

	<b>UNIVERSIDAD FRANCISCO DE PAULA SANTANDER OCAÑA</b>			
	Documento	Código	Fecha	Revisión
<b>FORMATO HOJA DE RESUMEN PARA TRABAJO DE GRADO</b>	<b>F-AC-DBL-007</b>	<b>21-10-2022</b>	<b>B</b>	
Dependencia	Aprobado		Pág.	
<b>DIVISIÓN DE BIBLIOTECA</b>	<b>SUBDIRECTOR ACADEMICO</b>		<b>1(69)</b>	

## RESUMEN – TRABAJO DE GRADO

<b>AUTORES</b>	Pedro Antonio Chacon Claro		
<b>FACULTAD</b>	Ingenierías		
<b>PLAN DE ESTUDIOS</b>	Ingeniería de sistemas		
<b>DIRECTOR</b>	Byron Cuesta Quintero		
<b>TÍTULO DE LA TESIS</b>	Desarrollo e implementación de una herramienta que permita la automatización de los softwares de monitoreo del banco de Bogotá (bdb).		
<b>TITULO EN INGLES</b>	Development and implementation of a tool that allows the automation of the monitoring software of the bank of Bogotá (bdb).		
<b>RESUMEN</b> (70 palabras)			
<p>El presente documento contiene el proyecto de pasantías, se puede reflejar los aportes realizados por el pasante Pedro Antonio Chacon Claro en la empresa Banco De Bogotá S.A.S, asignándole el cargo de Analista IV, cumpliendo con funciones de desarrollador en Python, se puede observar el desarrollo de una página web en Django y el desarrollo de una Inteligencia artificial que permite la automatización de procesos de monitoreo en la empresa.</p>			
<b>RESUMEN EN INGLES</b>			
<p>This document contains the internship project, it can reflect the contributions made by the intern Pedro Antonio Chacon Claro in the company Banco De Bogotá S.A.S, assigning him the position of Analyst IV, fulfilling developer functions in Python, you can observe the development of a web page in Django and the development of artificial intelligence that allows the automation of monitoring processes in the company.</p>			
<b>PALABRAS CLAVES</b>	Inteligencia Artificial Django Api Programacion apache red neuronal		
<b>PALABRAS CLAVES EN INGLES</b>	Artificial Intelligence Django Api Programming apache neural network		
<b>CARACTERÍSTICAS</b>			
<b>PÁGINAS:</b> 110	<b>PLANOS:</b> 4	<b>ILUSTRACIONES :</b> 73	<b>CD-ROM:</b> 0



Vía Acolsure, Sede el Algodonal, Ocaña, Colombia - Código postal: 546552  
 Línea gratuita nacional: 01 8000 121 022 - PBX: (+57) (7) 569 00 88  
 atencionalciudadano@ufpso.edu.co - www.ufpso.edu.co

**Desarrollo e implementación de una herramienta que permita la automatización de los softwares de monitoreo del banco de Bogotá(bdb).**

**Pedro Antonio Chacon Claro**

**Facultad de Ingenierías, Universidad Francisco De Paula Santander Ocaña**

**Ingeniería De Sistemas**

**Msc. Byron Cuesta Quintero**

**12 de agosto de 2023**

## Índice

Listado de Figuras .....	6
Listado de Tablas .....	10
1. Desarrollo e implementación de una herramienta que permita la automatización de los softwars de monitoreo del banco de Bogotá(bdb). .....	11
1.1 Descripción de la empresa .....	11
1.1.1 Misión. ....	13
1.1.2 Visión. ....	13
1.1.3 Valores. ....	14
1.1.3 Objetivos de la empresa. ....	14
1.1.4 Descripción de la estructura organizacional. ....	15
1.1.5 Descripción de la dependencia y/o proyecto al que fue asignado. ...	15
1.2 Diagnóstico inicial de la dependencia asignada .....	16
1.2.1 Planteamiento del problema .....	18
1.3 Objetivos de la pasantía .....	21
1.3.1 General. ....	21
1.3.2 Específicos. ....	21
1.4 Descripción de las actividades a desarrollar en la misma .....	22
1.5 Cronograma de actividades .....	25
2. Capitulo 2 Marco de Referencia .....	27

2.1 Marco Conceptual .....	27
2.1.1 inteligencia artificial .....	27
2.1.2 automatización robótica de procesos (rpa): .....	28
2.1.3 Django .....	29
2.1.4 Jinja .....	30
2.1.5 Bootstrap .....	31
2.1.6 SiteScope .....	31
2.1.7 Smax .....	32
2.1.8 Apache .....	33
2.1.9 SQL .....	34
2.1.10 API .....	34
2.1.11 metodología scrum .....	35
2.2 Marco Legal .....	37
2.2.1 Iso 27001 .....	37
2.2 .2 Ley 1341 de 2009 .....	37
2. 2.3 Ley 1581 de 2012 .....	38
2.2.4 Resolución 3066 de 2011: .....	38
2.2.5 <i>ley 23 de 1982 Ver la Ley 44 de 1993, Ver el Decreto Nacional 1474 de 2002</i> .....	38
2.3 Tabla herramientas usadas .....	39
3. Capitulo 3 informe de cumplimiento de trabajo .....	41

3.1 identificar el Proceso a seguir en el banco de Bogotá para solicitar un monitoreo de cualquier tipo. ....	41
3.2 Realizar búsqueda y revisión de tecnologías para ver cómo se puede implementar una herramienta de automatización para los colaboradores del banco. ....	58
3.3 Validar si la solución encontrada cumple con los distintos requerimientos. ....	59
3.4 Fortalecer mis conocimientos según los cursos que me entrego la empresa. ....	59
3.5 Estructurar el plan de implementación de la herramienta de automatización .....	60
3.6 Desarrollar la herramienta que permita la automatización de SITESCOP62	
3.6.1 creación de proyecto en Django. ....	62
3.6.2 Desarrollo vistas HTML .....	63
3.6.3 Desarrollo estilos css y traemos las imágenes .....	65
3.6.4 desarrollo js para traer información de sitescope .....	66
3.6.5 hacemos primera entrega .....	67
3.6.6 creación de login contra directorio activo del banco: .....	69
3.6.7 creación del funcionamiento traer reportes: .....	73
3.6.8 creación del funcionamiento habilitar y deshabilitar: .....	77
3.6.9 creación del funcionamiento crear monitoreo de url: .....	79
3.6.10 creación de alertas: .....	81
3.6.11 Hacemos entrega final de la primera herramienta: .....	83

3.7 Desarrollar la herramienta de que permita la automatización de smax ...	85
3.7.1 Desarrollo de chatbot .....	85
3.7.2 Entrenamiento de IA .....	87
3.7.3 Navegación a smax .....	94
3.8 Mejorar diseño de IA del Banco Bogotá .....	99
3.9 Planificación y diseño de pruebas. ....	100
3.10 Aplicación de pruebas funcionales. ....	101
3.11 Corregir errores y entregar las nuevas versiones de las herramientas	102
3.12 Crear manual de instrucciones de las herramientas .....	102
4. Diagnostico final .....	103
5. Conclusiones .....	104
6. Recomendaciones .....	106
Referencias .....	107

## Listado de Figuras

<i>Figura 1 Estructura organizacional de la empresa .....</i>	<i>15</i>
<i>Figura 2 Árbol de problemas .....</i>	<i>20</i>
<i>Figura 3 diseño red neuronal .....</i>	<i>28</i>
<i>Figura 4 Rest API.....</i>	<i>30</i>
<i>Figura 5 Inicio SITESCOPE .....</i>	<i>32</i>
<i>Figura 6 SMAX.....</i>	<i>33</i>
<i>Figura 7 Apache.....</i>	<i>34</i>
<i>Figura 8 funcionamiento de un api.....</i>	<i>35</i>
<i>Figura 9 Desarrollo scrum.....</i>	<i>36</i>
<i>Figura 10 solicitud monitoreo .....</i>	<i>41</i>
<i>Figura 11 tipo de monitoreo .....</i>	<i>42</i>
<i>Figura 12 Excel ejemplo monitoreo .....</i>	<i>43</i>
<i>Figura 13 conexión ping servidor .....</i>	<i>44</i>
<i>Figura 14 remote server .....</i>	<i>44</i>
<i>Figura 15 explicación uso de sitescope .....</i>	<i>45</i>
<i>Figura 16 explicación uso de sitescope .....</i>	<i>45</i>
<i>Figura 17 explicación uso de sitescope.....</i>	<i>46</i>
<i>Figura 18 explicación uso de sitescope .....</i>	<i>47</i>
<i>Figura 19 explicación uso de sitescope .....</i>	<i>48</i>
<i>Figura 20 explicación uso de sitescope .....</i>	<i>49</i>
<i>Figura 21 explicación uso de sitescope.....</i>	<i>49</i>

<i>Figura 22 explicación uso de sitescope</i> .....	50
<i>Figura 23 explicación uso de sitescope</i> .....	51
<i>Figura 24 explicación uso de sitescope</i> .....	52
<i>Figura 25 explicación uso de sitescope</i> .....	53
<i>Figura 26 explicación uso de sitescope</i> .....	54
<i>Figura 27 explicación uso de sitescope</i> .....	54
<i>Figura 28 explicación uso de sitescope</i> .....	55
<i>Figura 29 explicación uso de sitescope</i> .....	56
<i>Figura 30 explicación uso de sitescope</i> .....	56
<i>Figura 31 explicación uso de sitescope</i> .....	57
<i>Figura 32 manual uso de sitescope</i> .....	58
<i>Figura 33 recursos estudio</i> .....	60
<i>Figura 34 Reuniones diarias</i> .....	61
<i>Figura 35 Proyecto Django</i> .....	62
<i>Figura 36 vista HTML</i> .....	64
<i>Figura 37 Vista css</i> .....	65
<i>Figura 38 IMGs</i> .....	66
<i>Figura 39 JS</i> .....	67
<i>Figura 40 Primera entrega herramienta</i> .....	68
<i>Figura 41 vista herramienta</i> .....	69
<i>Figura 42 creación del login</i> .....	70
<i>Figura 43 cambio de base de datos</i> .....	70
<i>Figura 44 Vista.PY</i> .....	71

<i>Figura 45 Redireccionamiento</i> .....	72
<i>Figura 46 Login</i> .....	73
<i>Figura 47 get traer reportes</i> .....	74
<i>Figura 48 funcionalidad botones</i> .....	75
<i>Figura 49 acción clic en botón descargar</i> .....	76
<i>Figura 50 reporte que entrega la herramienta</i> .....	77
<i>Figura 51 visual habilitar deshabilitar</i> .....	78
<i>Figura 52 funcionalidad botón habilitar deshabilitar</i> .....	79
<i>Figura 53 visual crear monitoreo</i> .....	80
<i>Figura 54 funcionalidad api crear monitoreo</i> .....	80
<i>Figura 55 Alertas</i> .....	81
<i>Figura 56 subiendo herramienta a sitescope</i> .....	84
<i>Figura 57 limpia de mensajes</i> .....	85
<i>Figura 58 respuesta predeterminada</i> .....	86
<i>Figura 59 funcionamiento Bot</i> .....	87
<i>Figura 60 label y limpieza de datos</i> .....	88
<i>Figura 61 funcionalidad tokenizador</i> .....	89
<i>Figura 62 clase conjunto de datos</i> .....	90
<i>Figura 63 funcionalidad datos cargados</i> .....	91
<i>Figura 64 división de los datos prueba / entrenamiento</i> .....	91
<i>Figura 65 creación red neuronal</i> .....	92
<i>Figura 66 creación iteraciones</i> .....	93
<i>Figura 67 prueba funcional IA</i> .....	94

<i>Figura 68 Instancia Chrome</i> .....	95
<i>Figura 69 abrir smax</i> .....	96
<i>Figura 70 login smax</i> .....	97
<i>Figura 71 buscar id</i> .....	98
<i>Figura 72 cierre peticiones smax</i> .....	99
<i>Figura 73 pruebas</i> .....	100

**Listado de Tablas**

<i>Tabla 1 matriz DOFA</i> .....	16
<i>Tabla 2 Descripción de objetivos</i> .....	22
<i>Tabla 3 Cronograma</i> .....	25
<i>Tabla 4 Herramientas más usadas</i> .....	39

## **1. Desarrollo e implementación de una herramienta que permita la automatización de los softwares de monitoreo del banco de Bogotá(bdb).**

### **1.1 Descripción de la empresa**

La empresa BDB (Banco de Bogotá) es una compañía que opera principalmente en el sector banca.

