la aplicación web VARYS, utilizada en una organización ubicada en Bogotá. Actualmente, este proceso se ejecuta de forma manual, lo que genera demoras operativas, errores humanos y una alta dependencia de la intervención directa del usuario. La investigación se desarrolló bajo un enfoque cuantitativo de tipo aplicado, con un diseño preexperimental orientado a la medición comparativa del proceso antes y después de la intervención tecnológica. Para el desarrollo de la Aplicación VARYS se adoptó la metodología ágil Scrum como marco de trabajo, estructurando el proyecto en fases de análisis del proceso manual, diseño de la solución, programación, pruebas e implementación progresiva. Las técnicas de recolección de información incluyeron observación directa, entrevistas semiestructuradas y cronometraje de tareas, lo que permitió establecer indicadores cuantificables de tiempo y frecuencia de error. Los resultados evidencian que el script automatizado replica de manera precisa las tareas de cargue de información, logrando una reducción del tiempo de ejecución superior al 90,8 % y eliminando errores de digitación asociados al proceso manual. Estos hallazgos demuestran que la automatización mejora la eficiencia operativa, fortalece la estandarización del proceso y aumenta la confiabilidad de los registros, constituyéndose como una solución viable y sostenible para la optimización de tareas repetitivas en entornos organizacionales.

Palabras clave: Automatización, calidad de software, eficiencia operativa, mejora de procesos, diseño experimental, Cypress.

Abstract

This project presents the development and implementation of an automation script using the Cypress tool, aimed at optimizing repetitive data entry procedures in the VARYS web application, used within an organization located in Bogotá. Currently, this process is performed manually, generating operational delays, human errors, and a high dependency on direct user intervention. The research was conducted under a quantitative applied approach with a pre-experimental design focused on the comparative measurement of the process before and after technological intervention. Scrum agile methodology was adopted as the development framework, structuring the project into phases including manual process analysis, solution design, programming, testing, and progressive implementation. Data collection techniques included direct observation, semi-structured interviews, and task time measurement, enabling the establishment of measurable indicators related to execution time and error frequency. The results show that the automated script accurately replicates the data entry tasks, achieving a reduction in execution time of more than 90.8% and eliminating typing errors associated with the manual process. These findings demonstrate that automation improves operational efficiency, strengthens process standardization, and increases record reliability, becoming a viable and sustainable solution for optimizing repetitive tasks in organizational environments.

Keywords: automation, Cypress, operational efficiency, pre-experimental design, software quality, Scrum.

Introducción

En la actualidad, las organizaciones dependen cada vez más de las aplicaciones web para gestionar sus procesos internos, atender a los clientes y mantener la competitividad en el mercado. Sin embargo, muchos de estos sistemas presentan limitaciones cuando se trata de procesos manuales repetitivos, lo cual incrementa la probabilidad de errores, afecta los tiempos de ejecución y reduce la eficiencia operativa de los equipos de trabajo.

Frente a este panorama, surge la necesidad de implementar soluciones tecnológicas orientadas a optimizar tareas rutinarias, reducir la intervención manual y mejorar la calidad del proceso. El presente proyecto se centra en el desarrollo de un script de pruebas automatizadas utilizando la herramienta Cypress, aplicado a la aplicación web VARYS, una plataforma utilizada en Bogotá para el manejo de información en entornos organizacionales. La elección de esta herramienta responde a su adopción en el ámbito de automatización de pruebas para aplicaciones web y a su capacidad para simular de manera controlada las acciones de un usuario dentro del sistema, permitiendo validar flujos funcionales de forma estructurada.

La relevancia del proyecto radica en analizar el impacto que puede generar la implementación de una solución automatizada sobre un proceso manual específico. La automatización del cargue de información en VARYS busca reducir tiempos de ejecución, disminuir errores asociados a digitación y estandarizar el procedimiento, contribuyendo a la mejora de la eficiencia operativa. Además, el desarrollo realizado establece una base técnica que podría facilitar futuras iniciativas de automatización en otros procesos internos de la organización.

El documento está estructurado de manera que se pueda comprender la totalidad del proceso investigativo y de desarrollo. Se incluyen apartados como el planteamiento del problema, la justificación, los objetivos, el marco teórico, la metodología, el diseño de la solución y las conclusiones. Todo ello enmarcado en las normas de investigación de la ingeniería de sistemas y con el soporte de referencias académicas y legales pertinentes.

En resumen, este proyecto pretende aportar a la transformación digital de una organización local mediante una propuesta aplicable, viable y escalable. Cypress será la herramienta que servirá de puente entre la teoría y la práctica, mostrando cómo la automatización no solo responde a una necesidad técnica, sino que también contribuye a la mejora del trabajo humano y al desarrollo empresarial en un entorno cada vez más competitivo.

Líneas y Grupos de Interés Investigativo

Tabla 1

Relación de intereses investigativos, líneas y grupos de investigación.

Intereses en Ingeniería e Investigación	Línea de Investigación y Áreas temáticas	Grupo de Investigación
Implementación de sistemas de geolocalización para prevención de amenazas digitales	Ingeniería de Software y Gestión de Sistemas	DAVINCI
Aplicación de inteligencia artificial en la detección de patrones sospechosos en plataformas en línea	Ingeniería de Software y Gestión de Sistemas	GIDESTEC
Integración de tecnologías móviles con sistemas de análisis de datos en tiempo real	Gestión de sistemas	BYTE IN DESING

Diseño de arquitecturas seguras para aplicaciones orientadas a la protección del usuario	Ingeniería de Software y Gestión de Sistemas	GUANE
--	--	-------

Evaluación del rendimiento de sistemas preventivos frente a ataques digitales	Ingeniería de Software y Gestión de Sistemas	DAVINCI
---	--	---------

Nota. Elaboración propia con base en Universidad Nacional Abierta y a Distancia (s.f.), Investigación y productividad ECBTI.

Planteamiento del Problema

En el contexto actual de transformación digital, las organizaciones dependen cada vez más de aplicaciones web para la gestión de información y la ejecución de procesos administrativos. Sin embargo, la calidad del software no se limita únicamente a su funcionalidad, sino que también involucra atributos como eficiencia en el desempeño y fiabilidad. De acuerdo con la norma ISO/IEC 25010 (2023), un sistema de calidad debe minimizar errores y optimizar el uso de recursos durante su operación. Cuando los procesos dependen excesivamente de la intervención humana, aumenta la probabilidad de fallos y se reduce la eficiencia operativa.

En la aplicación web VARYS, utilizada por una organización ubicada en Bogotá, el proceso de cargue de información se realiza manualmente. Los trabajadores deben ingresar y validar datos de forma repetitiva, lo que genera una carga operativa constante y exposición a errores de digitación. Según los lineamientos del ISTQB (2023), la automatización de procesos repetitivos contribuye a mejorar la confiabilidad y reducir defectos en sistemas informáticos.

Con el fin de establecer una línea base objetiva del problema, se realizó una observación directa del proceso, complementada con revisión de registros internos y verificación de datos. Los resultados obtenidos se presentan en la Tabla 2.

Tabla 2

Diagnóstico inicial del proceso manual en la aplicación web VARYS.

Indicador Observado	Descripción	Fuente de Observación
Tiempo promedio por registro manual	3 a 5 minutos	Observación Directa
Registros realizados por empleado/día	10 a 15	Bitácora Interna
Errores promedio por cada 10 registros	2 a 3	Verificación de datos
Número de empleados internos involucrados	20 usuarios internos	Área Administrativa
Número de empleados externos involucrados	30 usuarios externos	Colaboradores

Nota. Elaboración propia. Validación de los procesos manuales en el formulario de información de VARYS

Como se evidencia en la Tabla 2, el proceso manual implica un tiempo significativo por registro y una frecuencia relevante de errores. Considerando que entre 30 y 50 empleados realizan diariamente entre 10 y 15 registros, el impacto acumulado en términos de tiempo operativo y probabilidad de inconsistencias es considerable para el área administrativa.

Las principales causas que originan esta problemática son:

- La falta de mecanismos de automatización en el flujo operativo del sistema.
- La dependencia excesiva de intervención humana en tareas críticas de digitación y validación.
- La ausencia de herramientas accesibles de automatización integradas al proceso.

Estas causas generan consecuencias como aumento en los tiempos de ejecución, reprocesos y riesgo de inconsistencias en la información almacenada.

Si esta situación no se interviene, el crecimiento en el volumen de datos incrementará proporcionalmente los tiempos operativos y la probabilidad de errores. A mediano y largo plazo, esto puede traducirse en mayores costos administrativos, disminución en la calidad de los datos y limitaciones para avanzar en procesos de modernización y transformación digital dentro de la organización. Desde la perspectiva de la ingeniería de software, la automatización contribuye a mejorar la consistencia y sostenibilidad de los sistemas (Pressman & Maxim, 2019).

En este contexto, herramientas modernas de automatización como Cypress representan una alternativa viable para optimizar procesos en aplicaciones web. (Bhimanapati, Chhapola y Jain 2023) señalan que las estrategias actuales de automatización deben ser accesibles e intuitivas, favoreciendo su adopción en entornos organizacionales. Asimismo, el uso de marcos ágiles como Scrum permite implementar soluciones tecnológicas de manera incremental y controlada (Schwaber & Sutherland, 2020).

Por lo anterior, surge la siguiente pregunta de investigación:

¿Cómo puede el desarrollo de un script de automatización con Cypress optimizar el proceso manual de cargue de información en la aplicación web VARYS, reduciendo errores y mejorando la eficiencia operativa dentro de una organización local en Bogotá?

El presente estudio se delimita al análisis, diseño, desarrollo e implementación de un script automatizado orientado exclusivamente al proceso de cargue de información en la aplicación web VARYS. La investigación evalúa indicadores como tiempo promedio de ejecución, número de errores por registro y porcentaje de reducción de tareas manuales, sin intervenir la arquitectura interna del sistema ni otros módulos funcionales de la plataforma.

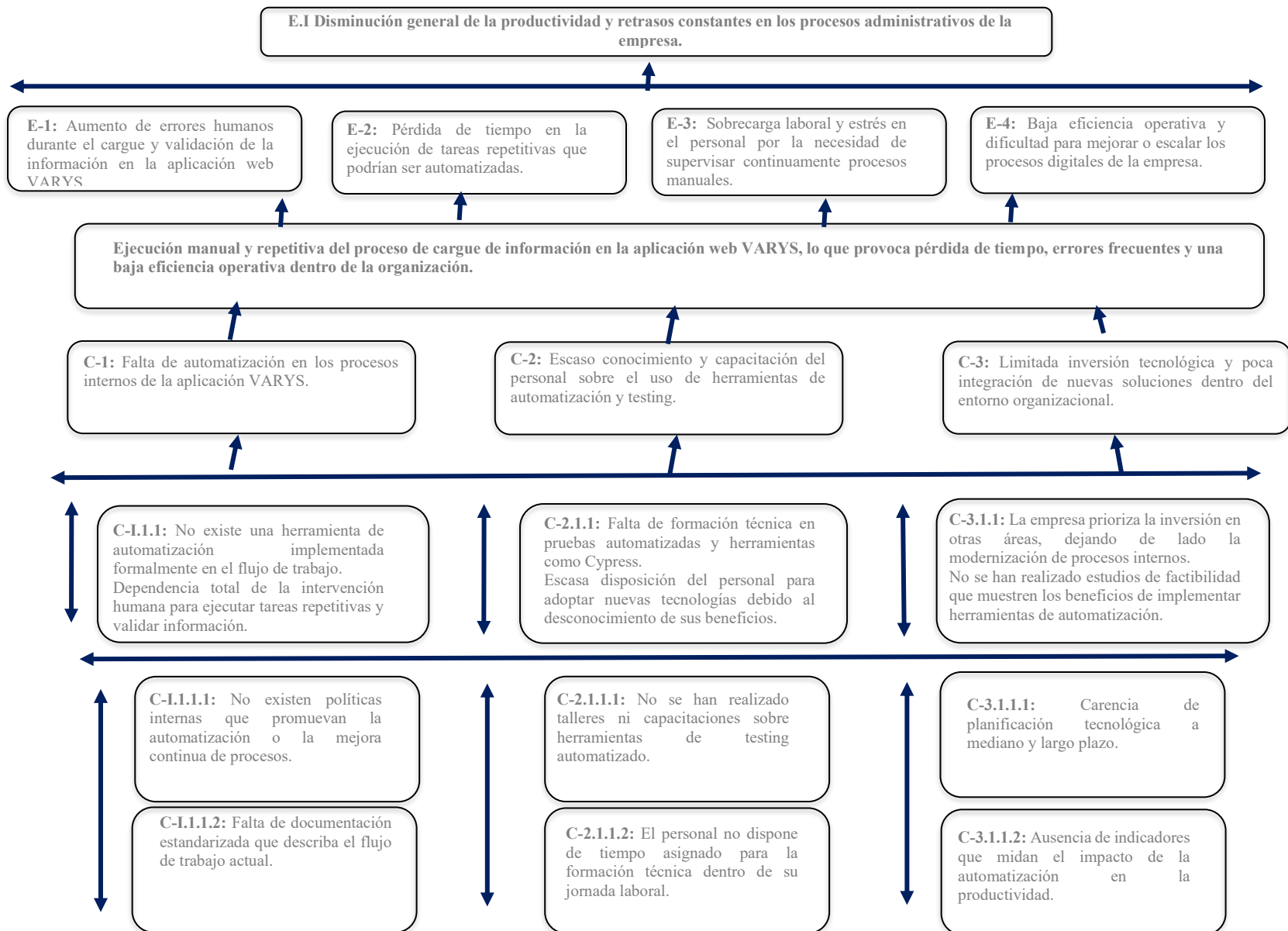
Árbol Causa – Efecto del Problema

El árbol de causa–efecto representa la relación estructural entre las causas directas e indirectas asociadas al problema principal, así como sus efectos sobre el funcionamiento organizacional. El problema se formula como una situación operativa identificada en el proceso de cargue manual de información, la cual impacta los indicadores de eficiencia y rendimiento.

Esta situación puede abordarse mediante la aplicación de soluciones técnicas, como el desarrollo de scripts automatizados con Cypress para reducir tareas repetitivas, la capacitación del personal en herramientas de automatización y testing, y la implementación de estrategias internas de mejora continua. Con ello, se espera optimizar los tiempos de ejecución, mejorar la precisión del proceso y fortalecer su mantenibilidad en el tiempo

Figura 1

Árbol de problema



Justificación

En la aplicación web VARYS, el proceso de cargue de información se ejecuta de manera manual, implicando la repetición constante de acciones dentro del sistema que demandan tiempo, concentración y precisión por parte del usuario. Esta dinámica genera dependencia del factor humano y aumenta la probabilidad de errores operativos, afectando la eficiencia del procedimiento. Si esta situación no se interviene, el impacto puede intensificarse a medida que aumente el volumen de información, comprometiendo la escalabilidad del proceso, la sostenibilidad operativa y la confiabilidad de los datos gestionados.

Desde la perspectiva de mejora de procesos, Syed et al. (2020) señalan que la automatización de procesos contribuye a la reducción de errores operativos, mejora la eficiencia organizacional y permite estandarizar actividades repetitivas dentro de los sistemas de información. En este contexto, la automatización se convierte en una estrategia orientada a eliminar actividades manuales que consumen recursos y generan variabilidad en los resultados. En este mismo sentido, los enfoques contemporáneos de gestión de procesos promueven la estandarización y el monitoreo continuo como mecanismos para garantizar calidad y consistencia.

En el campo de la ingeniería de software, la calidad no se limita al producto final, sino que involucra la mejora continua de los procesos asociados al desarrollo y uso del sistema (Pressman & Maxim, 2019). Sommerville (2016) plantea que el aseguramiento de la calidad del software (Software Quality Assurance – SQA) incluye prácticas orientadas a prevenir defectos y verificar sistemáticamente el funcionamiento correcto de

los sistemas. Dentro de estas prácticas, la automatización de pruebas y tareas repetitivas cumple un papel fundamental al permitir ejecuciones controladas, repetibles y medibles.

El testing automatizado, como parte de las buenas prácticas de calidad, busca ejecutar acciones previamente definidas de manera estandarizada, disminuyendo la variabilidad en la ejecución de procesos y facilitando la detección temprana de fallos (Pressman & Maxim, 2019). En este contexto, el uso de Cypress se enmarca como una herramienta técnica que permite aplicar principios formales de automatización, validación y control dentro del enfoque de la ingeniería de software, y no como un argumento independiente.