Banco de Bogotá S.A es el segundo banco más grande de Colombia cuando activos se refiere y ofrecen una gran gama de productos y servicios bancarios a sus distintos usuarios, la entidad a lo largo de la historia ha apoyado el desarrollo económico del país, acompañando a los empresarios e impulsado el crecimiento y progreso de sus negocios. Ofreciendo sus servicios desde hace 145 años.

La evolución del Banco de Bogotá ha traspasado fronteras, promoviendo los negocios de sus clientes en la región. Al estar presente en 12 países, Colombia, Panamá, Costa Rica, Nicaragua, Honduras, El Salvador, Guatemala, México, Estados Unidos, Bahamas, Islas Caimán y Nassau. Adicionalmente, la entidad en sus 145 años ha facilitado los servicios financieros a nivel nacional e internacional. El Banco de Bogotá consolidado, cuenta hoy con 1.477 oficinas,

3.375 cajeros electrónicos y 10.093 corresponsales bancarios, que permiten realizar pagos,

transferencias, retiros en efectivo, depósitos, recargas de celular, entre otros.

Y con el auge de las nuevas tecnologías y su implementación en el sistema financiero, el Banco de Bogotá ha puesto al servicio de sus clientes y del país servicios ágiles e innovadores de Banca Móvil, diseñados para facilitar la vida de sus clientes, así como para apoyar la bancarización y la inclusión financiera.

Sus áreas de desempeño son:

“SECTOR BANCARIO. Banca Personas es el segmento que incluye a nuestros clientes naturales mayores de edad. También, abarca el portafolio para niños y jóvenes.(De Bogotá, s. f 2023.)

Contamos con una amplia red de atención y un portafolio completo de productos con los que tendrás beneficios, descuentos y promociones.(De Bogotá, s. f. 2023)

Tenemos ofertas destacadas para ti en relación a cuentas, créditos, tarjetas, seguros, opciones de financiación, portafolios integrales e inversión.(De Bogotá, s. f. 2023)

En relación a Cuenta de Ahorros, puedes elegir la que más se ajuste a tus necesidades. Tenemos varias opciones como Flexiahorro, Libreahorro, AFC, Ahorro Programado, entre otros.(De Bogotá, s. f. 2023)

En Tarjetas Débito también tienes varias alternativas. Está la tarjeta electrón, preferencial, amparada, entre otras. Y en Tarjetas de Crédito también puedes elegir la opción que más se ajuste a tus necesidades dependiendo de las características y beneficios a los que quieras aplicar. (De Bogotá, s. f. 2023)

Aplica a créditos de vivienda, libranzas, vehículo, de estudio, leasing, adelanto denómina, entre otros. (De Bogotá, s. f. 2023)

TECNOLOGIA . El banco tiene un área de tecnologías que se dividen por módulos, estos módulos son encargados de desarrollar diferentes herramientas tecnológicas para mejorar la infraestructura del banco un claro ejemplo es el módulo NASA en el cual me encuentro actualmente, que es el encargado de automatizar procesos por medio de Python.

### ***1.1.1 Misión.***

Un banco líder en Colombia para el mercado de empresas, personas, sector social y oficial. Un banco siempre a la vanguardia para brindar a sus clientes soluciones anticipadas, que les permitan vivir una experiencia bancaria satisfactoria. (De Bogotá, s. f. 2023)

### ***1.1.2 Visión.***

Ser un banco líder en Colombia y centro américa que crece con el progreso de sus clientes, de su equipo humano, de sus accionistas y del país. (De Bogotá, s. f. 2023)

### **1.1.3 Valores.**

Estos son los valores que más representan a la empresa, que cimentan las bases del banco y lo que se espera que se aplique de forma permanente en el día a día.

Compromiso, Honestidad, Liderazgo, Eficiencia, Servicio, Respeto.(De Bogotá, s. f. 2023)

### ***1.1.3 Objetivos de la empresa.***

Cumplir los múltiples objetivos de desarrollo sostenible

Continuar creciendo, generando riqueza de forma sostenible, rentable y transparente

Implementar un sistema de gestión de calidad.

Contribuir al desarrollo sostenible y a una mejor calidad de vida en nuestro país.

Promover el desarrollo personal y colectivo de nuestros colaboradores, proveedores y otras contrapartes.

Disminuir al mínimo el impacto ambiental de nuestras operaciones y el que generan



medio de inteligencia artificial.

Esta célula es parte importante del banco ya que con los desarrollos logran ahorrar tiempo y dinero del banco, como tal busca mejorar los procesos de la herramienta MriroFocus y SITESCOPE, ya que son herramientas de monitoreo del banco y al automatizarlas el flujo de información es más eficiente y le permite al banco centrarse en otros tipos de problemas, Por otro lado, esta célula también ayuda con el desarrollo en React de varias herramientas web del banco Bogotá.

## 1.2 Diagnóstico inicial de la dependencia asignada

Tabla 1 matriz DOFA

	Oportunidades	Amenazas
Matriz DOFA	Interés por parte del cliente al tener procesos mucho más rápidos.	Cancelación de la célula.
	Aumento eficiencia por medio de inteligencia artificial en los procesos del banco.	Cambios en los requerimientos de los proyectos que tenga la célula.
	Mitigar el error humano.	Fallos de la Inteligencia artificial desarrollada.

Fortalezas	Estrategias FO	Estrategias FA
<p>Capacitaciones y cursos por parte de la empresa sobre Python y creación de inteligencia artificial (IA).</p> <p>Personal capacitado en temas esenciales para el desarrollo de inteligencia artificial.</p>	<p>Se hacen reuniones diarias para mejorar los procesos de cada integrante de la célula</p> <p>Desarrollo de inteligencia artificial de acuerdo a las necesidades del banco</p>	<p>El banco hace evaluaciones de las células anualmente para poder mejorar, reestructurar o eliminar alguna célula</p> <p>Tener un incremento de tecnologías como la inteligencia artificial y un buen desarrollo en las actividades se tendrá evaluación de estas actividades constantemente.</p>
Debilidades	Estrategias DO	Estrategias DA
<p>El tener que capacitar desde 0 varios integrantes para desarrollar sus actividades hacen que los procesos tomen más tiempo.</p>	<p>Contratar personal más capacitado, así que sea más sencillo entrenarlos</p> <p>Adquirir recursos que le permitan al banco agilizar los procesos</p>	<p>Preparar estrategias para mejorar los procesos del banco y de esta forma superar a la competencia.</p> <p>Diseñar una estrategia para mejorar un desarrollo del talento humano.</p>

<p>Muchos procesos se generan y se hacen de forma apresurada, lo cual puede generar errores en el código</p> <p>Cambios en otras células y procesos que afectan a nuestra célula.</p>		
---	--	--

### ***1.2.1 Planteamiento del problema***

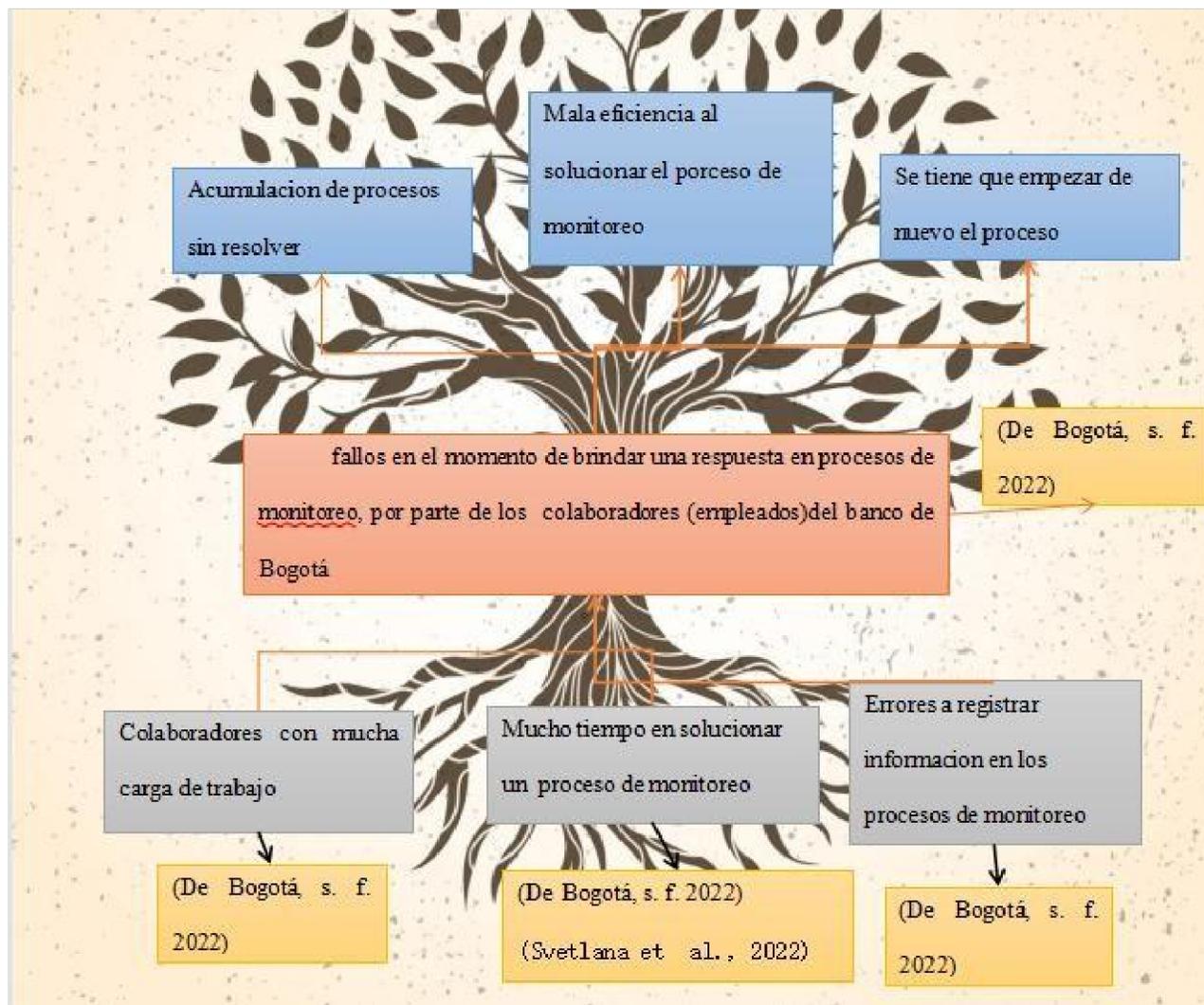
El Banco de Bogotá, identifico que se están presentando fallos en el momento de brindar una respuesta en procesos de monitoreo, referente a que los colaboradores (empleados) demoran en responder o cometen errores en el proceso, lo cual implica un mayor tiempo para solucionarlo solicitado.