Metodológicamente, el proyecto se desarrolla bajo el enfoque de investigación aplicada, cuyo propósito es resolver problemas concretos en contextos reales mediante la implementación de soluciones prácticas (Hernández Sampieri et al., 2014). La automatización propuesta permite evaluar de manera empírica variables como tiempos de ejecución y reducción de errores, fortaleciendo el carácter cuantitativo del estudio.

Los principales beneficiarios directos son los usuarios responsables del cargue de información en la aplicación VARYS, quienes experimentarán una disminución en la carga operativa y en la probabilidad de cometer errores asociados al proceso manual. Como beneficiarios indirectos se encuentran las áreas administrativas y de gestión que dependen de la información registrada, al contar con datos más consistentes y oportunos para la toma de decisiones. El cambio esperado consiste en la transición de un procedimiento manual dependiente del factor humano a un proceso automatizado, estandarizado y medible, orientado a la reducción de tiempos y errores operativos.

Desde el punto de vista institucional, la automatización del proceso contribuye al fortalecimiento de la productividad organizacional al optimizar el uso del recurso humano y reducir la dependencia de tareas repetitivas. Asimismo, favorece la competitividad al mejorar la gestión de la información y promover prácticas de mejora continua dentro del entorno tecnológico. En términos de sostenibilidad, la estandarización del procedimiento automatizado facilita su replicabilidad ante posibles incrementos en la demanda, asegurando estabilidad operativa a mediano y largo plazo sin requerir inversiones complejas en infraestructura.

El proyecto se fundamenta en principios de ingeniería de software, mejora de procesos y aseguramiento de la calidad, integrando un enfoque metodológico y técnico orientado a la optimización de un proceso organizacional específico.

Delimitación del Proyecto:

Este proyecto se desarrollará en la ciudad de Bogotá y se enfocará específicamente en una aplicación web llamada VARYS, la cual es utilizada por una empresa para realizar procesos de cargue y manejo de información. El trabajo estará limitado a mejorar un proceso ya existente dentro de esta plataforma: el ingreso repetitivo de datos que los trabajadores deben hacer de forma manual todos los días.

La idea principal no es rediseñar toda la aplicación ni crear nuevas funciones para el sistema, sino automatizar una tarea puntual y repetitiva que actualmente toma mucho tiempo y es propensa a errores humanos. Para lograr esto, se utilizará la herramienta Cypress, que permite realizar y ejecutar scripts de prueba de forma automática en aplicaciones web.

La automatización se centrará únicamente en un flujo específico dentro de VARYS relacionado con el cargue de información. No se abarcarán otros módulos del sistema ni se tratarán temas como seguridad, mantenimiento general del software o desarrollo de nuevas funcionalidades.

Además, el proyecto se llevará a cabo en un ambiente de pruebas o entorno controlado, no directamente sobre el sistema en producción. Esto permitirá hacer ajustes y pruebas sin afectar el funcionamiento normal de la empresa.

El alcance temporal del proyecto será de aproximadamente de doce meses, tiempo durante el cual se desarrollará el script, se harán las pruebas necesarias y se evaluará su efectividad. Se medirá el impacto en cuanto a reducción de errores, ahorro de tiempo y

facilidad de uso, pero no se profundizará en análisis económicos o financieros más allá de los beneficios evidentes de productividad.

En resumen, este proyecto se limita a crear, aplicar y evaluar un script automatizado con cypress para optimizar un proceso específico de una aplicación web usada en Bogotá, con el objetivo de mejorar la eficiencia del trabajo diario, sin necesidad de hacer grandes cambios técnicos o de inversión.

Objetivos

Objetivo General

Desarrollar un script de pruebas automatizadas mediante la herramienta Cypress para optimizar y estandarizar los procedimientos repetitivos de cargue de información en la aplicación web VARYS, mejorando la eficiencia operativa y reduciendo los errores humanos en los procesos administrativos de una organización en Bogotá.

Objetivos Específicos

Analizar los procesos manuales de cargue de información en la aplicación VARYS mediante observación directa y entrevistas, con el fin de identificar los puntos críticos y definir los indicadores de eficiencia a optimizar

Diseñar y desarrollar un script de automatización con Cypress que simule el comportamiento de un usuario real, buscando reducir los tiempos de ejecución y los errores durante el cargue de información, de manera que el proceso sea más eficiente y confiable.

Implementar y evaluar el script en un entorno de pruebas controlado, comparando los resultados con el proceso manual para medir el impacto en la productividad y la estandarización del procedimiento buscando su sostenibilidad a largo plazo.

Marco Referencial

Marco Teórico:

En este proyecto se plantea la automatización de un proceso repetitivo dentro de una empresa utilizando la herramienta Cypress. Para sustentar esta propuesta, es necesario revisar teorías, conceptos y fundamentos relacionados con la automatización en entornos laborales. A continuación, se presenta la base teórica que respalda el desarrollo del trabajo.

La automatización se ha convertido en una herramienta ampliamente utilizada para mejorar la eficiencia operativa y reducir la carga de trabajo manual. Según Parapat, Kusuma y Nur Majiid (2023), la implementación de pruebas automatizadas en entornos web disminuye los errores humanos y optimiza el tiempo y la consistencia en los resultados. Esta perspectiva respalda la aplicación de tecnologías como Cypress para ejecutar tareas repetitivas de manera controlada y precisa dentro de sistemas empresariales.

Automatización como Mejora de Procesos

La automatización no es un tema nuevo. Desde hace décadas, ha sido una estrategia para aumentar la productividad y mejorar la eficiencia de las organizaciones. Su objetivo es reemplazar tareas manuales, repetitivas y propensas al error por procesos automáticos que requieren poca intervención humana. Según IBM (s.f.), automatizar significa usar tecnologías que ejecutan tareas o procesos sin intervención humana constante, lo que no solo reduce errores, sino que también permite ahorrar tiempo y recursos.

Esto se relaciona con el enfoque de mejora continua en las empresas, ya que automatizar permite optimizar flujos de trabajo y liberar al personal para que pueda concentrarse en labores más estratégicas o creativas.

Importancia de Automatizar tareas Repetitivas

Las tareas repetitivas suelen ser necesarias en una empresa, pero no agregan tanto valor como otras que requieren análisis, creatividad o toma de decisiones. Por ejemplo, subir datos manualmente a una plataforma, llenar formularios o copiar información de un lugar a otro. Este tipo de tareas son ideales para ser automatizadas. Al automatizarlas, se reduce el riesgo de errores humanos y se mejora la calidad de los procesos.

Este tipo de trabajo repetitivo se puede hacer más eficiente gracias a herramientas como cypress, las cuales permiten crear scripts que repiten acciones de forma precisa y rápida (BrowserStack, 2025).

Cypress como Herramienta de Automatización

permite crear scripts para hacer pruebas automáticas sobre páginas web, Lo interesante es que no exige que el usuario sea un experto en programación, ya que cuenta con una interfaz visual amigable, con funciones para grabar y reproducir acciones. Esto hace que cualquier persona, con algo de entrenamiento, pueda empezar a automatizar sus tareas (Cypress documentation, s.f.).

En el proyecto, se va a usar cypress para simular el comportamiento de un usuario que carga información en una plataforma web llamada VARYS. El script creado con esta herramienta permitirá hacer esa tarea automáticamente, sin necesidad de intervención manual cada vez.

Pruebas Automatizadas y Eficiencia Operativa

Las pruebas automatizadas no solo ayudan a verificar que un sistema funcione correctamente, también permiten ahorrar tiempo, ya que se pueden repetir cuantas veces sea necesario sin esfuerzo adicional. Además, permiten detectar errores de forma rápida y hacer mejoras continuas en las aplicaciones.

IBM (2021) explica que los scripts automatizados simulan lo que haría un usuario normal y ayudan a comprobar que el sistema funcione como se espera. De esta manera, se asegura la calidad del software sin necesidad de hacer pruebas manuales cada vez que se actualiza el sistema.

Transformación Digital en el Entorno Laboral

La automatización forma parte de un proceso más amplio llamado transformación digital, en el que las organizaciones adoptan nuevas tecnologías para mejorar su funcionamiento. Automatizar un proceso, aunque sea pequeño, puede ser el primer paso hacia una cultura de mejora tecnológica.

Con la implementación del script en la plataforma VARYS, se busca demostrar que con herramientas accesibles como cypress, incluso los equipos que no tienen grandes conocimientos en programación pueden mejorar procesos, ahorrar tiempo y aumentar la calidad de su trabajo.

En conclusión, el uso de la automatización con herramientas como cypress se sustenta teóricamente en la necesidad de optimizar procesos repetitivos, reducir errores humanos y mejorar la eficiencia general dentro de una empresa. Este tipo de soluciones

permiten que la transformación digital sea más accesible para todos, independientemente del tamaño de la organización o del conocimiento técnico del equipo.

Marco Conceptual:

Para entender bien este proyecto, es necesario conocer algunos conceptos clave que ayudan a explicar de qué trata el trabajo y por qué es importante.

Automatización de procesos:

Automatizar procesos significa dejar que una máquina o programa haga tareas que antes hacía una persona. Estas tareas pueden ser simples o complejas, pero el objetivo siempre es que se hagan más rápido, con menos errores y sin necesidad de intervención constante del ser humano. La automatización ha sido muy útil para que las empresas mejoren su eficiencia, reduzcan costos y sean más competitivas. Según IBM, la automatización ayuda a que las empresas logren resultados con menor intervención humana, aumentando la productividad y la calidad del servicio (IBM, s.f.).

Procesos Repetitivos:

Se entienden como actividades operativas recurrentes y estandarizables, ejecutadas con alta frecuencia y baja variación. Su ejecución manual incrementa la carga operativa y la probabilidad de errores, por lo que son susceptibles de mejora mediante automatización.

Cypress:

Framework de automatización de pruebas para aplicaciones web, orientado principalmente a pruebas funcionales y de extremo a extremo (E2E). Permite definir y

ejecutar casos de prueba de manera repetible, registrar evidencia y facilitar la verificación sistemática del comportamiento de la aplicación (Cypress, s. f.).

Script:

Conjunto de instrucciones ejecutables que automatizan una secuencia de acciones previamente definida (por ejemplo, interacción con formularios, validación de campos y verificación de resultados). En este proyecto, los scripts permiten simular el comportamiento del usuario de forma controlada para asegurar la consistencia del proceso (IBM Docs, 2021).

Aplicación VARYS:

Aplicación web utilizada para el cargue y gestión de información dentro de procesos administrativos y operativos. El procedimiento actual presenta alta dependencia del diligenciamiento manual, lo que incrementa tiempos de ejecución y exposición a errores.

Pruebas Funcionales Automatizadas:

Pruebas orientadas a verificar el cumplimiento de requisitos funcionales mediante la ejecución automatizada de casos de prueba. Su automatización permite aumentar la repetibilidad, reducir la variabilidad de ejecución y facilitar la detección temprana de fallas ante cambios del sistema (BrowserStack, 2025).

Optimización:

Proceso de mejora orientado a incrementar la eficiencia y confiabilidad de un procedimiento, mediante la reducción de tiempos de ciclo, la disminución de errores y la estandarización de la ejecución.

Cierre de Bloque:

Estos conceptos proporcionan el marco conceptual para comprender el propósito de la automatización y su relación con la mejora de procesos y el aseguramiento de la calidad en entornos organizacionales.

Marco Legal:

Para desarrollar un proyecto que automatiza procesos usando herramientas tecnológicas como cypress, es importante tener en cuenta el marco legal que rige este tipo de actividades en Colombia. Esto asegura que el uso del software, los datos que se manejan y las acciones automatizadas cumplan con lo que exige la ley.

A continuación, se presentan las principales normas jurídicas que se relacionan directamente con el desarrollo de este proyecto.

Legislación Colombiana***Ley 1581 de 2012 Protección de Datos Personales***

Esta ley establece que toda persona tiene derecho a que su información personal sea tratada de forma segura y responsable. Si el proyecto automatiza tareas donde se usen o consulten datos de personas, como nombres, cédulas o correos, es obligatorio proteger esa información.

El uso de herramientas como cypress para automatizar tareas en una plataforma web debe cumplir con esta ley, asegurando que los datos personales no se usen sin permiso, no se filtren ni sean compartidos con terceros sin autorización.

Ley 527 de 1999 Comercio Electrónico y firma digital

Esta ley les da validez legal a los mensajes de datos, al comercio electrónico y al uso de firmas digitales. En otras palabras, reconoce que los procesos digitales, como las transacciones electrónicas y los sistemas automatizados, pueden tener efectos legales, siempre que se cumpla con ciertos requisitos de seguridad y autenticidad.

Dado que este proyecto implica la interacción de un script con una plataforma web (que puede incluir procesos similares a los de un trámite o una transacción), es fundamental considerar que estas acciones tienen implicaciones legales y deben ser confiables y verificables.

Ley 1341 de 2009 Sociedad de la Información

Esta ley promueve el desarrollo y uso responsable de las tecnologías de la información y la comunicación (TIC) en Colombia. Reconoce que el uso de herramientas tecnológicas como el software de automatización puede mejorar la productividad y eficiencia en las organizaciones.

Este proyecto, al contribuir a la modernización de procesos mediante tecnologías digitales, está alineado con los principios de esta ley, ya que busca que las TIC se usen para facilitar el trabajo, mejorar servicios y apoyar la transformación digital de las empresas.

Decreto 1377 de 2013 Reglamentación de la Ley de Protección de Datos

Este decreto complementa la Ley 1581 de 2012 y explica cómo deben recogerse, almacenarse y usarse los datos personales. Si en el proceso de automatización se llega a manejar información confidencial o privada, como los datos de usuarios dentro de una plataforma web, se deben seguir los lineamientos establecidos en este decreto.

Esto implica, por ejemplo, tener el consentimiento del titular de los datos y tomar medidas para garantizar la seguridad de la información durante su procesamiento automático.

Para desarrollar un proyecto de automatización como este es esencial conocer y aplicar las normas que protegen los datos personales, regulan el uso del software y respaldan legalmente los procesos digitales. Cumplir con estas leyes no solo evita problemas legales, sino que también muestra responsabilidad y compromiso con el buen uso de la tecnología.

Regulaciones Internacionales

Además de las leyes nacionales, es importante tener en cuenta algunas normas y marcos internacionales que regulan el uso de tecnologías, el tratamiento de datos personales y la automatización de procesos. Esto es clave sobre todo si en algún momento el sistema automatizado se llega a usar con usuarios o datos de otros países, o si se busca que cumpla con estándares de calidad reconocidos globalmente.

Reglamento General de Protección de Datos (GDPR)

El Reglamento General de Protección de Datos (GDPR) de la Unión Europea es una de las normativas más completas sobre el manejo de información personal. Aunque fue creado para los países de Europa, muchas empresas en todo el mundo lo han adoptado como referencia, ya que promueve el uso responsable de los datos personales, exige el consentimiento del usuario y establece medidas claras para proteger la información.

En este proyecto, aunque no se manejen datos altamente sensibles, es importante seguir el ejemplo del GDPR en cuanto a la protección y el tratamiento correcto de cualquier dato que se procese mediante automatización.

ISOIEC 27001:2022 Seguridad de la Información

La norma internacional ISOIEC 27001 proporciona un marco de trabajo para gestionar de forma segura la información. Aunque no es una ley, muchas empresas la aplican para proteger sus datos y prevenir riesgos tecnológicos. Esta norma promueve buenas prácticas en cuanto al manejo de archivos, acceso a sistemas y control de información, lo cual es muy útil cuando se trabaja con herramientas de automatización.

En el proyecto, adoptar algunos principios de esta norma ayuda a asegurar que el sistema automatizado sea confiable y esté protegido contra errores o accesos no deseados.

Convenio de Budapest sobre Ciberdelincuencia

El Convenio de Budapest, impulsado por el Consejo de Europa, fue el primer tratado internacional que abordó los delitos informáticos. Este acuerdo promueve la cooperación entre países para combatir el mal uso de tecnologías, como el acceso ilegal a sistemas, la manipulación de datos o el uso indebido de información digital.

Aunque Colombia no es parte oficial del convenio, muchas de sus recomendaciones han sido adoptadas como referencia. Es útil tenerlo presente en cualquier proyecto tecnológico para garantizar que se respeten principios básicos de seguridad y legalidad en el entorno digital.

Cumplimiento Legal de la Aplicación

Cuando una empresa decide automatizar procesos a través de herramientas tecnológicas, no solo debe pensar en la eficiencia o en los beneficios técnicos. También es muy importante asegurarse de que todo lo que se haga esté dentro del marco legal vigente. Esto garantiza que la organización respete los derechos de los usuarios, cumpla con la ley y evite problemas legales en el futuro.

Protección de Datos Personales (Ley 1581 de 2012 Colombia)

En Colombia, la Ley 1581 de 2012 establece normas claras para el tratamiento de datos personales. Cualquier sistema o herramienta que maneje información de personas debe protegerla y evitar que sea usada sin autorización. Aunque este proyecto de automatización no está directamente enfocado en manejar datos personales sensibles, es necesario aplicar los principios de esta ley, como el consentimiento, la confidencialidad y la seguridad de la información.

Esto significa que, si en algún momento el script automatizado accede o gestiona datos personales dentro de la aplicación web VARYS, se debe garantizar que ese proceso respete la ley

Habeas Data (Artículo 15 de la Constitución Política de Colombia)

El derecho al Habeas Data garantiza que todas las personas pueden conocer, actualizar y rectificar la información que se tenga sobre ellas en bases de datos o archivos. Esto obliga a cualquier sistema automatizado que maneje información personal a incluir mecanismos que permitan cumplir con este derecho.

En la aplicación de este proyecto, si en algún momento se manejan bases de datos de clientes, usuarios o empleados, el sistema debe poder garantizar que esas personas puedan ejercer su derecho al Habeas Data.

Responsabilidad en el Uso de Software (Ley 603 de 2000)

La Ley 603 de 2000 establece la obligación de las empresas de informar si utilizan software con licencia o de desarrollo propio. Esto es importante porque al usar herramientas como Cypress, se deben cumplir sus condiciones de uso. La empresa debe asegurarse de que la herramienta utilizada esté autorizada y no infrinja derechos de autor.

Este aspecto es clave cuando se implementan soluciones tecnológicas, ya que el uso indebido de licencias puede traer sanciones legales para la organización.

Cumplir con la ley no es solo una formalidad, sino una responsabilidad que debe acompañar cualquier proceso tecnológico. En este proyecto, aunque el objetivo principal es automatizar un proceso repetitivo, es fundamental que todas las acciones respeten las normativas vigentes relacionadas con el manejo de la información, el uso de software y los derechos fundamentales. Esto no solo evita problemas legales, sino que demuestra un compromiso ético por parte de la empresa con la seguridad y el respeto hacia las personas. Implementar las Siguietes Medidas:

Para garantizar que el desarrollo del script de pruebas automatizadas cumpla con los requerimientos técnicos, legales y funcionales de la aplicación web, es necesario implementar una serie de medidas que permitan mantener la calidad, la seguridad y la efectividad del proyecto. Estas medidas buscan tanto prevenir errores como optimizar los resultados obtenidos.

Cumplimiento Legal de Protección de Datos:

Se debe garantizar que todo el proceso de automatización respete la Ley 1581 de 2012 sobre protección de datos personales. Esto incluye evitar que los scripts capturen, almacenen o difundan información sensible sin el consentimiento de los usuarios.

Uso adecuado de herramientas:

Verificar que el uso de herramientas como cypress se haga bajo las condiciones permitidas por su licencia, evitando violaciones de propiedad intelectual, de acuerdo con la Ley 603 de 2000.

Diseño de Scripts con Buenas Prácticas:

Utilizar estructuras claras, organizadas y fáciles de mantener en el desarrollo de los scripts, aplicando buenas prácticas de codificación, como modularidad, reutilización de código y uso de comentarios descriptivos.

Control de Versiones:

Implementar un sistema de control de versiones como Git, para llevar un registro de los cambios en los scripts, facilitar el trabajo colaborativo y permitir la recuperación de versiones anteriores en caso de errores.

Validación Constante de Resultados:

Ejecutar pruebas periódicas con los scripts para verificar que los resultados obtenidos correspondan con los esperados. Esto ayudará a detectar fallas a tiempo y corregirlas sin afectar los procesos principales de la aplicación.

Documentación del Proceso:

Documentar cada paso del proyecto, incluyendo la lógica de los scripts, los casos de prueba, las herramientas utilizadas y los resultados esperados. Esta documentación servirá de guía para futuras actualizaciones o para que otros puedan continuar el trabajo.

Capacitación básica al equipo:

Realizar jornadas de capacitación básicas para que los miembros del equipo que interactúan con los resultados de las pruebas comprendan su funcionamiento y puedan interpretarlos correctamente.

Revisión Ética y Técnica:

Verificar que el proceso de automatización no afecte negativamente a los usuarios ni genere decisiones automáticas que vulneren sus derechos, revisando que los scripts no generen sesgos ni errores graves.

Marco Tecnológico:

El marco tecnológico de este proyecto se basa en la aplicación de herramientas modernas de automatización y desarrollo web orientadas a la mejora de procesos repetitivos dentro de entornos empresariales. La iniciativa propone la implementación de Cypress, un framework de pruebas automatizadas de código abierto diseñado para aplicaciones web modernas, con el fin de optimizar los procedimientos manuales de cargue de información en la aplicación VARYS, usada actualmente por la organización.

Cypress se ha convertido en una de las herramientas más utilizadas para la automatización de pruebas funcionales, debido a su arquitectura moderna y a que opera directamente en el navegador, lo que le permite interactuar con la aplicación tal como lo haría un usuario real (Cypress.io, 2024). A diferencia de otras plataformas como

Selenium, Cypress no requiere una configuración compleja ni el uso de drivers externos, lo que reduce considerablemente el tiempo de implementación y facilita su integración con herramientas de control de versiones como GitHub o con entornos de integración continua como Jenkins.

Estudios recientes respaldan la efectividad de este tipo de herramientas de automatización en entornos empresariales (Mahesh y Usha Sree 2024) destacan que, aunque Selenium sigue siendo una opción sólida y ampliamente utilizada, su mantenimiento puede volverse más complejo en proyectos dinámicos debido a su dependencia de múltiples controladores y configuraciones externas. En cambio, Cypress ofrece una estructura más moderna y eficiente, centrada en la experiencia del desarrollador y en la ejecución de pruebas en tiempo real. De manera similar, (Bhimanapati, Chhapola y Jain, 2023) subrayan que la automatización moderna debe priorizar la simplicidad y la adaptabilidad, características que Cypress incorpora al permitir un flujo continuo entre desarrollo, prueba e integración. Esta combinación de practicidad y rendimiento hace que sea una opción ideal para proyectos de optimización de procesos como el que plantea esta investigación.

El entorno tecnológico del proyecto está compuesto por un conjunto de herramientas y tecnologías que permiten desarrollar, probar y mantener el script de automatización. En primer lugar, se utiliza JavaScript como lenguaje de programación principal, ya que es el que emplea Cypress para definir los flujos de prueba y automatización. Este lenguaje es ampliamente conocido por su flexibilidad y compatibilidad con la mayoría de los navegadores web modernos. Además, se trabaja con

el entorno Node.js, necesario para la instalación y ejecución de Cypress, lo que proporciona una base sólida para ejecutar las pruebas en tiempo real.

En cuanto al control de versiones, se emplea Git junto con GitHub para almacenar el código fuente, registrar los cambios realizados y facilitar la colaboración entre los participantes del proyecto. Esto asegura la trazabilidad del desarrollo, permitiendo regresar a versiones anteriores o realizar pruebas paralelas sin afectar el código principal (Chacon & Straub, 2021). Asimismo, se planifica la posible integración del script con herramientas de integración continua (CI/CD), lo cual permitiría ejecutar las pruebas de manera automática cada vez que se realicen actualizaciones en el sistema.

Por otro lado, el entorno de pruebas estará configurado en sistemas operativos Windows 10/11, utilizando Google Chrome y Microsoft Edge como navegadores principales para garantizar la compatibilidad del script. También se empleará Visual Studio Code como editor de código por su facilidad de uso y sus extensiones que facilitan la depuración y documentación del proyecto.

Para la gestión del proceso de desarrollo, se aplica la metodología ágil Scrum, que permite dividir el trabajo en ciclos cortos llamados sprints. Esta metodología ofrece flexibilidad ante los cambios y facilita la entrega continua de avances funcionales, lo que es ideal para proyectos de automatización que requieren ajustes constantes según los resultados de las pruebas. Cada sprint incluye reuniones de planificación, desarrollo, revisión y retrospectiva, garantizando una mejora progresiva del script (Schwaber & Sutherland, 2020).

A nivel de seguridad, el proyecto implementa políticas básicas de protección de datos, cumpliendo con los lineamientos de la Ley 1581 de 2012 sobre protección de información personal en Colombia. Los datos utilizados durante las pruebas serán ficticios o anonimizados, asegurando la confidencialidad y el cumplimiento normativo.

La infraestructura tecnológica de la organización permite ejecutar este tipo de herramientas sin necesidad de grandes inversiones. Los equipos de cómputo actuales cuentan con procesadores Intel Core i5, 8 GB de RAM y 256 GB SSD, suficientes para correr Cypress de manera fluida. Al ser una herramienta gratuita y de código abierto, no implica costos adicionales en licencias, lo que mejora la viabilidad económica del proyecto.

En síntesis, el marco tecnológico combina tecnologías modernas, accesibles y confiables que permiten desarrollar un sistema de automatización funcional y adaptable. Cypress se selecciona debido a su simplicidad, rendimiento y capacidad de integración, mientras que el uso de metodologías ágiles y buenas prácticas de desarrollo favorece que la solución propuesta sea escalable y sostenible a largo plazo. Este conjunto tecnológico está orientado a abordar el problema actual del proceso manual de cargue de información en la aplicación VARYS y puede servir como base para futuras automatizaciones dentro de la organización.

Metodología

El presente marco metodológico describe la estructura investigativa que orienta el desarrollo del proyecto, el cual se enmarca en un enfoque cuantitativo, de carácter aplicado y con diseño preexperimental. La investigación busca evaluar el impacto de la automatización del proceso de cargue de información en la aplicación web VARYS mediante la medición de indicadores objetivos como tiempo de ejecución y tasa de error. Para la recolección de información se emplean técnicas como observación estructurada, cronometraje, entrevistas y encuestas de apoyo contextual. Paralelamente, el desarrollo del script automatizado se implementa bajo la metodología ágil Scrum, lo que permite organizar el trabajo en fases iterativas. De esta manera, la metodología de desarrollo y el diseño preexperimental se articulan para dar cumplimiento a los objetivos planteados, permitiendo tanto la construcción de la solución como la evaluación empírica de su impacto.

Metodología de Investigación

La metodología utilizada en este proyecto es de tipo cuantitativo, aplicada y con diseño preexperimental. Esto quiere decir que se busca aplicar conocimientos para resolver un problema real dentro de una empresa, utilizando datos que se pueden medir y comparar antes y después de la implementación de una solución tecnológica (Hernández Sampieri et al., 2014). Además, se busca que los resultados no solo reflejen una mejora inmediata, sino que también sirvan como base para futuras optimizaciones dentro del sistema.

Tipo de Investigación

Este proyecto corresponde a una investigación aplicada, ya que busca dar solución a un problema concreto dentro de un contexto organizacional específico mediante la implementación de una herramienta tecnológica. De acuerdo con (Hernández Sampieri y Mendoza Torres, 2018), la investigación aplicada se caracteriza por utilizar el conocimiento científico con el propósito de intervenir en situaciones particulares y generar soluciones prácticas sustentadas en evidencia empírica. En este sentido, la automatización del proceso manual en VARYS no solo analiza la problemática existente, sino que propone y evalúa una alternativa funcional que puede implementarse directamente en el entorno laboral.

Enfoque de Investigación

El estudio se desarrolla bajo un enfoque cuantitativo, ya que los resultados se fundamentan en la medición objetiva de variables como el tiempo promedio de ejecución del proceso manual, la reducción porcentual obtenida mediante la automatización y la disminución de errores operativos. Este enfoque permite realizar comparaciones verificables antes y después de la intervención, facilitando la toma de decisiones basada en datos empíricos (Hernández Sampieri et al., 2014).

De manera complementaria, se contemplan mediciones posteriores con el fin de evaluar la estabilidad y escalabilidad del script automatizado frente a posibles actualizaciones del sistema, manteniendo el análisis centrado en indicadores cuantificables

Diseño de la Investigación

El diseño que se usó es preexperimental, porque se trabaja con un solo grupo de prueba, al cual se le aplica una intervención y luego se miden los cambios. No se hace comparación con otro grupo, pero sí se analiza la diferencia entre el antes y el después de la intervención (Sampieri et al., 2014). Este enfoque es apropiado porque permite evaluar directamente el impacto real del script dentro del entorno de trabajo de la empresa.

Hipótesis de Investigación

H₁: La implementación de un script automatizado mediante Cypress en el proceso de cargue de información de la aplicación web VARYS reduce significativamente el tiempo promedio de ejecución y la tasa de errores operativos en comparación con el procedimiento manual, medido a través de indicadores cuantificables antes y después de la intervención.

H₀: La implementación del script automatizado no produce una reducción significativa en el tiempo promedio de ejecución ni en la tasa de errores operativos del proceso

Matriz de Trazabilidad Objetivo

Con el fin de mantener la coherencia entre los objetivos planteados, las técnicas aplicadas y los indicadores que permitirán medir los resultados del proyecto, se elaboró una matriz de trazabilidad. Esta herramienta facilita la conexión entre lo que se busca alcanzar, las estrategias empleadas para lograrlo y las evidencias que demostrarán su cumplimiento. Además, permite que el proceso de investigación mantenga un seguimiento estructurado y verificable, garantizando que cada objetivo tenga su respectiva técnica de recolección y su indicador de evaluación.

Tabla 3

Matriz de trazabilidad Objetivo.

Objetivo específico	técnica o Instrumento	Indicadores de Evaluación	Tipo de Dato
Analizar los procesos manuales de cargue de información en la aplicación VARYS mediante observación directa y entrevistas, con el fin de identificar los puntos críticos y definir los indicadores de eficiencia a optimizar	Observación directa del proceso de cargue de información. Entrevistas semiestructuradas con los trabajadores del área administrativa. Cronometraje de tiempos de ejecución.	<ul style="list-style-type: none"> ➤ Tiempo promedio de ejecución por registro. ➤ Número de errores cometidos por cada 10 registros. ➤ Frecuencia de tareas repetitivas. 	Cuantitativo y Cualitativo
Diseñar y desarrollar un script automatizado utilizando la herramienta Cypress que simule el comportamiento de un usuario real.	Metodología ágil Scrum aplicada al desarrollo del script. Revisión de código y pruebas funcionales. Registro de incidencias durante las pruebas.	<ul style="list-style-type: none"> ➤ Funcionalidad y estabilidad del script. ➤ Porcentaje de ejecución exitosa sin fallos. ➤ Nivel de cumplimiento de los requerimientos definidos. 	Cuantitativo
Implementar y evaluar el script en un entorno de pruebas controlado,	Ejecución de pruebas de campo en entorno de prueba.	<ul style="list-style-type: none"> ➤ Porcentaje de reducción del tiempo de ejecución. 	Cuantitativo y Cualitativo

comparando los resultados con el proceso manual para medir el impacto en la productividad y la estandarización del procedimiento buscando su sostenibilidad a largo plazo.	Cronometraje comparativo entre proceso manual y automatizado. Encuesta de percepción a los usuarios sobre facilidad de uso y efectividad.	<ul style="list-style-type: none"> ➤ Disminución en el número de errores por registro. ➤ Nivel de satisfacción del usuario con la automatización.
--	---	---

Nota. Elaboración propia. La matriz muestra la relación entre los objetivos del proyecto, las técnicas aplicadas y los indicadores usados para evaluar los resultados de la automatización con Cypress.

Esta matriz permite garantizar la trazabilidad del proyecto, asegurando que cada acción desarrollada responda directamente a los objetivos planteados. Además, ofrece una guía clara para la evaluación del impacto de la automatización en términos de eficiencia, reducción de errores y mejora en la productividad del proceso.