Debido a este problema observar que el rendimiento y eficacia de los procesos hace que los tiempos de solución de las distintas actividades del banco se retrasen, por ende, se pierde tiempo y dinero.

Algo que se debe considerar es que el banco de Bogotá inició labores el 15 de noviembre de 1870 como una de las primeras instituciones financieras creadas en el país, desde su creación hasta el 2023, el banco ha estado en constante evolución tecnológica, lo que le ha permitido posicionarse como uno de los mejores bancos del país, con la creación de grupo aval y una gran expansión, el nacional e internacionalmente, identifico que el problema humano, que es la demora y errores que cometen el personal al hacer procesos más simples, como congelar una actividad de monitoreo. (De Bogotá, s. f. 2023)

Por eso el banco de Bogotá creo la célula NASA y quiere que esta célula se encargue de mejorar los procesos de monitoreo del banco para que estos sean solucionados en el menor tiempo posible y se eviten fallos en el proceso, de esta forma los procesos se vuelvan más eficientes y que ya no exista tanto personal humano encargado de los mismos procesos hace que el banco Bogotá pueda centrarse en ser más competitivo en otras áreas dando más recursos y mejorando como empresa.

Figura 2 Árbol de problemas



Nota. Creación propia (2023)

### **1.3 Objetivos de la pasantía**

#### ***1.3.1 General.***

Implementar inteligencia artificial y automatización de procesos robóticos que permita automatizar procesos de MricroFocus y SITESCOP en la empresa Banco de Bogotá.

#### ***1.3.2 Específicos.***

Analizar el proceso actual de las herramientas de monitoreo del banco de Bogotá

Definir una estructura del modelo de trabajo para la automatización, teniendo en-cuenta las especificaciones del banco.

Desarrollar la herramienta de automatización que cumpla con las necesidades y funcionalidades estipuladas por el banco

Evaluar el Funcionamiento de la herramienta de automatización, que cumpla los requerimientos establecidos

#### 1.4 Descripción de las actividades a desarrollar en la misma

**Tabla 2** Descripción de objetivos

Objetivo General	Objetivos Específicos	Actividades para desarrollar
Implementar diferentes IA's que permitan automatizar procesos de MricroFocus y SITESCOP en la empresa Banco de Bogotá	Analizar el proceso actual de las herramientas de monitoreo del banco de bogota	<p>Realizar búsqueda y revisión de tecnologías para ver como es el comportamiento de la inteligencia artificial y automatización de procesos Robóticos.</p> <p>Evaluar las diferentes soluciones que puede ofrecer la inteligencia artificial y la automatización de procesos robótico.</p> <p>Validar si la solución</p>

---

encontrada cumple con los distintos requerimientos.

---

Definir una estructura del modelo de trabajo para la herramienta de automatización,

Fortalecer mis conocimientos según los cursos que me entrego la empresa

teniendo en-cuenta las especificaciones del banco

Estructurar el plan de implementación de la herramienta de automatización

---

Desarrollar la herramienta de automatización que cumpla con

Desarrollar la herramienta que permita la automatización de SITESCOP

Desarrollar herramienta que

---

---

las necesidades y funcionalidades permita la automatización de  
estipuladas por el banco

MricroFocus smax

Adecuar la herramienta a los  
diseños del Banco Bogotá

---

Evaluar el Funcionamiento de la  
herramienta de  
automatización, que cumpla los  
requerimientos  
establecidos

Planificación y diseño de  
pruebas.

Aplicación de pruebas  
funcionales.

Corregir errores y entregar las  
nuevas versiones de la IA.

Crear manual de  
instrucciones de la IA

---



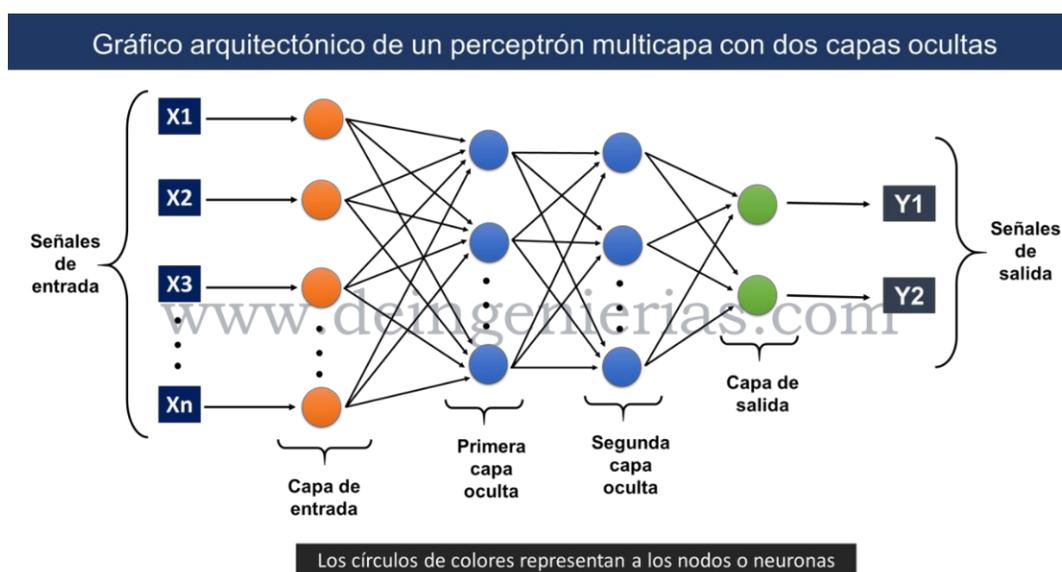




**Machine Learning:** Es un tipo de inteligencia artificial que permite que los sistemas aprendan y mejoren a partir de datos y experiencia previa sin estar especialmente programados para cada tarea.

**Redes Neuronales Artificiales:** Son sistemas que imitan el funcionamiento del cerebro humano, permitiendo el aprendizaje y la toma de decisiones en base a patrones.

**Figura 3** diseño red neuronal



Autor: (Deingenierias.com, 2020)

### 2.1.2 automatización robótica de procesos (rpa):

La rpa son "bots" creados para que puedan aprender, imitar y luego ejecutar procesos de

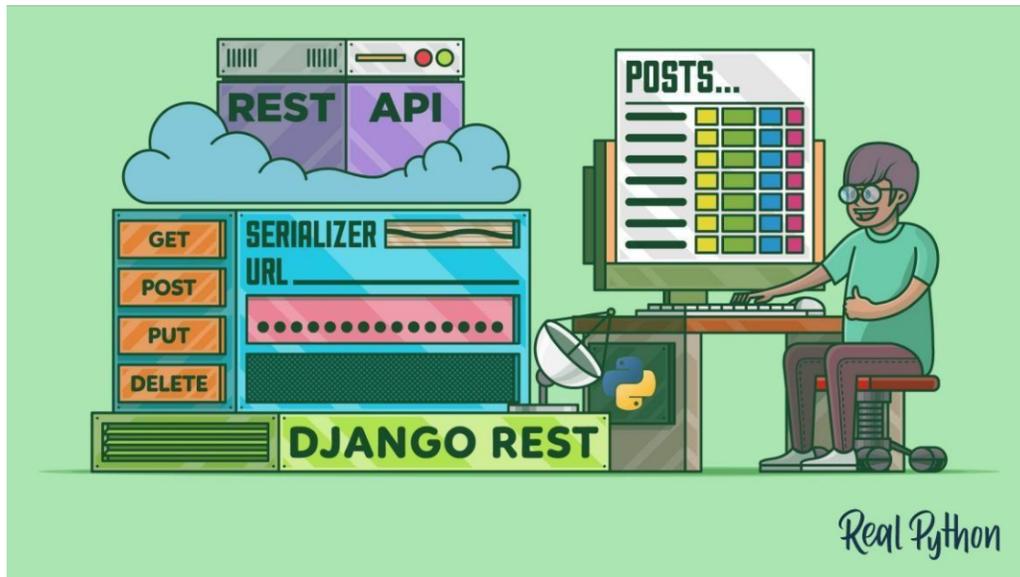
negocios basados en reglas, de esta forma las tareas repetitivas pueden solucionarse usando Rpa. RPA permite a los usuarios crear bots que son capaces de imitar e interactuar con cualquier aplicación o sistema de la misma forma que haría un ser humano, con la excepción de que los bots RPA pueden trabajar de forma continua durante todo el día sin parar, mucho más rápido y con 100 % de precisión y estabilidad. Por otra parte, este tipo de bots deben ser monitoreados para que no presenten ningún bug o fallo.(Whiteside et al., 2022)

### ***2.1.3 Django***

Django es un framework específico realizado para integrarlo a Python. que se puede utilizar para poder desarrollar aplicaciones web usando Python.

Los software o aplicaciones web tienen varias funciones comunes, como la autenticación, la recuperación de información y la administración de cookies.

Django facilita este trabajo y agrupa diferentes herramientas o funciones reutilizables para poder desarrollar de forma eficaz y eficiente, lo que permite disminuir tiempo de desarrollo y mejorar la calidad del código. (Khalid Eid Abougamea & Chun Lim, 2023)

**Figura 4** Rest API

Autor: (Python, 2014)

#### ***2.1.4 Jinja***

Jinja es un motor de plantillas para Python. Permite separar la lógica del programa de la presentación, lo que resulta en un código más limpio y fácil de mantener. Jinja se utiliza comúnmente en frameworks web como Flask y Django para renderizar HTML dinámico.