Técnicas de Recolección de Información

Para llevar a cabo este proyecto de automatización, se aplicaron varias técnicas que permitieron recolectar información clara, confiable y útil. Estas técnicas ayudaron a conocer a fondo el proceso actual, sus fallas y las oportunidades de mejora, con el propósito de construir una solución funcional y escalable.

Observación Estructurada

Se realizó una observación directa del proceso manual que actualmente se lleva a cabo en la aplicación VARYS. Esta técnica fue útil para entender paso a paso qué hacen los usuarios, cuánto tiempo tarda cada acción y en qué puntos se generan errores o demoras. La observación permitió tener un panorama real y detallado del problema.

Cronometraje

Se utilizó la técnica de cronometraje para medir cuánto tiempo toma el proceso manual en comparación con el proceso automatizado usando Cypress. Esto permitió obtener una medida precisa de la mejora en eficiencia, uno de los objetivos principales del proyecto (Hernández Sampieri et al., 2014).

Entrevistas Semiestructuradas

Se aplicaron entrevistas a algunos de los usuarios que ejecutan el proceso manual. Las preguntas eran abiertas y dirigidas, lo que permitió conocer su experiencia, las dificultades que enfrentan y sus ideas sobre cómo mejorar el proceso. Esta técnica es muy útil cuando se quiere obtener opiniones, pero también seguir una guía que permita comparar respuestas (Sampieri et al., 2014).

Revisión Documental

Se revisaron documentos internos y registros del sistema, como bitácoras de errores, reportes de tiempos de ejecución y manuales de usuario. Esta información permitió tener datos objetivos sobre el comportamiento del sistema y detectar patrones repetitivos que pueden ser automatizados.

Instrumentos de recolección

Para la aplicación de las técnicas mencionadas se utilizaron los siguientes instrumentos:

- Formato estructurado de observación para registrar tiempos, pasos y errores detectados durante la ejecución manual.
- Hoja de registro de cronometraje para medir el tiempo promedio del proceso en múltiples ejecuciones.
- Guía de entrevista semiestructurada con preguntas orientadas a identificar dificultades operativas.
- Encuesta estructurada de percepción aplicada a los usuarios.
- En caso de evaluar la usabilidad del sistema automatizado, se contempla la aplicación de la escala SUS (System Usability Scale), instrumento reconocido para medir percepción de usabilidad en sistemas tecnológicos.

Metodología de Desarrollo

Para el desarrollo del proyecto se adoptó una metodología ágil basada en Scrum, la cual permitió organizar el trabajo en ciclos iterativos (Sprints) y realizar entregas progresivas del script automatizado (Sutherland & Schwaber, 2020). El proyecto tuvo una duración total de doce meses, conforme al cronograma establecido. Las actividades se estructuraron en cinco fases generales (análisis, diseño, desarrollo, validación y cierre), las cuales integraron prácticas propias de Scrum como la planificación incremental, revisión continua de avances y ajustes progresivos del script en función de los resultados obtenidos.

Scrum se empleó como marco de trabajo para gestionar el desarrollo técnico de la solución. Paralelamente, el proyecto se enmarca en un diseño preexperimental, el cual permitió evaluar el impacto de la intervención tecnológica mediante la comparación de mediciones antes y después de la automatización.

Ambas metodologías no funcionan por separado, sino de manera articulada:

- Scrum organiza el proceso de construcción, ajuste y mejora del script automatizado.
- El diseño preexperimental permite medir los resultados obtenidos a partir de indicadores definidos previamente.

Los datos que alimentan la evaluación preexperimental se generan principalmente en dos momentos:

- Durante el diagnóstico inicial, cuando se registran los tiempos y errores del proceso manual.

- Durante la fase de pruebas y validación del script, donde se miden nuevamente esos mismos indicadores bajo el proceso automatizado.

De esta forma, la metodología ágil facilita el desarrollo técnico de la solución, mientras que el enfoque investigativo permite comprobar empíricamente si la automatización produce mejoras reales en eficiencia y reducción de errores.

Fases del Proyecto

El desarrollo del proyecto se estructuró en las siguientes fases, organizadas bajo el marco de trabajo Scrum

Diagnóstico y Análisis de Requerimientos

En esta fase se identificó el proceso susceptible de automatización dentro de la aplicación web VARYS mediante observación directa y entrevistas con los usuarios (Hernández Sampieri et al., 2014). Se evidenció que la tarea consumía tiempo considerable, generaba errores y requería supervisión constante. Posteriormente se realizó el levantamiento de requerimientos, clasificándolos en:

- Requerimientos funcionales: tareas que el script debía ejecutar (ingreso de datos, validaciones, confirmaciones del sistema).
- Requerimientos no funcionales: aspectos relacionados con rendimiento, estabilidad, compatibilidad y facilidad de mantenimiento (Sommerville, 2016).

Durante esta fase también se revisaron consideraciones legales relacionadas con el tratamiento de datos personales, verificando el cumplimiento de la Ley 1581 de 2012 (Congreso de Colombia, 2012). Se garantizó que los datos utilizados en las pruebas fueran anónimos.

Indicadores generados en esta fase:

- Tiempo promedio del proceso manual.
- Tasa de errores operativos inicial.

Estos datos constituyen la línea base del diseño preexperimental.

Planificación y Diseño de la Solución

Con base en los requerimientos identificados, se planificaron los Sprints priorizando las tareas más críticas y propensas a errores. Se definieron criterios de éxito y se diseñó el flujo lógico del script automatizado.

El diseño contempló la secuencia de acciones, validaciones del sistema, manejo de tiempos de espera y selección de elementos mediante selectores estables.

Se seleccionó Cypress como herramienta principal para el desarrollo del script, debido a su capacidad para automatizar pruebas en aplicaciones web de manera estructurada y controlada (Cypress, s.f.).

Desarrollo del Script Automatizado

Durante esta fase se programaron los pasos necesarios para automatizar el proceso manual dentro del entorno automatizado. Se aplicaron buenas prácticas como la modularización del código, uso de selectores estables y documentación interna del flujo automatizado.

Este enfoque responde a lo señalado por (Fadul e Ibrahim, 2025), quienes destacan la importancia de desarrollar scripts capaces de adaptarse a interfaces dinámicas sin comprometer su estabilidad.

Pruebas, Validación y Medición

Una vez desarrollado el script, se realizaron ejecuciones controladas para verificar su funcionamiento. En esta etapa se midieron nuevamente los indicadores definidos en la fase diagnóstica:

- Tiempo promedio de ejecución.
- Tasa de errores en el cargue de datos.
- Estabilidad del script en múltiples ejecuciones.

Estas mediciones constituyen el momento clave del diseño preexperimental, ya que permiten comparar el desempeño antes y después de la intervención tecnológica (Sommerville, 2016).

Evaluación de Resultados

Se analizaron los datos obtenidos con el fin de determinar si la automatización produjo mejoras significativas en eficiencia y reducción de errores. Además de los indicadores cuantitativos, se consideró la percepción de los usuarios respecto a la facilidad de uso y confiabilidad del nuevo proceso.

Esta fase permitió establecer conclusiones frente a la hipótesis planteada y evaluar la sostenibilidad de la solución en el tiempo.

Documentación y Cierre

Finalmente, se elaboró la documentación técnica y de usuario del script, incluyendo instrucciones de ejecución, posibles errores comunes y recomendaciones para futuras actualizaciones.

La documentación facilita el mantenimiento y la replicabilidad de la solución sin depender exclusivamente del desarrollador inicial.

Lenguaje de programación:

- JavaScript

Framework de automatización:

- Cypress

Muestra y Población del Proyecto

La población de este proyecto está conformada por los trabajadores que utilizan de forma constante la aplicación web VARYS, una herramienta destinada a la gestión y al cargue de información en los procesos administrativos y operativos de una organización ubicada en Bogotá. Estas personas son las encargadas de subir, validar y actualizar datos en la plataforma de manera manual, siguiendo un procedimiento que actualmente demanda tiempo y atención continua.

En total, la población está compuesta por 50 usuarios, de los cuales 20 son empleados internos que pertenecen directamente a la organización y usan la aplicación como parte de sus labores administrativas, y 30 son usuarios externos asociados a diferentes proyectos que también interactúan con el sistema para cargar y validar información. Aunque no se trata de una población muy numerosa, la frecuencia con la que estos usuarios repiten las mismas tareas convierte este entorno en un escenario pertinente para aplicar una estrategia de automatización.

Para el desarrollo del estudio se seleccionó una muestra conformada por 10 usuarios activos, elegidos por su experiencia en el manejo de la aplicación y por el tipo de actividades que realizan. Esta selección se hizo con base en la disponibilidad de los participantes y en su relación directa con el proceso que se busca automatizar. La muestra incluye tanto empleados internos como colaboradores externos, lo que permite tener una visión más completa de las diferentes formas en que se usa la aplicación en el día a día.

El principal criterio de selección fue la frecuencia de uso de la herramienta y el nivel de responsabilidad en el proceso de cargue de datos. Se dio prioridad a quienes

realizan tareas críticas de introducción, validación o revisión de información, ya que son quienes enfrentan de manera directa las limitaciones del proceso manual y pueden aportar información valiosa sobre los puntos que deben mejorarse. Además, se buscó incluir personas con diferentes niveles de dominio tecnológico, para evaluar si la automatización propuesta con Cypress beneficia tanto a usuarios con experiencia en herramientas digitales como a quienes tienen menos familiaridad con ellas.

El tamaño de la muestra se justifica por el tipo de investigación aplicada y el diseño preexperimental del proyecto, en el que lo más importante no es la cantidad de participantes, sino la observación detallada de los cambios antes y después de la intervención. En este caso, la intervención consiste en la implementación del script automatizado con Cypress, que reemplazará parcialmente las tareas manuales actuales.

Los datos recolectados a partir de esta muestra permitirán analizar variables cuantitativas y cualitativas, como el tiempo promedio de ejecución, la cantidad de errores detectados y el nivel de satisfacción de los usuarios antes y después de la automatización. Estos resultados servirán para determinar si la herramienta desarrollada realmente mejora la eficiencia del proceso y si puede aplicarse a otros módulos o áreas de la organización.

De manera complementaria, esta muestra también facilita evaluar el impacto humano de la automatización: cómo perciben los usuarios la herramienta, si consideran que facilita su trabajo y qué ajustes serían necesarios para integrarla en sus rutinas. Esta retroalimentación es fundamental, ya que el éxito del proyecto depende no solo del componente técnico, sino también de la aceptación y adaptación del personal al cambio tecnológico.

Instrumento de Medición y Recolección de los Datos

Para obtener información precisa, confiable y útil en este proyecto, se emplearon varios instrumentos diseñados específicamente para comparar el desempeño del proceso manual con el proceso automatizado mediante Cypress. La aplicación de estos instrumentos se realizó en dos momentos clave: antes de la implementación del script y después de su puesta en marcha, con el fin de medir los cambios en términos de eficiencia, precisión y satisfacción de los usuarios.

El primer instrumento fue una guía de observación estructurada, creada para registrar detalladamente cada paso del proceso manual que los usuarios realizan al cargar información en la aplicación VARYS. Esta guía permitió identificar los tiempos de ejecución, la frecuencia de errores, las pausas durante el trabajo y la interacción de los empleados con la interfaz del sistema. Se realizaron varias sesiones de observación con los integrantes de la muestra, lo que ayudó a detectar patrones de comportamiento y puntos críticos que podrían optimizarse con la automatización.

El segundo instrumento utilizado fue una hoja de cronometraje, la cual permitió medir el tiempo que tarda cada fase del proceso, tanto manual como automatizado. Este instrumento fue clave, ya que el ahorro de tiempo es uno de los principales indicadores para evaluar la eficiencia del proyecto. Los tiempos se registraron en segundos y minutos, abarcando desde la apertura de la aplicación hasta la confirmación del cargue exitoso. Posteriormente, estos datos fueron comparados con los resultados obtenidos al ejecutar el script en Cypress.

También se aplicó un cuestionario de percepción, dirigido a los usuarios seleccionados en la muestra. Este cuestionario se aplicó en dos momentos: antes y después de la automatización. Su objetivo fue conocer la opinión de los participantes sobre el proceso manual y luego evaluar su experiencia con la herramienta automatizada. Las respuestas permitieron medir aspectos como la facilidad de uso, la confianza en el sistema y la satisfacción general con la automatización.

Otro instrumento importante fue el registro de errores, en el cual se documentaron los distintos tipos de fallos detectados durante la carga manual y las pruebas automatizadas. Entre los errores más comunes se identificaron digitaciones incorrectas, omisiones de datos, cargas incompletas y fallas en las validaciones. Este registro fue esencial para comparar la cantidad y el tipo de errores antes y después de la implementación del script, evidenciando una mejora significativa con el uso de la automatización.

Adicionalmente, se analizaron los logs del sistema, tanto de la aplicación VARYS como de Cypress. Estos archivos registran automáticamente cada acción ejecutada, incluyendo tiempos de respuesta, validaciones, mensajes del sistema e interacciones con los formularios. Su análisis permitió verificar que las pruebas automatizadas siguieran correctamente el flujo programado, garantizando que los pasos se cumplieran sin intervención humana y sirviendo como evidencia técnica del comportamiento del sistema.

Toda la información recolectada se organizó en una base de análisis elaborada en hojas de cálculo, donde se aplicaron medidas estadísticas descriptivas como promedios y

porcentajes. Con estos datos se calcularon indicadores clave de rendimiento, como la reducción del tiempo de ejecución, la disminución de errores y el nivel de satisfacción del usuario. De esta manera, fue posible realizar una comparación clara entre el proceso manual y el automatizado, demostrando el impacto real de la automatización con Cypress.

Para garantizar la confiabilidad de los resultados, todas las pruebas se realizaron bajo condiciones controladas: en horario laboral regular, con los mismos equipos, acceso al mismo entorno de la aplicación y sin variaciones en la red o en la versión del sistema. Esto permitió mantener la consistencia de los datos y evitar factores externos que alteraran los resultados.

En general, la aplicación de estos instrumentos permitió obtener una base sólida de información que respalda el análisis y diagnóstico del proceso investigativo. Gracias a la combinación de técnicas cuantitativas y cualitativas, fue posible comprender no solo los aspectos técnicos del proceso, sino también la experiencia de los usuarios durante la automatización. Esta información se convirtió en el punto de partida para identificar las causas del problema, evaluar la efectividad del script desarrollado con Cypress y establecer las mejoras necesarias dentro del entorno laboral.

Análisis y Diagnóstico del Proceso Investigativo

El proceso investigativo de este proyecto inició con la identificación de una problemática real presente en el entorno laboral, específicamente en el uso de la aplicación web VARYS, una herramienta empleada para gestionar y cargar información dentro de los procesos administrativos y operativos de una organización en Bogotá. A

través del uso diario de esta aplicación, se observó que gran parte de las tareas eran repetitivas, manuales y dependientes de la atención constante de los trabajadores. Estas condiciones generaban demoras, errores frecuentes y un desgaste operativo significativo. Ante esta situación, surgió la necesidad de diseñar una solución tecnológica que optimizara estos procedimientos mediante la automatización con Cypress, una herramienta moderna enfocada en pruebas automáticas para entornos web.

Desde el inicio, el propósito del análisis fue comprender de manera profunda las causas que originaban la ineficiencia en el proceso y, al mismo tiempo, identificar los puntos clave donde la automatización podría tener mayor impacto. El enfoque de la investigación fue cuantitativo y aplicado, porque se buscó obtener datos medibles que permitieran comparar el desempeño del sistema antes y después de la intervención. Además, se trató de una investigación aplicada porque no se limitó a describir un problema, sino que propuso una solución práctica y verificable dentro del entorno laboral.

Durante la fase de diagnóstico, se desarrollaron diferentes actividades de observación directa y recolección de información. En primer lugar, se analizaron los pasos que los trabajadores seguían para realizar el cargue de datos en la aplicación. Este proceso se documentó con detalle, registrando tiempos, cantidad de errores y pausas durante la ejecución. Se observó que los usuarios dedicaban una parte considerable de su jornada a tareas repetitivas, como copiar y pegar información, llenar formularios y verificar manualmente los datos cargados. Este tipo de actividades, aunque necesarias, no aportaban valor adicional al proceso y generaban un uso ineficiente del tiempo laboral.

Otro aspecto que se identificó en esta fase fue la falta de estandarización en los procedimientos. Cada usuario realizaba las tareas a su manera, sin seguir un protocolo único ni un flujo de trabajo formalmente definido. Esto ocasionaba inconsistencias en los resultados, errores en la información almacenada y dificultades para rastrear las causas de los fallos. La ausencia de validaciones automáticas dentro de la aplicación hacía que los datos incorrectos se registraran sin control, afectando la confiabilidad del sistema y obligando a realizar reprocesos frecuentes.

A partir de estas observaciones, se establecieron tres causas principales del problema:

El tiempo excesivo requerido para completar las tareas de cargue manual.

La dependencia total del operador, que debía estar presente para ejecutar cada paso del proceso.

La ausencia de validaciones automáticas, lo que aumentaba la posibilidad de errores humanos y afectaba la integridad de los datos.

Con base en este diagnóstico inicial, se aplicaron entrevistas semiestructuradas a los trabajadores que utilizaban la aplicación con mayor frecuencia. Estas entrevistas tuvieron como objetivo conocer la percepción del personal sobre el proceso manual y su disposición ante la implementación de una herramienta automatizada. Los empleados coincidieron en que el trabajo manual era tedioso, repetitivo y requería mucha atención para evitar errores. Sin embargo, mostraron una actitud positiva frente a la posibilidad de contar con un sistema automatizado, siempre que este fuera fácil de usar y no afectara sus funciones principales.

Posteriormente, se llevó a cabo un análisis cuantitativo de tiempos y errores. Para ello, se midió cuánto tardaba en promedio cada trabajador en completar una carga de información y cuántos errores cometía durante el proceso. Se utilizaron cronómetros digitales y registros automáticos del sistema para garantizar la precisión de los datos. Los resultados mostraron que un proceso manual podía tardar entre tres y cinco minutos por registro, dependiendo de la complejidad de los datos, mientras que la proyección inicial del script automatizado reducía ese tiempo a menos de un minuto.

En cuanto a los errores, se detectó que el 50% de ellos estaban relacionados con la digitación incorrecta o con omisiones durante la validación final. Con la automatización, se estimó que este porcentaje se reduciría a menos del 10%, al eliminar la intervención directa del usuario en los pasos más propensos a fallar. Estos datos respaldaron la viabilidad técnica y operativa del proyecto, demostrando que la automatización podía mejorar de forma tangible el rendimiento del sistema.

Otro hallazgo relevante fue la ausencia de herramientas modernas de automatización dentro de la organización. Aunque en el mercado existen soluciones como Selenium o Playwright, la empresa no contaba con ninguna plataforma similar. Esto generó una oportunidad clara para introducir Cypress, una herramienta moderna, sencilla y adaptable, que no requiere un conocimiento avanzado en programación y permite obtener resultados precisos en poco tiempo.

Durante el diagnóstico también se evaluó la infraestructura tecnológica disponible. Se comprobó que la empresa contaba con los recursos necesarios para la implementación: equipos con buena capacidad de procesamiento, navegadores

compatibles con Cypress y conexión a internet estable. Por lo tanto, no se requería inversión adicional en hardware ni licencias costosas, lo que fortaleció aún más la viabilidad económica del proyecto.

En conjunto, el análisis permitió construir un panorama completo de la situación actual del proceso y justificar la necesidad de implementar una herramienta automatizada. Se concluyó que la falta de automatización no solo afectaba la productividad, sino también la motivación del personal, ya que muchas de las tareas resultaban monótonas y poco estimulantes. Con la introducción del script desarrollado en Cypress, se busca reducir la carga operativa, mejorar la precisión de los datos y liberar tiempo que los trabajadores puedan destinar a actividades de mayor valor para la empresa.

En resumen, el análisis y diagnóstico del proceso investigativo demostraron que la automatización representa una solución efectiva, viable y sostenible para optimizar los procesos repetitivos en la aplicación VARYS. La información obtenida en esta etapa no solo permitió validar la existencia del problema, sino que también sirvió de base para las siguientes fases del proyecto: el diseño, desarrollo, prueba y evaluación del script automatizado. Con estos resultados, se establecieron las condiciones necesarias para garantizar una implementación exitosa y una mejora real en la eficiencia operativa de la organización.

De acuerdo con Dubey (2025), la automatización inteligente es una herramienta transformadora en la gestión de calidad del software, ya que facilita la detección temprana de errores, mejora la trazabilidad y promueve una cultura de innovación tecnológica dentro de las organizaciones. En este proyecto, esa visión se ve reflejada en

los resultados obtenidos tras la implementación del script automatizado, donde la automatización no solo optimizó los tiempos de ejecución, sino que también fortaleció la fiabilidad del proceso y permitió establecer un modelo de mejora continua en la gestión operativa.

Metodología de Desarrollo de Software

La metodología de desarrollo seleccionada para este proyecto fue Scrum, un marco de trabajo ágil ampliamente utilizado en la ingeniería de software moderna. Esta elección se basó en la necesidad de contar con un enfoque flexible, adaptable y colaborativo que permitiera enfrentar los cambios que pudieran surgir durante el desarrollo del script de automatización con Cypress. Scrum se caracteriza por dividir el trabajo en ciclos cortos e iterativos llamados sprints, donde en cada uno se generan entregas funcionales y se revisan los resultados antes de avanzar a la siguiente etapa.

El uso de Scrum resultó ideal para este proyecto porque, al tratarse de una solución tecnológica aplicada a un entorno real de trabajo, era fundamental poder ajustar y mejorar el desarrollo constantemente según la retroalimentación obtenida durante las pruebas. Además, esta metodología promueve la comunicación activa entre los participantes, el aprendizaje continuo y la mejora progresiva del producto.

En el marco de este proyecto, Scrum se implementó a través de sus tres pilares principales: roles, eventos y artefactos. Cada uno cumplió una función específica para asegurar que el proceso se mantuviera organizado, controlado y enfocado en los objetivos establecidos. Los roles principales definidos fueron:

- El desarrollador, encargado de crear, codificar y ejecutar el script de automatización en Cypress. Este rol incluyó la programación de las funciones, la verificación del comportamiento del sistema y la corrección de los errores detectados durante las pruebas.
- El Product Owner, quien definió los requerimientos del proceso que debía automatizarse dentro de la aplicación VARYS, priorizando las tareas más importantes y garantizando que el producto final cumpliera con las necesidades de la empresa.
- El Scrum Máster, responsable de coordinar las actividades, eliminar obstáculos y asegurar que el equipo siguiera los principios de la metodología. Este rol también se encargó de supervisar las reuniones de revisión y planificación, promoviendo la colaboración entre los participantes.

El trabajo se organizó en sprints de dos semanas de duración, lo que permitió avanzar de forma ordenada y con objetivos concretos en cada etapa. En total, el proyecto se estructuró en cuatro sprints principales, que abarcaron desde el análisis inicial hasta la entrega final del script automatizado:

- Sprint 1: Análisis de requerimientos y documentación del proceso manual. En esta fase se identificaron las tareas repetitivas dentro de la aplicación VARYS, se levantó la información sobre los flujos de trabajo actuales y se elaboraron los diagramas de proceso. Además, se definieron los objetivos técnicos del script y los criterios de éxito que permitirían evaluar su efectividad.
- Sprint 2: Diseño y desarrollo inicial del script en Cypress.

Aquí se construyó la estructura base del script, se configuró el entorno de trabajo y se desarrollaron las primeras funciones automatizadas. También se documentó el código para asegurar su mantenibilidad y comprensión por otros miembros del equipo.

- Sprint 3: Pruebas controladas, corrección de errores y optimización.

Durante este sprint se realizaron pruebas exhaustivas del script en un entorno simulado, verificando su funcionamiento paso a paso. Los errores encontrados se corrigieron y se optimizó el código para mejorar la velocidad de ejecución y la estabilidad del sistema.

- Sprint 4: Evaluación de resultados y documentación final.

En la última fase, se realizó la comparación entre el proceso manual y el automatizado, midiendo el tiempo de ejecución, la reducción de errores y la satisfacción de los usuarios. Finalmente, se elaboró la documentación técnica y se preparó la versión final del proyecto para su entrega.

La aplicación de Scrum permitió que la retroalimentación continua fuera un componente esencial del proceso. Al final de cada sprint, se realizaron reuniones de revisión y retrospectiva, en las cuales se presentaron los avances, se discutieron los problemas encontrados y se propusieron mejoras para el siguiente ciclo. Estas sesiones fomentaron la comunicación abierta y ayudaron a mantener la alineación del equipo con los objetivos del proyecto.

Uno de los mayores beneficios de Scrum fue la posibilidad de integrar otras herramientas complementarias. Por ejemplo, el desarrollo del script en Cypress se gestionó a través de GitHub, donde se almacenó el código y se controlaron las versiones. Esto facilitó la colaboración entre los miembros del equipo y permitió tener un registro

claro de los cambios realizados. Además, se consideró la integración futura con sistemas de integración continua (CI/CD) como Jenkins, lo que permitiría ejecutar pruebas automatizadas de manera automática cada vez que se actualice el sistema VARYS, asegurando así una mejora continua.

Scrum también hizo posible establecer métricas claras de evaluación del avance y del rendimiento del equipo. Entre los indicadores más relevantes estuvieron el tiempo promedio de desarrollo por sprint, el número de errores corregidos, la cantidad de funcionalidades entregadas y la percepción de los usuarios tras probar la versión automatizada del proceso. Estos indicadores no solo ayudaron a medir la productividad del equipo, sino también la calidad del producto entregado.

Otro aspecto importante fue la capacidad de adaptación que ofrece Scrum. Durante el desarrollo, algunos requerimientos iniciales fueron ajustados para optimizar el desempeño del script y hacerlo más compatible con la infraestructura de la empresa. Gracias a la flexibilidad del marco ágil, estos cambios pudieron incorporarse sin afectar significativamente el cronograma ni la calidad del resultado final.

En el contexto del proyecto, Scrum demostró ser mucho más que una metodología de desarrollo; fue un modelo de trabajo colaborativo que permitió combinar la disciplina técnica con la creatividad y la mejora constante. Su enfoque iterativo garantizó que cada entrega parcial aportara valor real y que las decisiones se basaran en evidencia obtenida a partir de pruebas continuas.

En conclusión, la metodología de desarrollo de software Scrum fue la más adecuada para este proyecto porque permitió trabajar de manera ordenada, ágil y

enfocada en resultados concretos. Su estructura fomentó la comunicación constante, la retroalimentación continua y la entrega de versiones funcionales en cada etapa. Gracias a este enfoque, el proyecto logró alcanzar su objetivo en menos de cuatro meses, con una solución automatizada estable, confiable y adaptable a las necesidades futuras de la empresa. Además, la flexibilidad de Scrum deja abierta la posibilidad de ampliar el uso de la automatización hacia otros procesos dentro de la organización, fortaleciendo así la cultura de mejora continua y la eficiencia operativa.

Análisis de Requerimientos

El análisis de requerimientos fue una de las etapas más importantes del desarrollo del script de automatización, ya que permitió definir con claridad qué debía construirse, cómo debía funcionar y en qué condiciones debía operar. Este proceso fue esencial para asegurar que la solución realmente respondiera a las necesidades de los usuarios de la aplicación VARYS, evitando errores de interpretación y garantizando que el proyecto se desarrollara sobre una base sólida y bien estructurada.

Desde el inicio, se entendió que una buena definición de requerimientos no solo delimita el alcance del proyecto, sino que también previene retrabajos y facilita la validación final del producto. Por eso, el análisis se llevó a cabo de manera detallada y participativa, involucrando tanto al equipo de desarrollo como a los usuarios que diariamente utilizan el sistema.

El proceso comenzó con una serie de sesiones de observación directa y entrevistas semiestructuradas a los empleados encargados de realizar el proceso de cargue de información. Estas actividades permitieron identificar los pasos específicos que los

trabajadores ejecutaban de forma manual, los errores más comunes, los tiempos de respuesta y los puntos donde se presentaban mayores dificultades. A partir de esta información, se elaboró una lista inicial de procesos susceptibles de ser automatizados.

Posteriormente, esa lista fue revisada junto con el líder técnico y se procedió a clasificar los requerimientos en dos grandes categorías: funcionales y no funcionales, siguiendo las recomendaciones planteadas por Sommerville (2016) en su metodología de ingeniería de software.

Los requerimientos funcionales describen las acciones concretas que el sistema debía ejecutar, es decir, lo que el script debía ser capaz de hacer para cumplir con su propósito. En este caso, se establecieron los siguientes:

- Iniciar sesión en la aplicación VARYS utilizando credenciales seguras y validadas.
- Cargar automáticamente los datos en los formularios correspondientes, respetando el formato y estructura definidos por el sistema.
- Verificar que los campos obligatorios estuvieran diligenciados correctamente antes de enviar la información.
- Confirmar que el sistema emitiera mensajes de éxito o error según el resultado de la carga.

Generar un reporte automático con el detalle de cada ejecución, indicando los registros cargados, los errores detectados y el tiempo total del proceso.

Estos requerimientos funcionales definieron la esencia del script, ya que establecieron las operaciones mínimas que debía realizar para cumplir su objetivo

principal: automatizar el proceso manual de cargue de información en VARYS de forma segura, rápida y confiable.

Por otro lado, los requerimientos no funcionales especificaron las características de calidad y desempeño que debía tener la solución. Aunque no describen funciones concretas, sí determinan las condiciones bajo las cuales el sistema debe operar correctamente. En este proyecto, se definieron los siguientes:

- El tiempo total de ejecución del script no debía superar los 60 segundos por registro.
- Debía garantizar la compatibilidad con navegadores modernos, especialmente Google Chrome y Microsoft Edge.
- El código debía ser modular, limpio y fácil de mantener, de manera que cualquier otro desarrollador pudiera entenderlo y modificarlo sin dificultad.
- La automatización debía ejecutarse sin intervención humana una vez iniciada.
- Se debía asegurar la confidencialidad de los datos utilizados durante las pruebas.

Además de los requerimientos técnicos, se tuvieron en cuenta los aspectos legales y de seguridad, cumpliendo con lo establecido en la Ley 1581 de 2012 sobre la protección de datos personales en Colombia. Por esta razón, se decidió trabajar únicamente con datos ficticios o anonimizados durante las pruebas del script, evitando cualquier riesgo de exposición o mal uso de información sensible. Este enfoque no solo

garantizó el cumplimiento normativo, sino también la confianza del equipo y la organización en la implementación del proyecto.

Durante el análisis, también se identificaron riesgos potenciales que podrían afectar el desempeño del script una vez implementado. Entre ellos se destacaron los posibles cambios en la interfaz de la aplicación, actualizaciones en el sistema que alteraran las rutas de acceso, interrupciones del servidor y variaciones en los tiempos de respuesta del sitio. Para mitigar estos riesgos, se propuso desarrollar un código adaptable, con el uso de selectores dinámicos y funciones reutilizables, de modo que el script pudiera ajustarse fácilmente a modificaciones futuras sin requerir una reprogramación completa.

Otro aspecto clave dentro del análisis fue la definición de los criterios de aceptación. Estos criterios se utilizaron como guía para determinar cuándo un requerimiento se consideraba correctamente implementado

El análisis también sirvió para establecer una comunicación clara entre los involucrados, ya que tanto el equipo técnico como los usuarios compartían una visión común de lo que se esperaba del sistema. Esto evitó confusiones, duplicidad de esfuerzos y retrabajos en etapas posteriores.

Finalmente, el resultado de este análisis fue un conjunto sólido y bien definido de especificaciones técnicas y funcionales, que aseguraron que el desarrollo del script cumpliera con su propósito de manera eficiente, confiable y sostenible. Además, sentó las bases para futuras mejoras, ya que el sistema fue diseñado de forma modular,

permitiendo agregar nuevas funciones o adaptarlo a otros módulos de la aplicación VARYS sin grandes modificaciones.