Además, Jinja admite características avanzadas como herencia de plantillas, filtros personalizados y macros. (Ramachandran et al., 2018)

### ***2.1.5 Bootstrap***

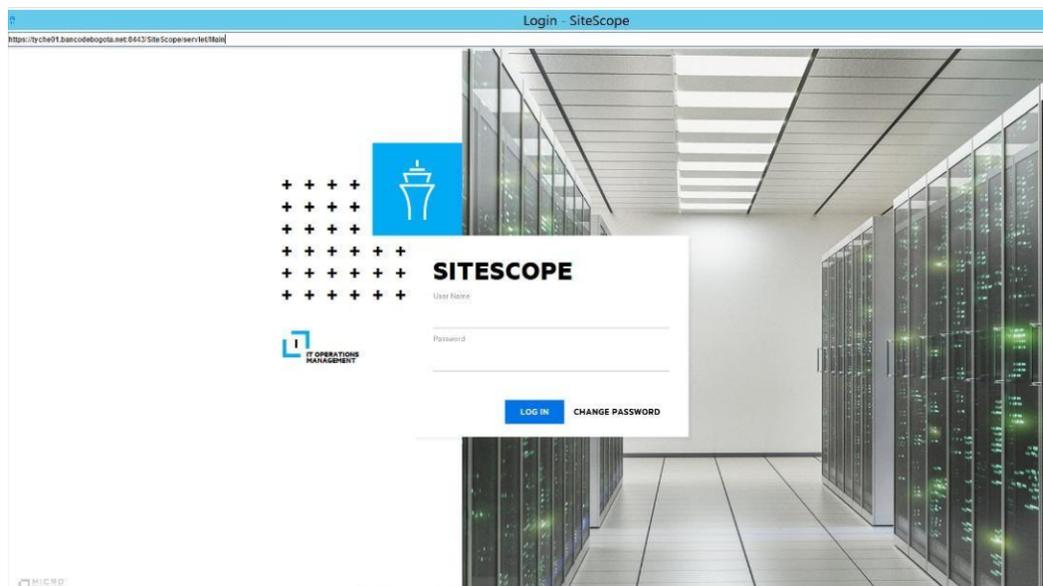
Bootstrap es un framework de código abierto para css, este framework combina css y js para decorar y mejorar la estética de una página HTML, Bootstrap nos proporciona distintos componentes que facilitan al usuario crear menús, formatos, botones, alertas, modales entre otros. Además de todas las características que ofrece Bootstrap una de las funciones principales del framework es que permite construir sitios web 100% responsive para todo tipo de dispositivos móviles.(Pérez-Rave et al., 2022)

### ***2.1.6 SiteScope***

SiteScope es una herramienta de monitoreo de aplicaciones y servidores desarrollada por Micro Focus. Permite monitorear la disponibilidad, el rendimiento y la integridad de los sistemas informáticos, incluyendo servidores, aplicaciones web, bases de datos y dispositivos de red. SiteScope utiliza una variedad de protocolos de monitoreo, como SNMP, HTTP, JDBC y JMX, y proporciona alertas en tiempo real y análisis detallados para ayudar a los administradores de sistemas a identificar y solucionar problemas de manera eficiente. SiteScope es utilizado por muchas empresas para garantizar la disponibilidad y el rendimiento de sus sistemas críticos. (Proceedings of the 31st annual ACM SIGUCCS fall conference 2003)

---

**Figura 5** Inicio SITESCOPE



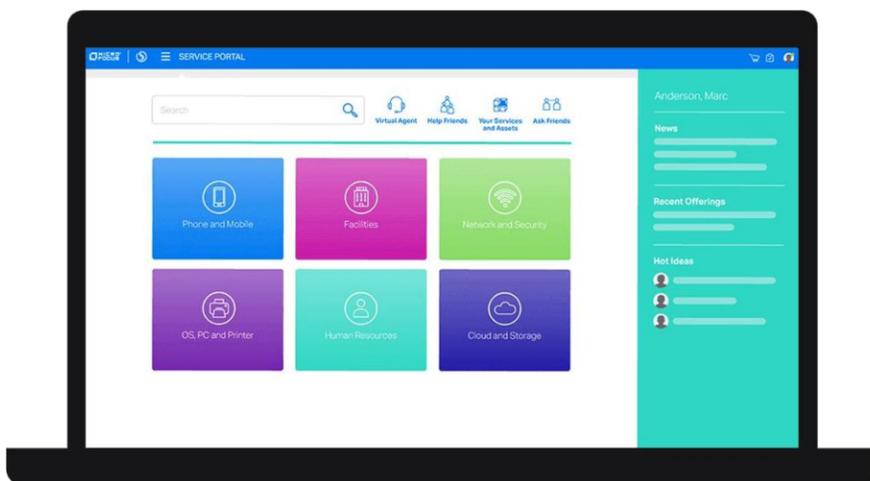
Autor: (De Bogotá, s. f. 2023)

### 2.1.7 Smax

SMAX es una plataforma de gestión de servicios empresariales desarrollada por Micro Focus. Proporciona una solución integral para la gestión de servicios de TI, la gestión de servicios empresariales y la gestión de servicios de negocio. SMAX permite a las empresas gestionar eficazmente sus servicios y procesos empresariales, automatizar tareas y mejorar la colaboración entre los equipos. La plataforma utiliza tecnologías de inteligencia artificial y aprendizaje automático para mejorar la eficiencia y la calidad de los servicios prestados( SMAX: Solución de gestión de servicios de TI (ITSM), n.d. 2023). SMAX se puede personalizar para satisfacer las necesidades específicas de cada empresa y está diseñado para ser escalable y adaptable a medida que las necesidades

empresariales cambian con el tiempo.

**Figura 6** SMAX



Autor: (SMAX: Solución de gestión de servicios de TI (ITSM), n.d.2023)

### ***2.1.8 Apache***

Apache es un servidor web de código abierto que se utiliza en ingeniería de sistemas para alojar y servir sitios web y aplicaciones web. Es uno de los servidores web más populares del mundo y es utilizado por millones de sitios web en todo el mundo. Apache se puede ejecutar en una variedad de sistemas operativos, incluyendo Linux, Unix, Windows y macOS. Además de alojar sitios web y aplicaciones web, Apache también puede ser utilizado para servir archivos y datos a través de protocolos como FTP y WebDAV. Apache es altamente personalizable y se puede configurar para satisfacer las necesidades específicas de cada sitio web o aplicación web (Swag, n.d. 2023).

**Figura 7 Apache**



Autor: (Swag, n.d. 2023)

### ***2.1.9 SQL***

Sql o Structured Query Language o en español, lenguaje de consulta estructurado.

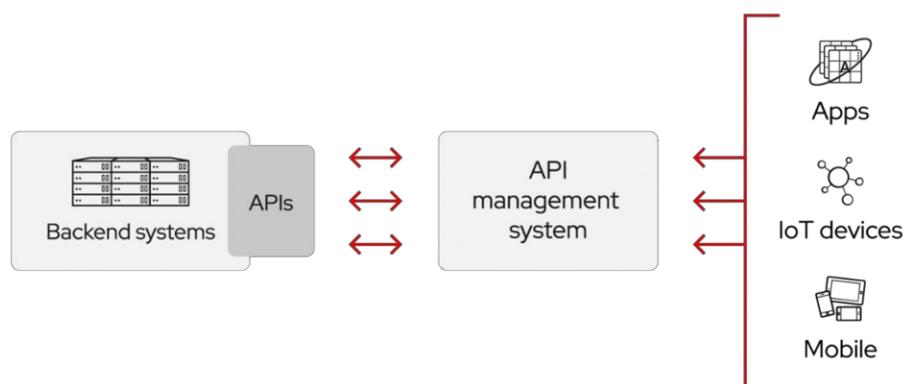
Es un tipo de lenguaje de programación que le permite al usuario crear, descargar, manipular información de una base de datos. También permite realizar relaciones entre las bases de datos y realizar cálculos avanzados, es uno de los lenguajes más utilizados por las empresas para almacenar datos y poder manipularlos.

### ***2.1.10 API***

Las Api's son un conjunto de protocolos que se incorporan en una aplicación y permiten que los productos y servicios se comuniquen con otros, nos permiten llamar datos sin la necesidad de entender dónde están implementados, gracias a esto un api otorga la

oportunidad de innovar con los datos que almacena nos permite crear nuevas herramientas, nuevas aplicaciones (productos) o gestionar las ya existentes (¿Qué es una API y cómo funciona?, n.d. 2023).

**Figura 8** funcionamiento de un api



Autor: (¿Qué es una API y cómo funciona?, n.d. 2023)

### **2.1.11 metodología scrum**

La metodología de Scrum es un marco de trabajo ágil utilizado en el desarrollo de software y otros proyectos complejos. Se enfoca en el trabajo en equipo, en la comunicación y la entrega continua de valor al cliente (Iskandar et al., 2023).

Scrum se basa en equipos autoorganizados y multidisciplinarios que trabajan en sprints (iteraciones cortas de tiempo) para completar objetivos específicos. Cada sprint comienza con una reunión de planificación donde el equipo define las tareas a realizar y el tiempo estimado para terminar cada tarea. Durante el sprint, el equipo se reúne todos los días, esta reunión es denominada como una reunión diaria de scrum para discutir el

progreso y cualquier problema que surja. Al final del sprint, el equipo se reúne en una revisión de sprint para demostrar el trabajo completado y recibir retroalimentación del cliente o del equipo de gestión.

Para utilizar Scrum, es importante tener un equipo comprometido y una mentalidad ágil. El equipo debe estar dispuesto a colaborar y trabajar juntos para alcanzar los objetivos establecidos. También es importante tener un product owner (dueño del producto) que pueda priorizar las tareas y proporcionar una visión clara del producto. Además, se necesitan herramientas para gestionar el flujo de trabajo, como tableros Kanban o software de gestión de proyectos ágiles (Iskandar et al., 2023).

En resumen, Scrum es una metodología ágil que se enfoca en la entrega continua de valor al cliente mediante equipos autoorganizados y multidisciplinarios que trabajan en sprints. Para utilizar Scrum, se necesita un equipo comprometido, un product owner y herramientas adecuadas para la gestión del flujo de trabajo.

**Figura 9** Desarrollo scrum



Autor: (“Metodología Scrum, una herramienta útil para agilizar tus proyectos,” 2020)

## **2.2 Marco Legal**

### ***2.2.1 Iso 27001***

La norma iso 27001 permite que las empresas puedan evaluar y tratar riesgos de seguridad en la infraestructura de su información, es una norma de internacional de seguridad de información que establece controles para garantizar la confidencialidad de las organizaciones, permite asegurar datos propios y de los clientes.

Se establecen 3 objetivos principales en la norma ISO 27001

Confidencialidad

Disponibilidad

Integridad

### ***2.2 .2 Ley 1341 de 2009***

Esta ley establece el marco regulatorio público para las tecnologías de la información y las comunicaciones en Colombia. Entre otras cosas, establece los derechos y obligaciones de los clientes y proveedores de servicios de tecnología, así como las condiciones para el acceso a la información y la protección de datos personales (Ley 1341 de 2009 - Gestor

Normativo, n.d.).

### ***2. 2.3 Ley 1581 de 2012***

Esta ley regula la protección de datos personales en Colombia. Establece que cada individuo tiene un régimen de protección de datos, las obligaciones de los responsables del tratamiento de dichos datos y las sanciones por incumplimiento(Ley 1581 de 2012 - Gestor Normativo, n.d.).

### ***2.2.4 Resolución 3066 de 2011:***

Esta resolución establece los requisitos técnicos y administrativos para la implementación de sistemas de gestión de seguridad de la información en Colombia. Establece los procedimientos para la evaluación y certificación de dichos sistemas, protege los derechos de los usuarios de los servicios de comunicaciones

### ***2.2.5 ley 23 de 1982 Ver la Ley 44 de 1993, Ver el Decreto Nacional 1474 de 2002***

En Colombia esta ley establece los derechos de autor en materia de software, es la protección de los derechos de autor y aplica para una persona natural o jurídica, establece que los derechos sobre programas de computación tienen 50 años de duración a partir de enero del siguiente año en el que el software fue creado, también habla sobre la distribución o reproducción del software, que deben contar con la autorización del autor

(Ley 23 de 1982 - Gestor Normativo, n.d.).

En resumen, el desarrollo de software en Colombia está regulado por varias leyes y normativas que establecen los derechos y deberes de los usuarios y proveedores de tecnología, la protección de datos personales, el comercio electrónico, la formación técnica y tecnológica y la seguridad de la información. Es importante que los desarrolladores de software estén

familiarizados con estas leyes y normativas para garantizar el cumplimiento de las mismas en sus proyectos.

### 2.3 Tabla herramientas usadas

Tabla 4 Herramientas más usadas

	Framework desarrollado por Python
Django	
jinja	Framework desarrollado por Python
Selenium	Herramienta de automatización que permite hacer testing web
Re	Módulo de Python que proporciona operaciones con expresiones regulares
Api sitescope	Herramienta que permite comunicación mediante un conjunto de protocolos

---

beautifulsoup	Librería de Python que permite extraer información en formato HTML o XML
Random	Modulo permite generar números aleatorios
Apache	Servidor http código abierto
Transformer	Permite entrenamiento IA
Bootstrap	Biblioteca multiplataforma para diseño desitios
pandas	Librería de Python para manejo de datos

---

### 3. Capítulo 3 informe de cumplimiento de trabajo

Para llevar a cabo la creación de la herramienta de automatización el grupo nasa debe entender el proceso por el que pasa un colaborador del banco para realizar una actividad de monitoreo en el banco y cómo funcionan las herramientas sitescope y smax.

#### 3.1 identificar el Proceso a seguir en el banco de Bogotá para solicitar un monitoreo de cualquier tipo.

Para llevar a cabo un monitoreo de cualquier tipo en el banco, primero el colaborador debe recibir un smax a la célula nasa como el siguiente:

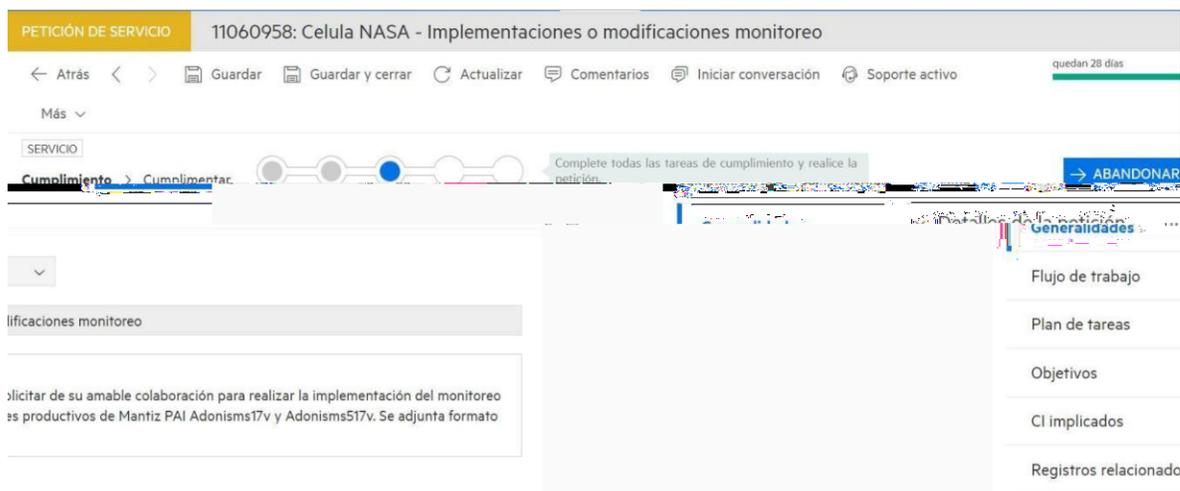
**Figura 10** solicitud monitoreo

ID	Título	Prioridad	SLT	Fecha/hora de	Destinatario de la	Estado de cl	Asignación	Grupo de asignación2	Asignatario
1004	Celula NASA - Implementaciones o...	Baja	Yonadi Lopez		Alonso Carbonero	Ningun...	Centro de...	DIO_NASA_MONIT...	
9978	Celula NASA - Implementaciones o...	Baja	Quenda Torres		Piñeros Corre	Ningun...	Centro de...	DIO_NASA_MONIT...	
9956	Celula NASA - Implementaciones o...	Baja	Blanca Torres		Pava Zambano	Ningun...	Centro de...	DIO_NASA_MONIT...	
9894	Celula NASA - Implementaciones o...	Baja	Yonadi Lopez		Vera Vargas	Ningun...	Centro de...	DIO_NASA_MONIT...	
9844	Celula NASA - Implementaciones o...	Baja	Yonadi Lopez		Vera Vargas	Ningun...	Centro de...	DIO_NASA_MONIT...	
8260	Celula CRE - Implementaciones o...	Baja	2 meses transcurridos		Cruz Narvaez	Ningun...	Centro de...	DIO_NASA_MONIT...	
8180	Celula CRE - Implementaciones o...	Baja	2 meses transcurridos		Garzon Linares	Ningun...	Centro de...	DIO_NASA_MONIT...	
8087	Celula CRE - Implementaciones o...	Baja	2 meses transcurridos		Vera Vargas	Ningun...	Centro de...	DIO_NASA_MONIT...	

Nota. Creación propia (2023)

Esta solicitud indica que maquina requiere un monitoreo y que tipo de monitoreo como se muestra a continuación

**Figura 11** tipo de monitoreo



Nota. Creación propia (2023)

Con la solicitud que se muestra en la figura anterior, se adjunta un Excel que indica el ítem a monitorear una descripción del monitor, unos umbrales para que reporte cuando se genera ese tipo de error, la frecuencia de monitoreo y la célula a la que se deben notificar los errores.

Adjunto una figura que muestra ejemplo del tipo de Excel comentado.

**Figura 12** Excel ejemplo monitoreo

Infraestructura Base		
Item a Monitorear	Monitor / Descripción	Umbral Sugerido
Interfaz de red	Disponibilidad de la Interfaz de Red	Ping de disponibilidad. Máxima respuesta (milisegundo) > 128 (warning) > 1000 critical
Memoria Swap Unix	Cantidad de memoria que está actualmente en uso en un servidor.	CRITICAL > 20% WARNING > 10%
Memoria Fisica Unix	Cantidad de memoria que está actualmente disponible en un servidor.	CRITICAL <= 512 MB WARNING > 1024 MB
Memoria Fisica AIX	Script que mide la memoria física- smmon -G -O unit=MB	CRITICAL <= 512 MB WARNING <1024 MB

sdiaz3:  
No aplica para Red Hat

	Frecuencia Monitoreo	Notificación correo electrónico
>	5 MINUTOS	<a href="#">DIO CELULA BACKENDS APP</a> <a href="#">DIO CELULA BACKENDS MQ CICs Sos</a>

Nota. Creación propia (2023)

Con las especificaciones dadas como se muestra en las figuras anteriores, el colaborador ingresa a la herramienta de monitoreo y crea los monitoreos específicos de la siguiente forma

primero se asegura que existe una conexión con dicho servidor, e identifica si el servidor es Linux o Windows.

Para asegurarse que existe Conexión con el servidor, el colaborador envía un ping y revisa el comportamiento.

**Figura 13** conexión ping servidor

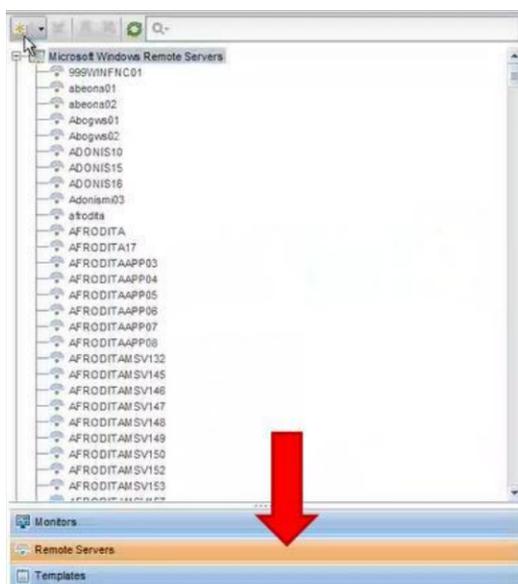
```
C:\Users\jcortes1>ping NEPTUNOLXU046
Pinging NEPTUNOLXU046.bancodebogota.net [10.86.101.42] with 32 bytes of data:
Reply from 10.86.101.42: bytes=32 time=2ms TTL=58
```

Nota. Creación propia (2023)

Si no hay ping disponible el usuario debe comunicar a la célula que el servidor no es visible debe otorgar permisos de accesos a thyche01

Si existe ping y es un servidor Windows el usuario hace lo siguiente. Da click en remote servers dentro de la aplicación de sitescope

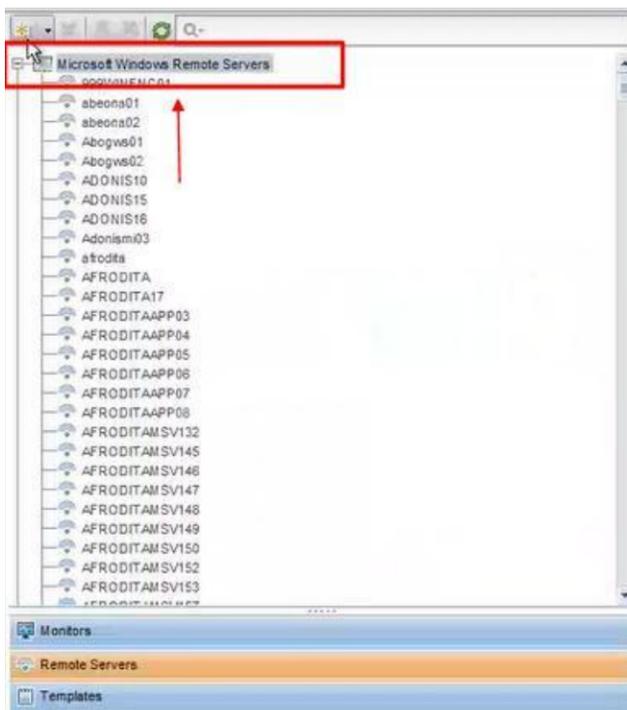
**Figura 14** remote server



Nota. Creación propia (2023)