4. Analizar los errores frecuentes, tiempos de ejecución y dificultades del proceso.	X	X		
Fase 2: Definición de requerimientos y diseño de la solución				
1. Clasificar los requerimientos funcionales y no funcionales del sistema	X	X		
2. Diseñar la estructura general del script de automatización con Cypress.		X	X	
3. Definir los criterios de validación, seguridad y compatibilidad del sistema.		X	X	
4. Elaborar la documentación		X	X	X

técnica del diseño del script.				
Fase 3: Desarrollo e implementación del script automatizado				
1. Configurar el entorno de desarrollo e instalar Cypress y herramientas asociadas.	X	X		
2. Programar las funciones principales del script (inicio de sesión, carga y validación de datos).	X	X	X	
3. Ejecutar pruebas internas del script y registrar los resultados.		X	X	
4. Realizar ajustes y optimización del código según los resultados obtenidos.			X	

Fase 4: Pruebas finales y validación			
1. Implementar el script en un entorno controlado con usuarios reales.	X	X	
2. Comparar los resultados del proceso automatizado frente al proceso manual.	X	X	
3. Analizar métricas de tiempo, errores y satisfacción de los usuarios.		X	X
4. Documentar los resultados y validar el cumplimiento de los objetivos del proyecto.		X	X
Fase 5: Entrega del proyecto y cierre			

1. Elaborar el informe final con los resultados del proyecto.	X	X		
2. Presentar la propuesta de mejora continua basada en la automatización.	X	X	X	
3. Preparar la exposición y defensa del proyecto.			X	X
4. Cierre formal del proyecto y entrega del documento de usuario.				X

Nota. Elaboración propia. Cronograma de actividades del ciclo de vida del proyecto

Recursos Necesarios para la Implementación

Tabla 5

Presupuesto necesario para la implementación de la solución.

Recurso	Descripción	Presupuesto anual
Equipo Humano	Este recurso incluye a los profesionales involucrados en el desarrollo e implementación del proyecto: un programador encargado del desarrollo del script en Cypress, un analista de pruebas responsable de validar el funcionamiento del sistema automatizado.	\$ 24.207.076
Equipos y Software	Se utilizará un computador portátil con características adecuadas para desarrollo: procesador Intel Core i5, 8 GB de memoria RAM y disco sólido de 256 GB SSD, con un costo promedio de \$2.998.500. El framework Cypress es gratuito, así como el sistema de control de versiones Git/GitHub. Además, se requerirá una licencia de Microsoft Office 365 (anual) por \$369.999 para la documentación técnica.	\$ 3,368.499
Viajes y Salidas de Campo	Aunque el desarrollo se realizará principalmente de manera remota, se estiman dos visitas presenciales a la	\$ 200,000.00

	organización ubicada en Bogotá para la validación del proceso con los usuarios. Esto incluye transporte urbano y alimentación.	
Materiales y suministros	Incluye papelería, cuadernos, bolígrafos, carpetas y demás elementos necesarios para la planificación, toma de notas y documentación de las pruebas.	\$150.000
Bibliografía	Adquisición de material académico y libros digitales relacionados con automatización, desarrollo web y metodologías ágiles.	0
TOTAL 27.925.575		

Nota. Elaboración propia. Tabla donde se muestran los recursos necesarios para la ejecución de la propuesta con los respectivos valores en el mercado.

Diseño de la Solución

El diseño de la solución se centró en crear un sistema de automatización que reemplazara las tareas manuales de cargue de información en la aplicación web VARYS, utilizando la herramienta Cypress. Este diseño se planteó con el propósito de optimizar los tiempos de ejecución, reducir los errores humanos y estandarizar las actividades que antes dependían totalmente del operador.

El proceso inició con la identificación del flujo exacto del procedimiento que los usuarios realizaban de manera manual. Para ello, se elaboró un diagrama de flujo que permitió visualizar las etapas del proceso, desde el inicio de sesión hasta la confirmación de los registros en el sistema. Con base en este análisis, se definieron los puntos del flujo más críticos, es decir, aquellos donde ocurrían más errores o donde el tiempo de respuesta era mayor.

Análisis del Proceso Actual (AS-IS)

Antes de diseñar la automatización, fue necesario comprender con claridad cómo se ejecutaba el proceso manual dentro del sistema. El procedimiento identificado incluía las siguientes etapas generales:

El proceso manual identificado seguía, de manera general, las siguientes etapas:

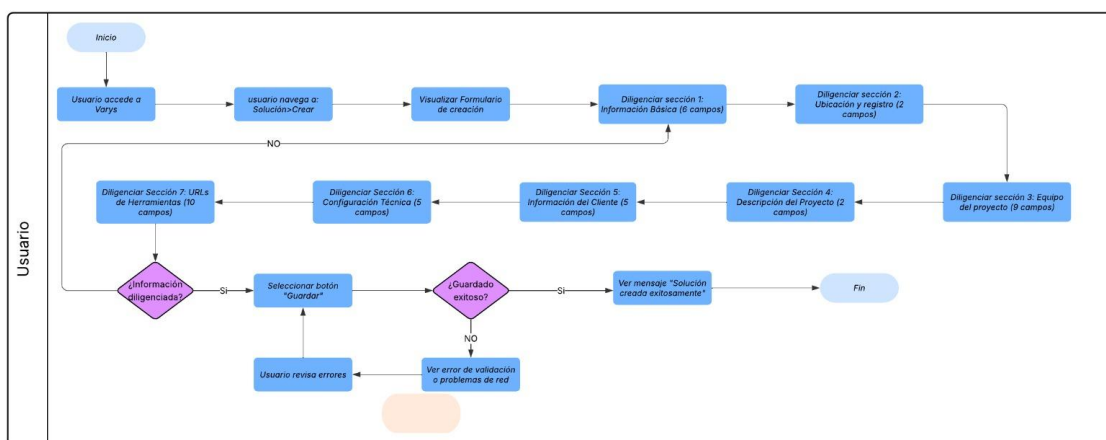
1. Ingreso a la aplicación web VARYS.
2. Autenticación con credenciales de usuario.
3. Selección del módulo correspondiente.
4. Diligenciamiento manual de formularios.
5. Verificación visual de campos obligatorios.

6. Guardado del registro.

Este análisis permitió evidenciar que el mayor consumo de tiempo se concentraba en el diligenciamiento repetitivo y en la validación manual de los datos.

Figura 2

Diagrama de flujo del proceso manual (AS-IS).



Nota. Elaboración propia. La representación gráfica permitió identificar los puntos críticos del proceso y fundamentar técnicamente la necesidad de automatización.

Una vez identificado el flujo, se procedió a diseñar la estructura lógica del script de automatización. Se utilizó Cypress, debido a su capacidad para interactuar directamente con la interfaz del navegador y simular el comportamiento real de un usuario. El script fue diseñado modularmente, dividido en funciones que ejecutan tareas específicas como:

- Ejecución del comando del script
- Cargue de la configuración del proyecto
- Iniciar sesión automática.

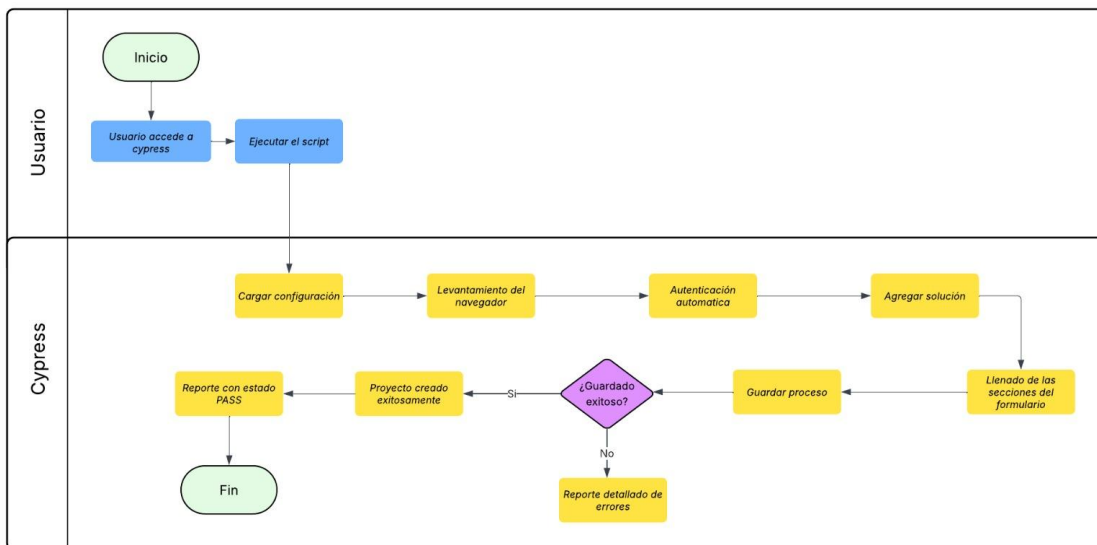
- Cargar datos en los formularios del sistema.
- Validar campos obligatorios y verificar respuestas del servidor.
- Registrar resultados en reportes automáticos de ejecución.

Arquitectura de la Solución Automatizada (TO-BE)

El diseño contempló una arquitectura en la que el script desarrollado en Cypress realiza el levantamiento del navegador, Esta estructura permite que la automatización reproduzca exactamente el comportamiento de un usuario real sin intervenir directamente en la base de datos del sistema.

Figura 3

Arquitectura de la solución automatizada.



Nota. Elaboración propia. La arquitectura muestra la relación entre el script automatizado, el navegador y la aplicación web, evidenciando cómo se ejecuta la automatización dentro del entorno real del sistema.

Para garantizar su funcionamiento estable, se definieron criterios de validación y protocolos de seguridad que permitieron asegurar que los datos cargados fueran verificados antes de finalizar el proceso. Además, se implementaron mecanismos de manejo de errores, como validaciones condicionales y mensajes personalizados, que permiten al sistema reaccionar ante cualquier fallo detectado durante la automatización.

Metodología de Desarrollo Adoptada

El desarrollo del script se realizó de manera incremental, organizando el trabajo en pequeñas iteraciones donde se implementaban funcionalidades específicas y posteriormente se validaban antes de continuar con la siguiente etapa. Este enfoque permitió realizar ajustes progresivos y mejorar la estabilidad del sistema en cada fase del desarrollo.

Este tipo de organización se alinea con el carácter de investigación aplicada del proyecto, ya que permitió diseñar, probar y ajustar la solución en un contexto real, generando mejoras verificables y transformaciones concretas en el entorno donde se identificó la problemática (Hernández Sampieri & Mendoza Torres, 2018).

Se elaboró una documentación técnica del script, incluyendo las instrucciones de instalación, configuración y ejecución. Esta documentación permite que cualquier integrante del equipo pueda ejecutar la automatización sin depender del desarrollador original. Además, se incluyeron comentarios dentro del código para facilitar su comprensión y mantenimiento.

Validación del Diseño

Finalmente, el diseño fue validado a través de un entorno de pruebas controlado, donde se ejecutaron múltiples escenarios simulados con diferentes volúmenes de datos.

Durante estas pruebas se verificó:

- El tiempo total de ejecución.
- La reducción de errores humanos.
- La estabilidad del script en ejecuciones repetidas.

Los resultados confirmaron que el script podía realizar las tareas con precisión, reduciendo significativamente el tiempo total del proceso y los errores humanos.

En síntesis, el diseño de la solución se basó en los principios de eficiencia, escalabilidad y confiabilidad, aprovechando las ventajas técnicas de Cypress y aplicando buenas prácticas de ingeniería de software. Este diseño representa una base sólida no solo para resolver el problema actual, sino también para futuras expansiones de automatización en otros módulos del sistema VARYS o incluso en otras plataformas de la organización.

Análisis de Resultados

Se realizaron varias pruebas comparativas para medir la diferencia entre el proceso manual y el proceso automatizado usando el script desarrollado en Cypress. Las pruebas se realizaron bajo condiciones controladas, utilizando el mismo equipo de cómputo, el mismo navegador y entorno de ejecución, con el fin de garantizar que los resultados fueran comparables y no estuvieran influenciados por factores externos.

En total, se realizaron 20 ejecuciones del proceso manual y 20 ejecuciones del proceso automatizado, utilizando los mismos 20 registros en ambos casos. Los registros fueron seleccionados por representar casos frecuentes dentro del sistema, es decir, registros estándar con campos obligatorios y validaciones comunes del módulo analizado. Esto permitió asegurar que la comparación fuera equitativa y metodológicamente válida.

Los resultados fueron bastante claros y mostraron una mejora importante en la velocidad, la precisión y la forma en que se ejecutan las tareas dentro de la aplicación VARYS.

Cuando el proceso se hacía de manera manual, ingresar toda la información en la aplicación podía tardar entre 3 y 5 minutos por registro, dependiendo de la cantidad de datos y de la atención del usuario. En cambio, al ejecutar el proceso con el script automatizado, el mismo trabajo se completó en un tiempo promedio de 14 a 20 segundos. Esta diferencia demuestra una reducción de más del 90.8% en el tiempo total de ejecución, lo cual representa un avance significativo en términos de eficiencia y optimización del trabajo.

Estos resultados están directamente relacionados con las decisiones tomadas en el diseño de la solución, particularmente con la modularización del script, la implementación de validaciones automáticas y el manejo estructurado de errores definidos en la arquitectura del sistema y en la metodología incremental adoptada durante el desarrollo.

También se pudo notar una gran disminución en los errores humanos. En el proceso manual, era frecuente que se presentaran errores por digitación, omisión de campos o validaciones incorrectas. Sin embargo, el script en Cypress ejecutó todas las acciones siguiendo el mismo flujo establecido en el diagrama del proceso automatizado (TO-BE), sin interrupciones ni fallos, lo que eliminó casi por completo los errores de registro. Esto hace que la información sea mucho más confiable y estandarizada.

Otro aspecto importante es que, gracias a la automatización, ahora se generan reportes automáticos de cada ejecución. Estos reportes permiten tener un control más claro de lo que sucede en cada prueba, saber cuántos registros fueron procesados correctamente y detectar cualquier anomalía en caso de que ocurra. Este elemento está alineado con el diseño técnico del script y fortalece la trazabilidad entre diagnóstico, diseño y resultados.

Con esto se demuestra que automatizar el proceso con Cypress fue una decisión correcta, ya que no solo se redujeron los tiempos y los errores, sino que también se mejoró la eficiencia general del sistema y se liberó tiempo que los trabajadores pueden dedicar a otras tareas de mayor valor dentro de la empresa.

Tabla 6

Tabla de registros creados de forma manual y con la automatización.

Nº Registro	Tiempo manual (segundos)	Tiempo automatizado (segundos)
1	285	24
2	300	28
3	240	23
4	270	25
5	255	26
6	295	27
7	180	22
8	260	25
9	275	29
10	290	23
11	305	30
12	240	21
13	250	24
14	280	28
15	295	22
16	300	27
17	260	25
18	285	29
19	275	23
20	250	26

Nota. Elaboración propia. creaciones de registros de forma manual y automatizados, *fuentes.* autoría propia

Tabla 7

Tabla de promedio por ejecuciones de registros.

Tipo de proceso	Tiempo mínimo (s)	Tiempo máximo (s)	Tiempo promedio (s)	Reducción aproximada
Manual	180	305	272.75	
Automatizado (cypress)	21	30	25.10	= 90.8% menos de tiempo

Nota. Elaboración propia. Se realizan promedios de ejecuciones por caso, autoría propia

Tabla 8

Comparación de errores entre el proceso manual y el automatizado.

Tipo de error	Proceso Manual	Proceso Automatizado (Cypress)
Errores de digitación	A veces el usuario escribe mal la información o comete errores al ingresar los datos.	No se presentan este tipo de errores porque el script llena los datos automáticamente.
Registros incompletos	Es común que se olviden algunos campos obligatorios o se pasen por alto durante el registro.	El sistema revisa todos los campos antes de enviar la información, evitando que queden vacíos.
Duplicación de registros	Puede pasar que un usuario repita una carga por error, generando registros duplicados.	Cypress valida previamente la información, impidiendo que se repita un registro.
Errores en la validación de campos	En algunos casos no se revisan bien ciertos campos o se validan de forma incorrecta.	El script hace las validaciones automáticamente, asegurando que todos los campos estén correctos.
Omisión de pasos en el flujo de trabajo	El usuario puede saltarse algún paso del proceso sin darse cuenta.	El script sigue todos los pasos en orden, sin posibilidad de omitir ninguno.
Retrasos por validaciones manuales	La revisión manual toma tiempo y puede generar demoras o errores.	Las validaciones se hacen de inmediato dentro del script, lo que agiliza el proceso.