Luego da click en la carpeta Microsoft Windows Remote Servers

**Figura 15** explicación uso de sitescope



Nota. Creación propia (2023)

Luego da click en New Microsoft Windows Remote Servers

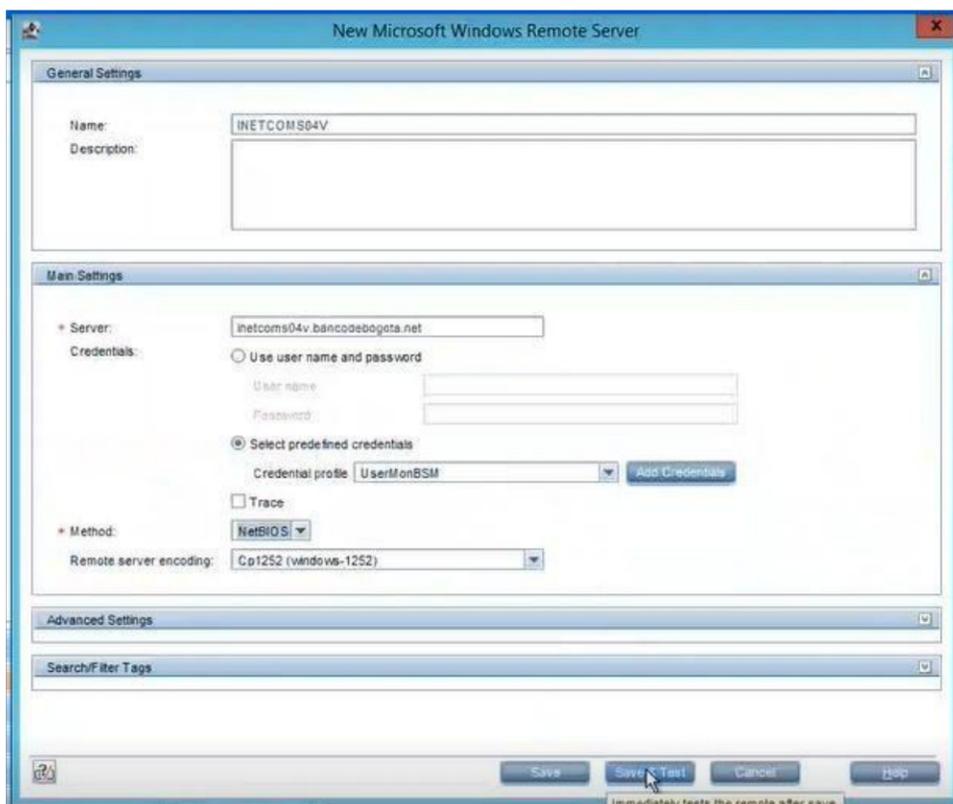
**Figura 16** explicación uso de sitescope

Server	Status	Last test	Operating system	Method
1099WINF1C01	Connection successful	12/3/2020 2:40 PM	Windows	NetBIOS
abeona01	Connection successful	6/15/2019 5:19 PM	Windows	WMI
abeona02	Connection successful	6/15/2019 5:22 PM	Windows	WMI
Abogvs01	Conexión correcta	10/16/2019 4:31 PM	Windows	NetBIOS
Abogvs02	Conexión correcta	10/16/2019 4:35 PM	Windows	NetBIOS
ADONIS10	unknown		Windows	NetBIOS
ADONIS15	unknown		Windows	NetBIOS
ADONIS16	unknown		Windows	NetBIOS
Adonism03	Connection successful	11/15/2016 8:49 AM	Windows	NetBIOS

Nota. Creación propia (2023)

Se abrirá la siguiente ventana

**Figura 17** explicación uso de sitescope



Nota. Creación propia (2023)

En esta pestaña el usuario tiene que ingresar el nombre del servidor, se recomienda un estándar de ingresar este nombre en mayúsculas, pero él puede ingresarlo como mejor se ajuste a la célula.

Si el desea puede agregar una descripción del servidor

**Figura 18** explicación uso de sitescope



The image shows a screenshot of a 'General Settings' dialog box. It has a title bar with the text 'General Settings' and a close button. The dialog contains two input fields: 'Name:' and 'Description:'. The 'Name:' field is filled with the text 'INETCOMS04V'. The 'Description:' field is currently empty.

Nota. Creación propia (2023)

Luego el usuario ingresa los main settings, ingresa las credenciales del servidor.

Luego selecciona las credenciales del perfil, para el caso de Windows el Usuario maestro es UserMonBSM

**Figura 19** explicación uso de sitescope

Main Settings

**Main Settings**

\* Server:

Credentials:

Use user name and password

User name

Password

Select predefined credentials

Credential profile

Trace

\* Operating system:

\* Method:

Prompt:

Login prompt:

Password prompt:

Secondary prompt:

Mask secondary response

Server:

Credentials:

Use user name and password

User name

Password

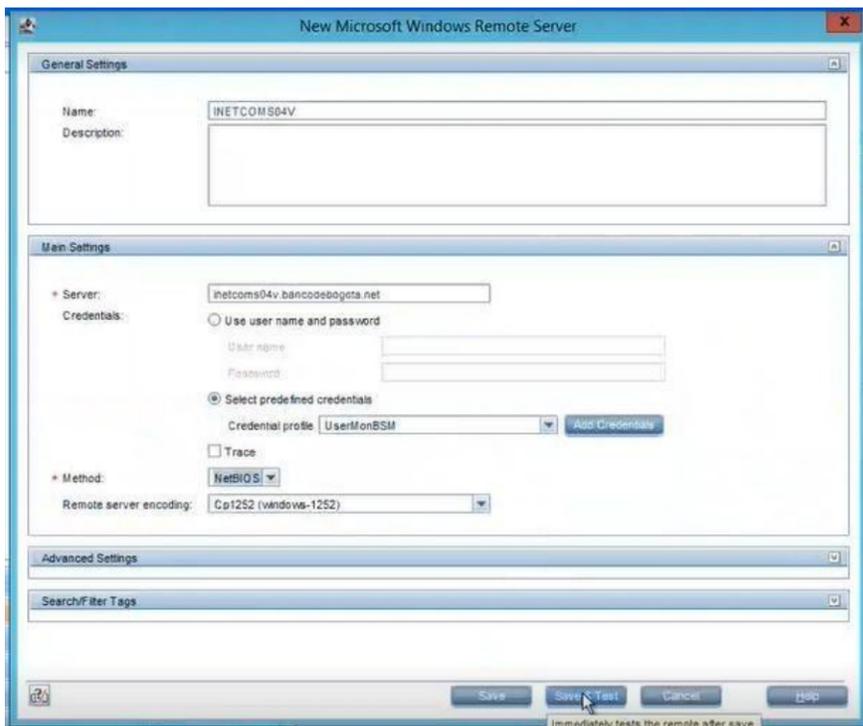
Select predefined credentials

Credential profile

Nota. Creación propia (2023)(2023)

Por último, el usuario selecciona el método, el recomendable es NetBios ya que es el que nos permite traer más información, y deja el resto de información por defecto

Figura 20 explicación uso de sitescope



Nota. Creación propia (2023)(2023)

Da clic en Save & Test y salta la notificación de que se creó una conexión con el servidor como la siguiente

Figura 21 explicación uso de sitescope



Nota. Autor (2023)

Luego de crear los servidores el usuario debe crear el monitoreo del servidor por ejemplo

Para crear un monitoreo de cpu sobre infraestructura él debe hacer los siguientes pasos

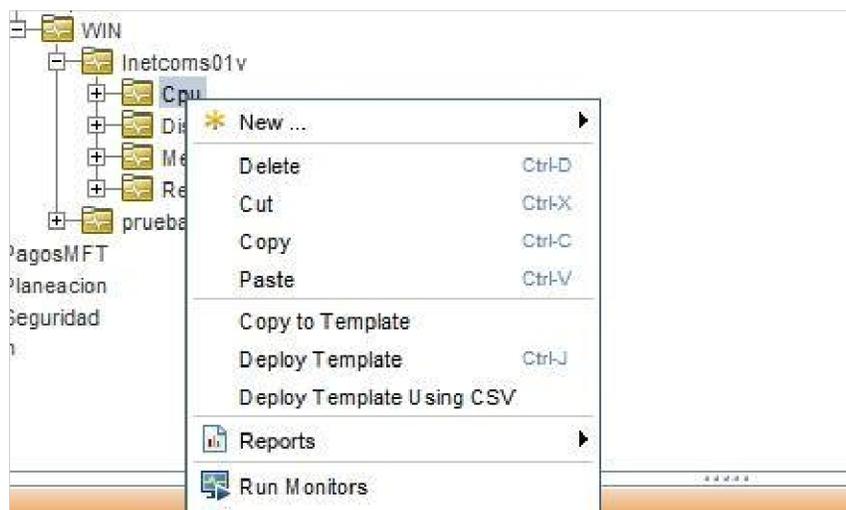
Diríjase a la arpeta INF en la célula que quiera crear el monitoreo para este ejemplo se hará enla célula nasa

Dependiendo del tipo de servido LINUX, AIX, Windows,el crea el monitoreo en la carpeta según el servidor en este caso se creará un monitoreo CPU de un servidor

Windows

El usuario se dirige a la carpeta en la que se va a crear el monitoreo y hace clic derecho en lacarpeta como se muestra en la siguiente Figura