Nota. Elaboración propia. lista de los errores más comunes que se presentan en el proceso manual y automatizado.

Conclusiones

El desarrollo de este proyecto permitió demostrar cómo la automatización de procesos puede transformar significativamente la eficiencia operativa dentro de una organización. A lo largo de las diferentes fases, se logró identificar, analizar y resolver un problema real relacionado con el cargue manual de información en la aplicación web VARYS, aplicando herramientas y metodologías propias de la ingeniería de software.

En primer lugar, el análisis y diagnóstico del proceso permitió evidenciar los principales factores que afectaban la productividad: la repetitividad de las tareas, la falta de estandarización y la alta posibilidad de errores humanos. Estos aspectos fueron representados en el diagrama de flujo del proceso manual (AS-IS), donde se identificaron puntos críticos como la digitación repetitiva, la validación visual de campos y la posibilidad de omitir pasos del procedimiento.

Posteriormente, con la definición de los requerimientos y el diseño del script en Cypress, se estableció una solución clara y viable, estructurada bajo principios de modularidad, validaciones automáticas, manejo de errores y generación de reportes. Estas decisiones técnicas influyeron directamente en los resultados obtenidos, ya que permitieron reducir el tiempo promedio de ejecución en un 90.8 %, además de eliminar errores de digitación y omisión de campos obligatorios.

Durante la etapa de implementación y pruebas, se comprobó que la automatización no solo redujo los tiempos y errores, sino que también aumentó la estabilidad del proceso mediante ejecuciones repetidas sin fallos críticos. Además, el uso

de una metodología ágil como Scrum facilitó el desarrollo iterativo y la incorporación de mejoras constantes.

El cumplimiento de los objetivos específicos se reflejó en la identificación del proceso crítico, el diseño estructurado del script, su implementación controlada y la evaluación cuantitativa de los resultados frente al método manual. Todos estos pasos condujeron al logro del objetivo general, que fue optimizar los procedimientos repetitivos mediante la automatización con Cypress.

Finalmente, en términos de sostenibilidad, la solución desarrollada presenta condiciones favorables para su mantenimiento y escalabilidad. La modularidad del código y la posibilidad de integración continua permiten que el sistema pueda adaptarse a cambios sin necesidad de reconstruirlo completamente. Asimismo, la estandarización del proceso y la generación automática de reportes facilitan su replicación en otros módulos del sistema o en contextos similares. Estos resultados se alinean con lo planteado por Parapat et al. (2023), quienes señalan que la automatización contribuye a la eficiencia operativa y a la mejora continua en entornos organizacionales. De igual manera, Dubey (2025) destaca que el uso de herramientas modernas de testing fortalece la calidad del software y reduce la probabilidad de errores humanos en procesos repetitivos.

Se evidenció que la automatización no solo mejora la eficiencia y reduce errores, sino que también fortalece la calidad del proceso, su estabilidad en el tiempo y su capacidad de crecimiento dentro de la organización, coherente con los beneficios reportados en estudios recientes sobre automatización y testing en contextos reales.

Referencias

- Bhimanapati, V., Chhapola, A., & Jain, S. (2023, diciembre). Automation strategies for web and mobile applications in media domains. *International Journal for Research Publication and Seminars*, 14(5), 225–239.
https://www.researchgate.net/publication/383551103_Automation_Strategies_for_Web_and_Mobile_Applications_in_Media_Domains
- BrowserStack. (2025). What is Automated Functional Testing: Types, Benefits & Tools.
<https://www.browserstack.com/guide/what-is-automated-functional-testing>
- Chacon, S., & Straub, B. (2016). *Pro Git* (2nd ed.). Apress.
<https://progit2.s3.amazonaws.com/en/2016-03-22-f3531/progit-en.1084.pdf>
- Congreso de Colombia. (2012). Ley 1581 de 2012 – Por la cual se dictan disposiciones generales para la protección de datos personales.
<https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=49981>
- Congreso de la República de Colombia. (1999). Ley 527 de 1999 - Comercio Electrónico, Mensajes de Datos y Firma Digital en Colombia. Función Pública.
<https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=4276>
- Congreso de la República de Colombia. (2009). Ley 1273 de 2009 - Protección de la información y los datos personales, delitos informáticos y medidas contra el cibercrimen. Policía Nacional de Colombia.
<https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=34492>
- Consejo de Europa. (s. f.). The Budapest Convention. Recuperado el 22 de febrero de 2025, de <https://www.coe.int/en/web/cybercrime/the-budapest-convention-old>

- Constitución Política de Colombia. (1991). Artículo 15 – Derecho al Habeas Data.
<https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=4125>
- Creswell, J. W. (2014). Research design: Qualitative, quantitative, and mixed methods approaches (4th ed.). SAGE Publications.
https://books.google.com.co/books?id=4uB76IC_pOQC&printsec=frontcover&source=gbs_atb#v=onepage&q&f=false
- Cypress. (s.f.). Why Cypress? Cypress Documentation. Cypress.
<https://docs.cypress.io/app/get-started/why-cypress>
- Cypress.io (s.f.). Cypress: Browser testing for modern teams.
 Cypress.<https://www.cypress.io/#create>
- Evans, J. R., & Lindsay, W. M. (2020). Managing for quality and performance excellence (11th ed.). Cengage Learning. <https://studylib.net/doc/27268222/james-r.-evans--william-m.-lindsay---managing-for-quality...>
- Dubey, S. (2025, septiembre). Transforming software quality assurance through intelligent automation.
https://www.researchgate.net/publication/395491713_Transforming_Software_Quality_Assurance_through_Intelligent_Automation_1
- Dumas, M., La Rosa, M., Mendling, J., & Reijers, H. A. (2018). Fundamentals of business process management (2nd ed.). Springer.
https://books.google.com.pe/books?id=KgVTDwAAQBAJ&printsec=frontcover&hl=es&source=gbs_vpt_read#v=onepage&q&f=false

- Fadul, A. A., & Ibrahim, K. A. (2025, agosto). Challenges in GUI test automation for dynamic web applications: A systematic review. Nile Valley University.
https://www.researchgate.net/publication/394449177_Challenges_in_GUI_Test_Automation_for_Dynamic_Web_Applications_A_Systematic_Review
- Falabella. (2025, febrero 20). Portátil HP 15 Intel Core I5-1334U 512GB SSD 16GB RAM 15.6" FHD. Falabella Colombia. https://www.falabella.com.co/falabella-co/product/145395348/Portatil-HP-15-Intel-Core-I5-1334U-512GB-SSD-16GB-RAM-15.6-FHD/145395349?kid=shopp261fa&gclsrc=aw.ds&gad_source=1&gad_campaign_id=21861273110&gclid=CjwKCAjwu9fHBhAWEiwAzGRC_zWN0ZeiFmLUTmndSAbb4B2pf_RG_2MnhgCWKvSX9DTmBBqpd6OCnBoC7U8QAvD_BwE
- Fenton, N., & Bieman, J. (2014). Software metrics: A rigorous and practical approach (3rd ed.). CRC Press. https://github.com/gourab98/SWE-Recommended-Books/blob/master/Books/3-2/Fenton,Norman-SoftwareMetrics_A-Rigorous-and-Practical-Approach,ThirdEdition-CRC-Press.pdf
- Hernández Sampieri, R., & Mendoza Torres, C. (2018). Metodología de la investigación: Las rutas cuantitativa, cualitativa y mixta. McGraw-Hill Education.
<https://bellasartes.upn.edu.co/wp-content/uploads/2024/11/METODOLOGIA-DE-LA-INVESTIGACION-Sampieri-Mendoza-2018.pdf>
- Hernández Sampieri, R., Fernández-Collado, C., Baptista Lucio, P. (2014). Metodología de la investigación (6.a ed.). McGraw-Hill.

- https://apiperiodico.jalisco.gob.mx/api/sites/periodicooficial.jalisco.gob.mx/files/metodologia_de_la_investigacion_-_roberto_hernandez_sampieri.pdf
- IBM Docs. (2021). Creación de un script de prueba desde un script cargado.
<https://www.ibm.com/docs/es/capm?topic=tests-creating-script-test-from-uploaded-script>
- IBM. (s.f.). ¿Qué es la automatización? IBM. <https://www.ibm.com/mx-es/think/topics/automation>
- Indeed. (2025). Sueldo de QA automation engineer/a en Bogotá, Cundinamarca.
Recuperado de <https://co.indeed.com/career/qa-automation-engineer/salaries/Bogotá--Cundinamarca>
- International Organization for Standardization. (2017). ISO/IEC/IEEE 12207:2017 systems and software engineering — Software life cycle processes.
<https://www.iso.org/standard/63712.html>
- International Organization for Standardization. (2022). ISO/IEC 27001:2022 Information security, cybersecurity and privacy protection Information security management systems Requirements. <https://www.iso.org/standard/27001>
- International Organization for Standardization. (2023). ISO/IEC 25010:2023 systems and software engineering — Systems and software quality models.
<https://www.iso.org/standard/78176.html>
- Kim, G., Humble, J., Debois, P., & Willis, J. (2016). The DevOps handbook: How to create world-class agility, reliability, and security in technology organizations. IT Revolution Press.

- http://images.itrevolution.com/documents/DevOps_Handbook_Intro_Part1_Part2.pdf
- Mahesh, G. M., & Usha Sree, R. (2024, noviembre). Optimizing test automation with Selenium for enhancing web application quality. *International Journal of Advanced Research in Science Communication and Technology*.
https://www.researchgate.net/publication/386539833_Optimizing_Test_Automation_with_Selenium_for_Enhancing_Web_Application_Quality
- Graham, D., Erik van Veenendaal, I., & Black, R. (2020). *Foundations of software testing: ISTQB certification (4th ed.)*. Cengage Learning.
https://www.utcluj.ro/media/page_document/78/Foundations%20of%20software%20testing%20-%20ISTQB%20Certification.pdf
- Parapat, F. A. C., Kusuma, G. P., & Nur Majiid, M. R. (2023). Automation testing using Silk Test Workbench for websites. *Procedia Computer Science*, 216, 128–135.
<https://www.sciencedirect.com/science/article/pii/S1877050922021974>
- Parlamento Europeo y Consejo de la Unión Europea. (2016). Reglamento (UE) 2016/679 relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos (Reglamento general de protección de datos). *Diario Oficial de la Unión Europea*. <https://eur-lex.europa.eu/eli/reg/2016/679/oj>
- Presidencia de la República de Colombia. (2013). Decreto 1377 de 2013 – Reglamenta parcialmente la Ley 1581 de 2012.
<https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=53646>

- Pressman, R. S., & Maxim, B. R. (2019). Software engineering: A practitioner's approach (9th ed.). McGraw-Hill Education.
- <https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwiX16aQu-mSAXWTSTABHfGeLPUQFnoECBgQAQ&url=https%3A%2F%2Fstudylib.net%2Fdoc%2F26312018%2Fdokumen.pub-software-engineering-a-practitioners-approach...&usg=AOvVaw1ywFo4tAB9BQozSddjC4HX&opi=89978449>
- Sampieri, R. H., Collado, C. F., Lucio, P. B. (2014). Metodología de la investigación: Las rutas cuantitativa, cualitativa y mixta (6.a ed.). McGraw-Hill.
- https://apiperiodico.jalisco.gob.mx/api/sites/periodicooficial.jalisco.gob.mx/files/metodologia_de_la_investigacion_-_roberto_hernandez_sampieri.pdf
- Schwaber, K., & Sutherland, J. (2020). The Scrum Guide. Scrum.org.
- <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-Latin-South-American.pdf>
- Sommerville, I. (2016). Ingeniería de software (9.ª ed., V. Campos Olgún, Trad.). Pearson Educación. Recuperado de
- https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwj6idWk1PeSAXWBSDABHdrRHh8QFnoECBoQAQ&url=https%3A%2F%2Fwww.studyhalo.com%2Fmedia%2Fresources%2Fresources%2FINF3705%2FTextbook%2FSoftware_Engineering_-_Ian_Sommerville.pdf&usg=AOvVaw2OqaZW8ZG8vas6Tq4S4aEA&opi=89978449

Spillner, A., Tretmans, J., & Linnemann, T. (2014). Software testing foundations (4th ed.). Rocky Nook.

<https://dl.icdst.org/pdfs/files3/729e5abcef3748912096536550f28bb1.pdf>

Syed, R., Suriadi, S., Adams, M., Bandara, W., Leemans, S. J. J., Ouyang, C., ter Hofstede, A. H. M., van de Weerd, I., Wynn, M. T., & Reijers, H. A. (2020). Robotic process automation: Contemporary themes and challenges. *Computers in Industry*, 115, 103162.

https://researchonline.lse.ac.uk/id/eprint/71146/1/Willcocks_Robotic%20process%20automation_author_2017%20LSERO.pdf

Westerman, G., Bonnet, D., & McAfee, A. (2014). *Leading digital: Turning technology into business transformation*. Harvard Business Review Press.

http://repo.darmajaya.ac.id/4452/1/Leading%20Digital_%20Turning%20Technology%20into%20Business%20Transformation%20%28%20PDFDrive%20%29.pdf

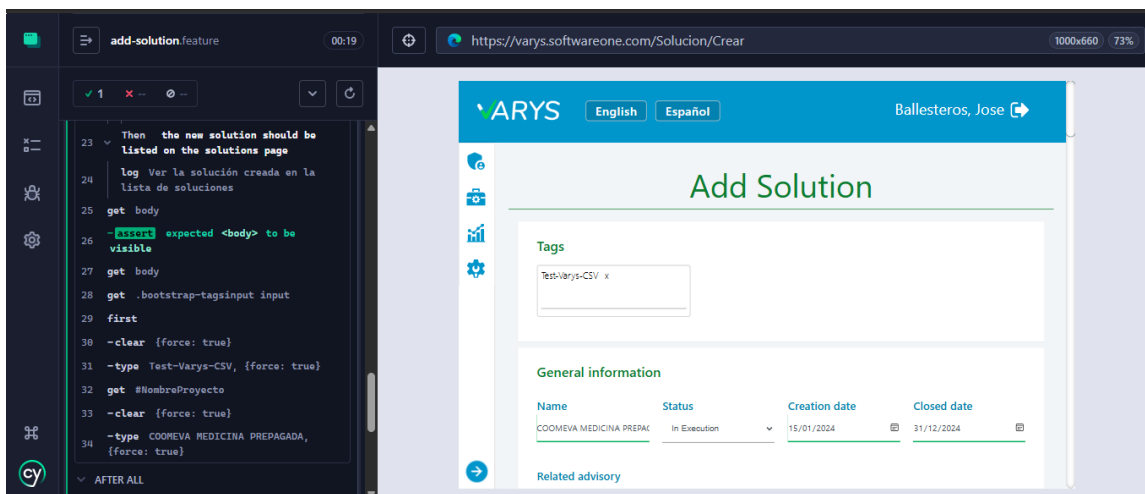
f

Anexos

Repositorio de código GitHub: https://github.com/BallesterosMir/BDD_P

Figura 4

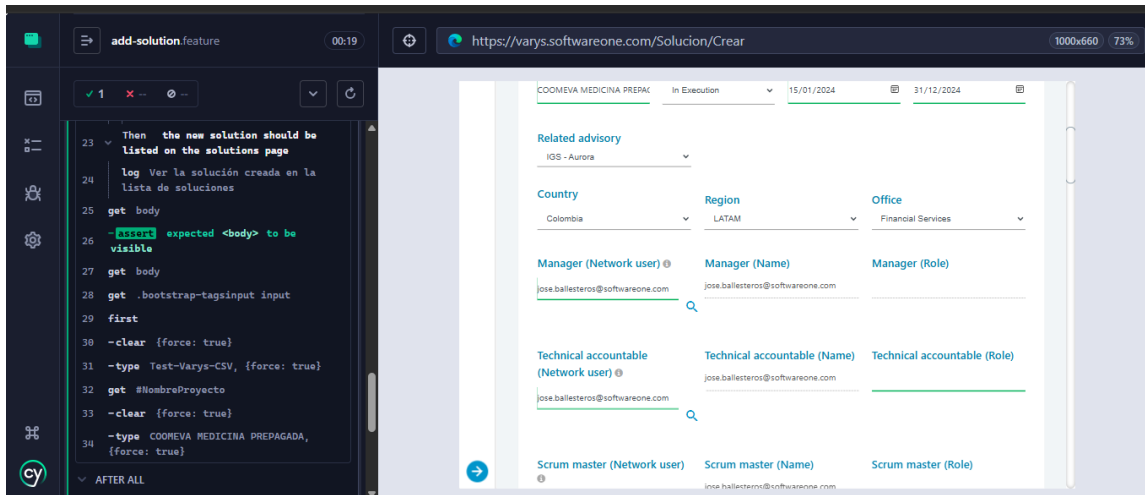
Ejecución de cypress del aplicativo web.



Nota. Captura de pantalla del sistema VARYS durante la ejecución del proceso automatizado con Cypress. Elaboración propia.

Figura 5

Aplicación completando la información del formulario.

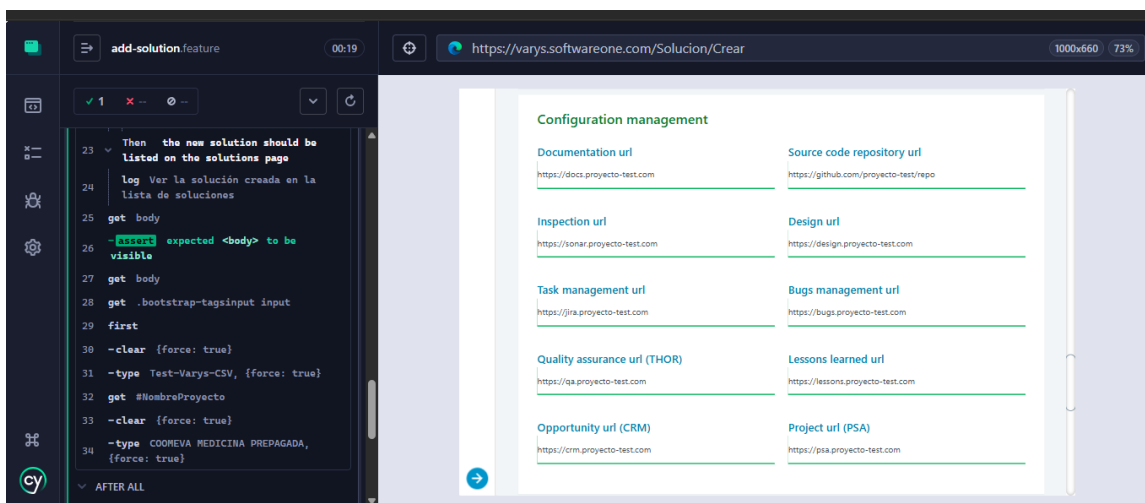


Nota. Prueba ejecutada exitosamente con la información registrada en el formulario.

Elaboración propia.

Figura 6

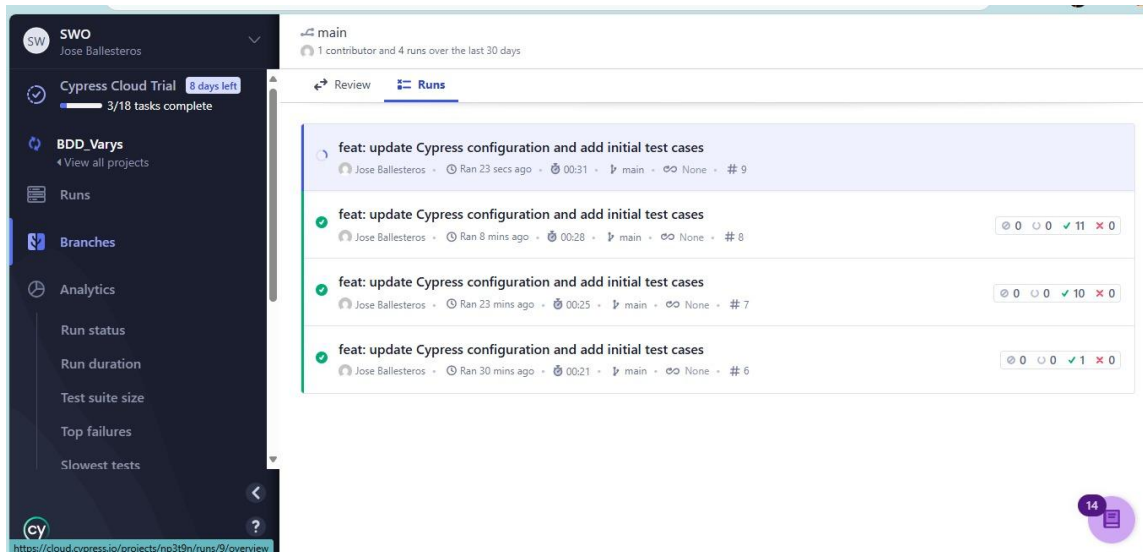
Prueba exitosa con Cypress tiempo de ejecución



Nota. Prueba finalizada exitosamente mediante Cypress. Elaboración propia.

Figura 7

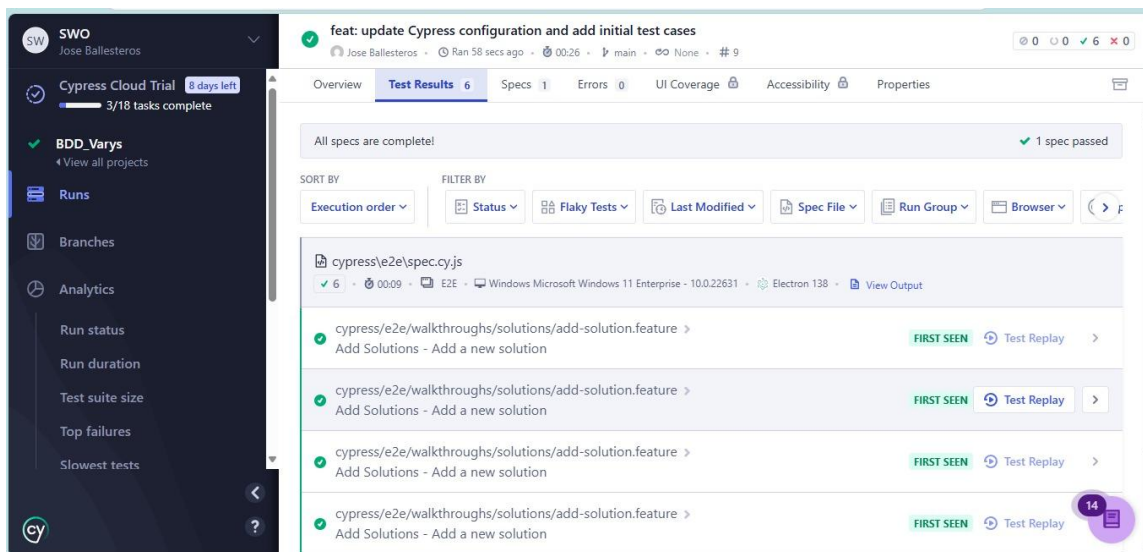
Muestra de ejecución en cloud de cypress.



Nota. Ejecuciones de los procedimientos automatizados en Cypress. Elaboración propia.

Figura 8

Visualización de los resultados en cloud cypress.



Nota. Visualización de las ejecuciones en Cypress Cloud. Elaboración propia.

Figura 9

Ejecución fallida en cloud cypress.

The screenshot displays the Cypress Cloud dashboard for user Jose Ballesteros. The main area shows a test run with the following details:

- Specs:** 2 specs on 1 machine, 2 specs failed.
- Message:** "All specs are complete! 2 specs failed"
- Run Summary:** "Make your tests run 26 seconds faster by adding 1 machine in CI."
- Failed Specs:**
 - Spec 1: `cypress\e2e\walkthroughs\login\login-wt.feature` (00:22)
 - Spec 2: `cypress\e2e\walkthroughs\solutions\add-solution.feature` (00:16)

The interface includes a sidebar with navigation options like "Cypress Cloud Trial", "BDD_Varys", "Runs", "Branches", and "Analytics".

Nota. Muestra de resultados fallidos en Cypress Cloud. Elaboración propia.

Figura 10

Ingreso de la información de forma manual con tiempo incluido.

The screenshot shows the "Add Solution" form in the VARYS system. The interface includes a header with the VARYS logo, language options (English, Español), and the user name "Ballesteros, Jose".

The form contains the following sections:

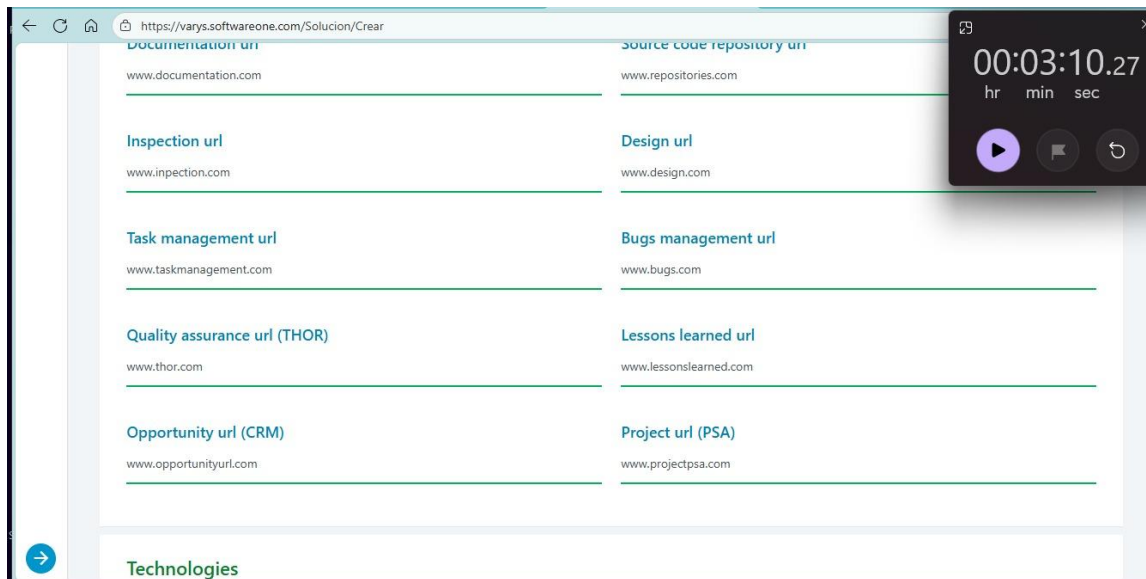
- Tags:** A text input field containing "Test-Varys-CSV x".
- General information:** A table with the following data:

Name	Status	Creation date	Closed date
COOMEVA MEDICINA PREPAC	In Execution	15/01/2024	31/12/2024
- Related advisory:** A section for adding related advisory information.

Nota. Formulario con información ingresada manualmente durante el proceso. Elaboración propia.

Figura 11

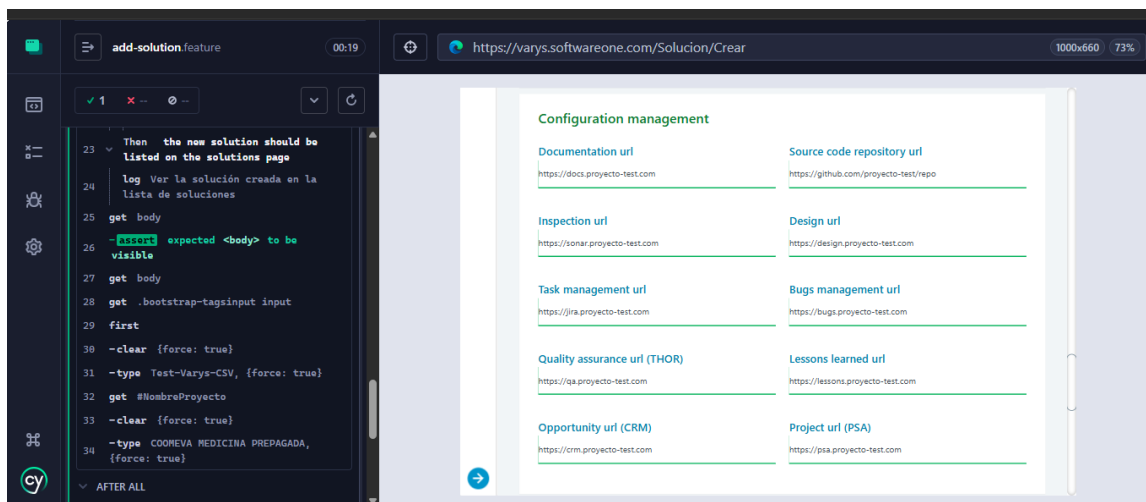
Validación de los tiempos en ejecución manual.



Nota. Ejecución de una prueba manual en VARYS con registro de tiempos. Elaboración propia.

Figura 12

Duración del registro de forma automatizada con Cypress.



Nota. Duración de una prueba automatizada ejecutada en Cypress. Elaboración propia.

Figura 13

Formato de Observación Estructurada del Proceso Manual.

Formato de Observación Estructurada del Proceso Manual

Proyecto:	Automatización del proceso de registro en VARYS
Observador	
Fecha:	
Usuario observado:	
Número de ejecución:	

1. Registro de pasos del proceso

Paso del proceso	Descripción de la acción	Tiempo (segundos)	¿Se presentó error? (Sí/No)	Tipo de error
1	Ingreso al sistema y crear solución			
2	Diligenciamiento del formulario			
3	Validación de campos			
4	Confirmación de los campos diligenciados			
5	Guardado final			

2. Observaciones generales

- ¿Se omitió algún paso? _____
- ¿Hubo dudas del usuario? _____
- ¿Se repitió algún procedimiento? _____

Nota. Formato utilizado como instrumento de recolección de datos. Elaboración propia.

Figura 14

Hoja de Registro de Cronometraje.

Hoja de Registro de Cronometraje

Objetivo: Medir el tiempo total de ejecución del proceso manual y automatizado.

N° Ejecución	Tipo de proceso (Manual/Automatizado)	Tiempo total (segundos)	Observaciones
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

Promedio total calculado: _____ segundos

Nota. Formato utilizado para la recolección de datos de los registros. Elaboración propia.

Figura 15

Guía de Entrevista Semiestructurada.

Guía de Entrevista Semiestructurada

Dirigida a: Usuarios que realizan el proceso manual en VARYS

Duración estimada: 10–15 minutos

Preguntas:

1. ¿Cuánto tiempo considera que tarda en promedio realizar un registro?
2. ¿Qué parte del proceso le parece más repetitiva o demorada?
3. ¿Ha cometido errores durante el diligenciamiento? ¿Cuáles?
4. ¿Considera que el proceso podría automatizarse? ¿Por qué?
5. ¿Qué mejoraría del proceso actual?

Observaciones del entrevistador:

Nota. Formato de recolección de datos. Elaboración propia.

Figura 16

Encuesta Estructurada de Percepción.

Encuesta Estructurada de Percepción

Instrucciones: Marque la opción que mejor represente su opinión.

Escala:

1 = Totalmente en desacuerdo

2 = En desacuerdo

3 = Neutral

4 = De acuerdo

5 = Totalmente de acuerdo

Ítem	1	2	3	4	5
El proceso manual es fácil de realizar					
El proceso manual consume demasiado tiempo					
Es frecuente cometer errores durante el registro					
La automatización mejora la eficiencia del proceso					
Me siento más seguro usando el sistema automatizado					

Nota. Formato utilizado para la recolección de datos de los registros. Elaboración propia.

Figura 17

*Escala SUS (System Usability Scale)***Escala SUS (System Usability Scale)**

Escala:

1 = Totalmente en desacuerdo

2 = En desacuerdo

3 = Neutral

4 = De acuerdo

5 = Totalmente de acuerdo

Ítem	1	2	3	4	5
1. Considero que me gustaría usar este sistema con frecuencia.					
2. El sistema me parece innecesariamente complejo.					
3. Es frecuente cometer errores durante el registro					
4. El sistema es fácil de usar.					
5. Necesitaría apoyo técnico para usar este sistema					
6. Las funciones del sistema están bien integradas.					
7. El sistema presenta demasiadas inconsistencias.					
8. La mayoría de las personas aprenderían a usar este sistema rápidamente					
9. El sistema es complicado de usar.					
10. Me siento seguro usando el sistema.					
11. Necesité aprender muchas cosas antes de poder usar el sistema.					

Nota. Formato utilizado para la recolección de datos de los registros. Elaboración propia.