**Figura 22** explicación uso de sitescope

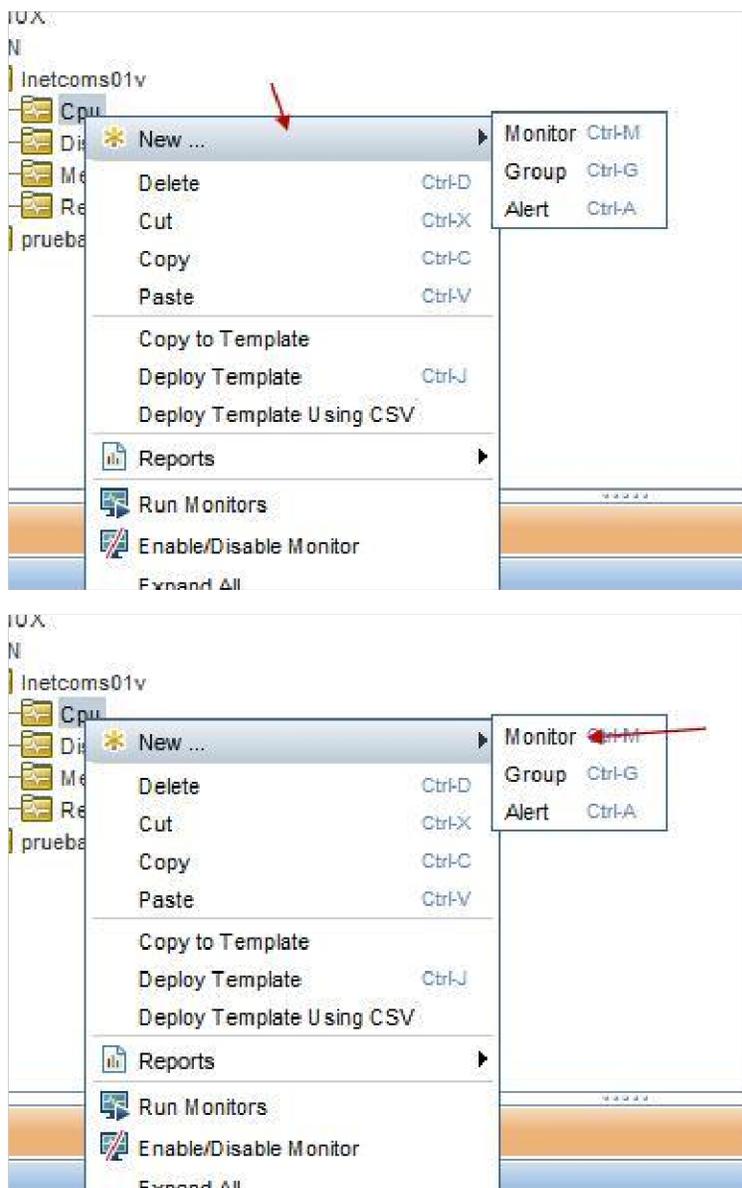


Nota. Creación propia (2023)

Se abrirá una ventana hace clic en NEW

Al hacer clic se abre una pestaña aquí hace clic en MONITOR

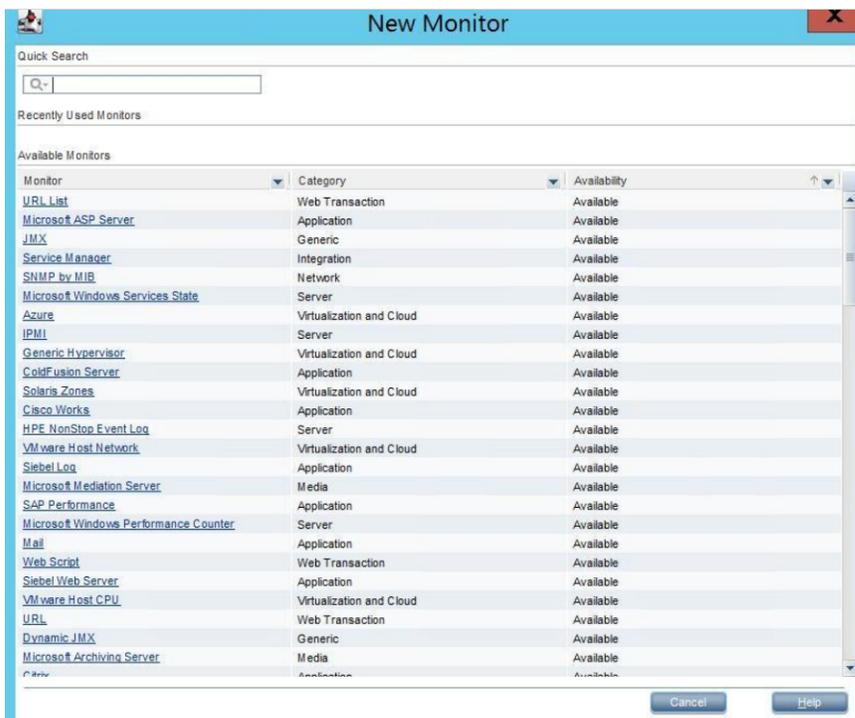
Figura 23 explicación uso de sitescope



Nota. Creación propia (2023)

Al hacer clic en MONITOR se abrirá el siguiente buscador

**Figura 24** explicación uso de sitescope



Nota. Creación propia (2023)

Aquí el usuario busca el monitoreo de CPU que va hacer

\* Hace clic en CPU de categoría SERVER Al hacer clic se abrirá la siguiente ventana

Empieza a llenar la información del monitoreo

Empieza por el nombre que se encuentra en la parte superior de la ventana

El nombre que se le asigna al monitoreo debe tener las siguientes características: Primero como es un monitoreo de infraestructura debe iniciar como INF Segundo debe tener el nombre de la célula a la que pertenece en este caso NASATercero debe tener el tipo de

servidor es en este caso Win

Cuarto debe decir que tipo de monitoreo es en este caso CPU y el nombre del servidor al que apunta en este caso inetcoms01v

Quedará de la siguiente forma INF\_Nasa: WIN - CPU on Inetcoms01v

El usuario escribe el nombre y opcional una descripción queda de la siguiente manera

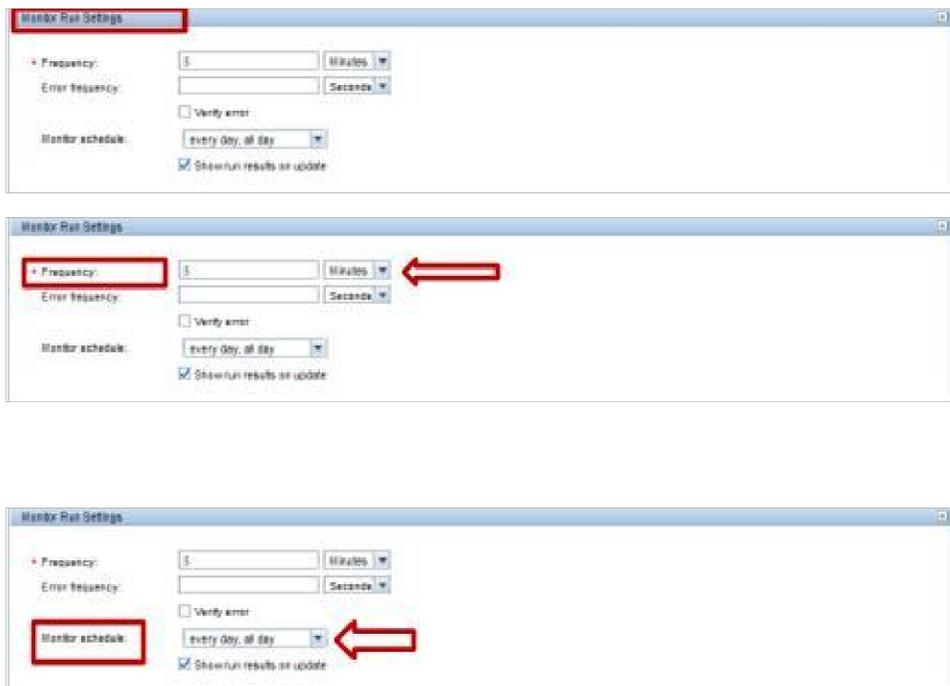
**Figura 25** explicación uso de sitescope

General Settings	
Name:	INF_Nasa: WIN - CPU on Inetcoms01v
Monitor description:	
Report description:	
Instruction text:	Si la condición indica "n/a" quiere decir que por alguna razón el monitor de consulta no alcanzó el servidor. Posibles causas: Consumo de CPU, Memoria, red etc. Por favor validar el servidor.

Nota. Creación propia (2023)

Ahora el usuario dirige a la pestaña MONITOR RUN SETTINGS Selecciona la frecuencia en la que el monitor será monitoreado Selecciona el horario que será monitoreado

**Figura 26** explicación uso de sitescope



Nota. Creación propia (2023)

Ahora nos el usuario se dirige a la pestaña CPU MONITOR SETTINGS Selecciona el nombre del servidor en este caso es inetcoms01v

**Figura 27** explicación uso de sitescope



Nota. Creación propia (2023)

Por último, el usuario se dirige a THRESHOLD SETTINGS

Escoge las métricas según los requerimientos que previamente le enviaron, selecciona cuando está en crítico cuando está en peligro y cuando el monitor esté funcionando correctamente

**Figura 28** explicación uso de sitescope

Threshold Settings

If unavailable: Set monitor status according to Thresholds

Default status: Good

On internal error: Set monitor status according to Thresholds

Add Default Thresholds Remove Default Thresholds

Error if

Condition	Operator	Value	Schedule	Indicator State and Severity
utilization	==	'n/a'	every day, all day	Bottlenecked(Critical)
utilization	>=	98	every day, all day	Bottlenecked(Critical)

Warning if

Condition	Operator	Value	Schedule	Indicator State and Severity
utilization	>=	95	every day, all day	Busy(Minor)

Good if

Condition	Operator	Value	Schedule	Indicator State and Severity
utilization	<	95	every day, all day	Normal(Normal)

Threshold Settings

If unavailable: Set monitor status according to Thresholds

Default status: Good

On internal error: Set monitor status according to Thresholds

Add Default Thresholds Remove Default Thresholds

Error if

Condition	Operator	Value	Schedule	Indicator State and Severity
utilization	==	'n/a'	every day, all day	Bottlenecked(Critical)
utilization	>=	98	every day, all day	Bottlenecked(Critical)

Warning if

Condition	Operator	Value	Schedule	Indicator State and Severity
utilization	>=	95	every day, all day	Busy(Minor)

Good if

Condition	Operator	Value	Schedule	Indicator State and Severity
utilization	<	95	every day, all day	Normal(Normal)

Nota. Creación propia (2023)

Hace clic en guardar y testear

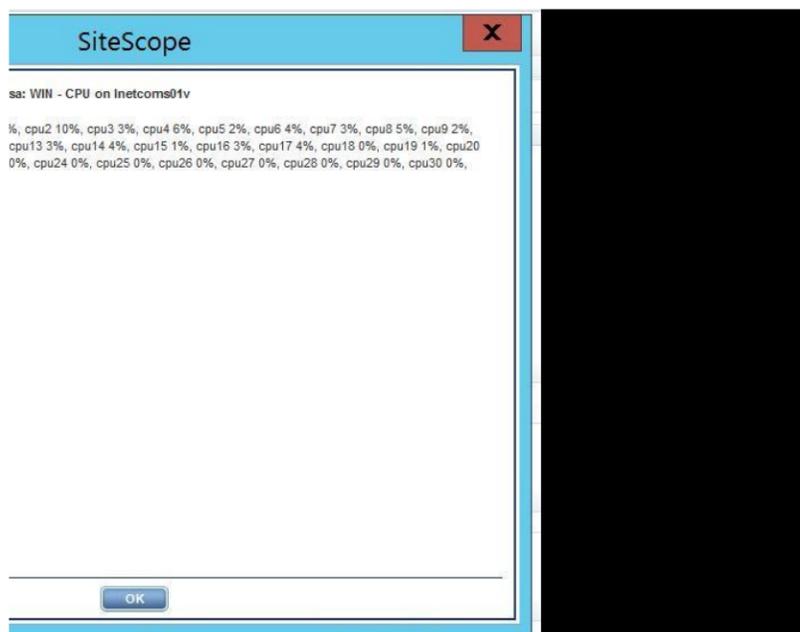
**Figura 29** explicación uso de sitescope



Nota. Creación propia (2023)

Se abrirá la siguiente ventana, que indica en qué estado está el monitor

**Figura 30** explicación uso de sitescope

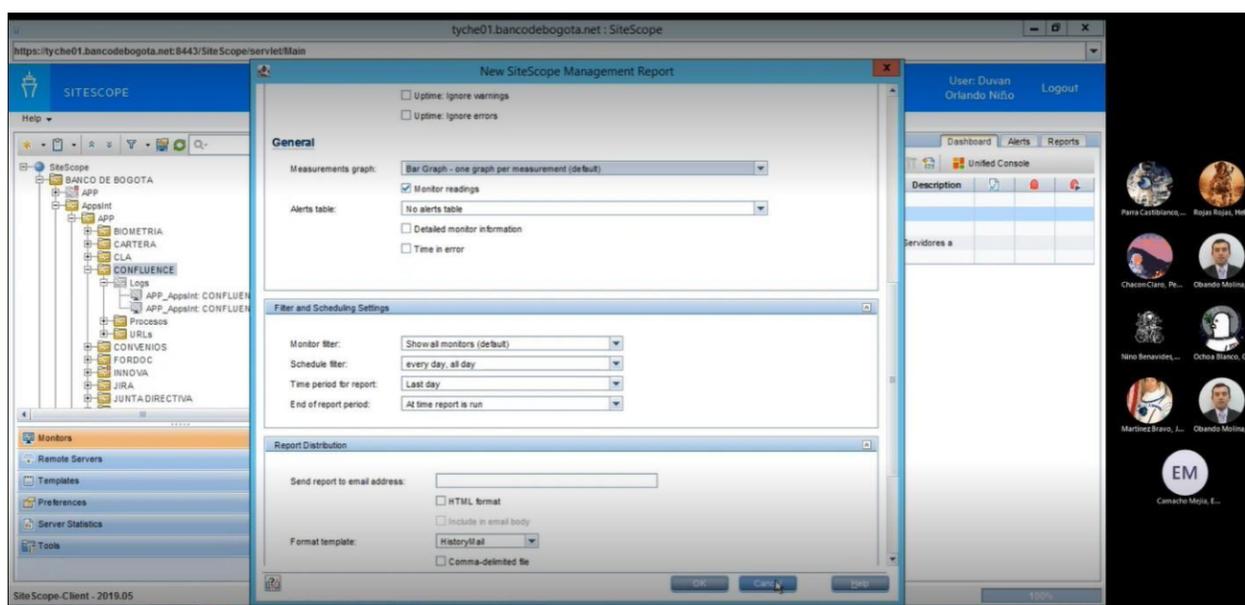


Nota. Creación propia (2023)

Después de que se identificó que el proceso a seguir consumía mucho tiempo para un colaborador del banco, se tomó la decisión de automatizar este proceso.

Por ende, el primer paso a realizar fue hacer un refuerzo de conocimiento en la herramienta de sitescope para tomar el mejor camino a automatizar el proceso.

**Figura 31** explicación uso de sitescope



Nota. Creación propia (2023)

Luego la primera tarea y antes del desarrollo de las herramientas de automatización, es importante la creación de un manual de sitescope que explicara detalladamente cómo funciona sitescope, que rutas sigue y como acceder a esta herramienta.

**Figura 32** manual uso de sitescope



Nota. Creación propia (2023)

Después de que el grupo nasa entendió el funcionamiento de la herramienta para crear distintos monitoreos y cómo funciona sitescope, se pasó a la siguiente fase.

### **3.2 Realizar búsqueda y revisión de tecnologías para ver cómo se puede implementar una herramienta de automatización para los colaboradores del banco.**

Al realizar la búsqueda se tuvieron en cuenta muchas herramientas, el grupo optó por crear una página web con Python que va a servir como vista, esta página web estar diseñada con Django y Jinja 2, con estilos propios del banco mezclados con estilos de

Bootstrap, esta página usara el api de sitescope para hacer las tareas repetitivas de crear monitoreos, traer reportes, enviar información por correo etc.

Por otra parte, para automatizar las solicitudes de smax se planteó crear una IA que pueda interpretar textos y cerrar los casos que se hagan en smax.

Una vez definidas las herramientas a usar y como se va a crear la herramienta de automatización, se dio pie al siguiente objetivo a realizar

### **3.3 Validar si la solución encontrada cumple con los distintos requerimientos.**

La solución que se ofreció al banco fue aceptada y se ajustó a varios requerimientos que solicitaron, uno de los más importantes es que a la aplicación solo deben tener acceso los usuarios del banco, por otra parte, nos suministraron un servidor nuevo para comenzar el desarrollo en ese servidor e instalar las herramientas necesarias.

Al validar esta información se da comienzo al siguiente objetivo

### **3.4 Fortalecer mis conocimientos según los cursos que me entrego la empresa.**

Antes de comenzar el desarrollo, el banco ofreció varios cursos de desarrollo en Python,

estos cursos deben ser desarrollados al completo para poder dar comienzo a programar

**Figura 33** recursos estudio



Nota. Creación propia (2023)

Este curso contiene todo lo que necesitamos saber para desarrollar la herramienta de automatización.

al finalizar los estudios se da paso al siguiente objetivo

### **3.5 Estructurar el plan de implementación de la herramienta de automatización**

Luego que el grupo nasa entendió cómo funciona sitescope y entendió como se va a desarrollar el proyecto, por ende, realizaron los cursos para reforzar el conocimiento.

Esto dio paso a que el grupo nasa seleccione la metodología scrum para desarrollar la entrega del proyecto, primero se estableció un daily a las 8 am en el que se entregan los avances de cada día y se presentan bloqueantes que se tengan sobre la herramienta que se está creando, cada 2 semanas se hará entrega del proyecto frente a los usuarios finales del software y se añadirán los cambios y correcciones que soliciten.

**Figura 34** Reuniones diarias

	29 Lunes	30 Martes	31 Miércoles	01 Jueves	02 Viernes
6 a. m.					
7 a. m.					
8 a. m.	Daily Celula NASA Micr	Daily Celula NASA Microsoft Teams Meeting Rojas Rojas, H	Daily Celula NASA Microsoft Teams Meeting Roja:	Daily Celula NASA Micr	Daily Celula NASA Micr

Nota. Creación propia (2023)

por otra parte, se instalaron todos los frameworks y herramientas a utilizar como el visual code en el starfoxms05 (el servidor donde se alojará la aplicación), se comenzará a automatizarla herramienta sitescope y finalizará con el desarrollo de la IA para smax

Ya con las herramientas y el plan de trabajo por desarrollar se comienza el siguiente objetivo

## 3.6 Desarrollar la herramienta que permita la automatización de SITESCOP

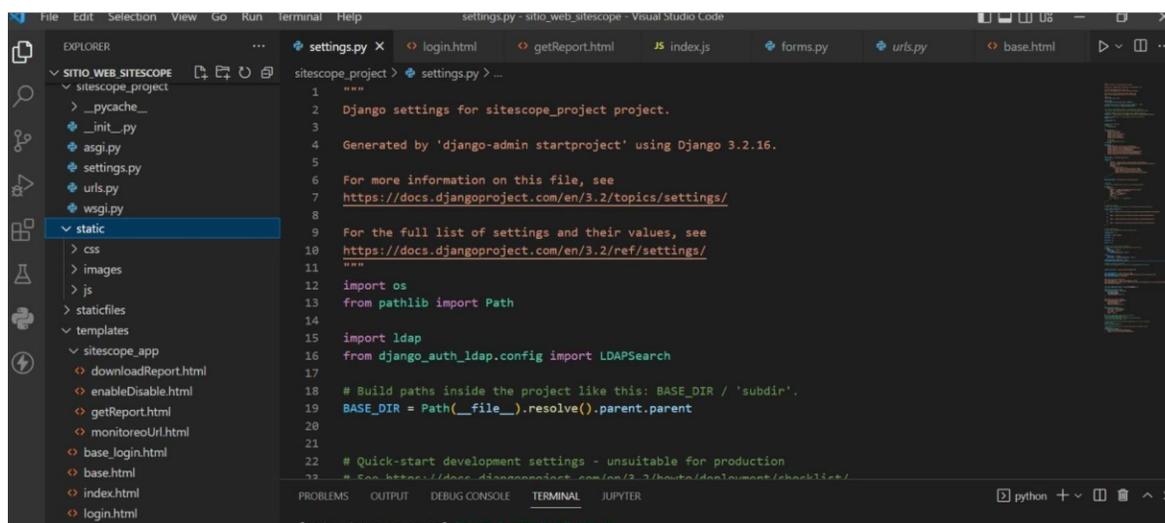
### 3.6.1 creación de proyecto en Django.

Se inicializa el proyecto básico en Django y se ejecuta por el puerto que se encuentra por defecto, usando visual studio code para facilitar la forma de escribir código.

Se inicio el proyecto con la creación de la carpeta static donde se encontrará el ccs, el js y las imágenes, por otra parte, se crea una carpeta de templates donde escribiremos el código HTML combinado con Python.

Por último, se crearon las vistas

**Figura 35** Proyecto Django



Nota. Creación propia (2023)

Una vez creada la estructura el grupo nasa comienza con el desarrollo del siguiente

objetivo

### ***3.6.2 Desarrollo vistas HTML***

Se crean las vistas HTML donde el usuario pueda navegar en la página y pedirle al api que le entregue la información requerida.

Para esta primera entrega la herramienta usando el api puede generar reporte de los monitoreos, puede habilitar y deshabilitar y puede crear monitoreos de url.

Django nos permite programar usando componentes como en angular, React, Vue, así que se crea un HTML base donde se encontraran las pestañas para traer reportes, para habilitar y deshabilitar y para crear los servidores, en el HTML base cargamos la carpeta static Bootstrap y los js necesarios para el desarrollo de esta herramienta e invocamos el componente HTML base en los otros HTML para que tomen estilos y el header

Figura 36 vista HTML

The image displays a web application's HTML structure. At the top, a file explorer shows a directory named 'sitescope\_app' containing several HTML files: 'downloadReport.html', 'enableDisable.html', 'getReport.html', 'monitoreoUrl.html', 'base\_login.html', 'base.html' (selected), and 'login.html'. Below this, two code editors are visible. The first editor shows the content of 'base.html', which includes a DOCTYPE declaration, a head section with meta tags for charset, http-equiv, and viewport, and several link tags for CSS files. The second editor shows the content of 'getReport.html', which includes a title tag, a link tag for a CSS file, and a list element.

```

sitescope_app
├── downloadReport.html
├── enableDisable.html
├── getReport.html
├── monitoreoUrl.html
├── base_login.html
├── base.html
└── login.html

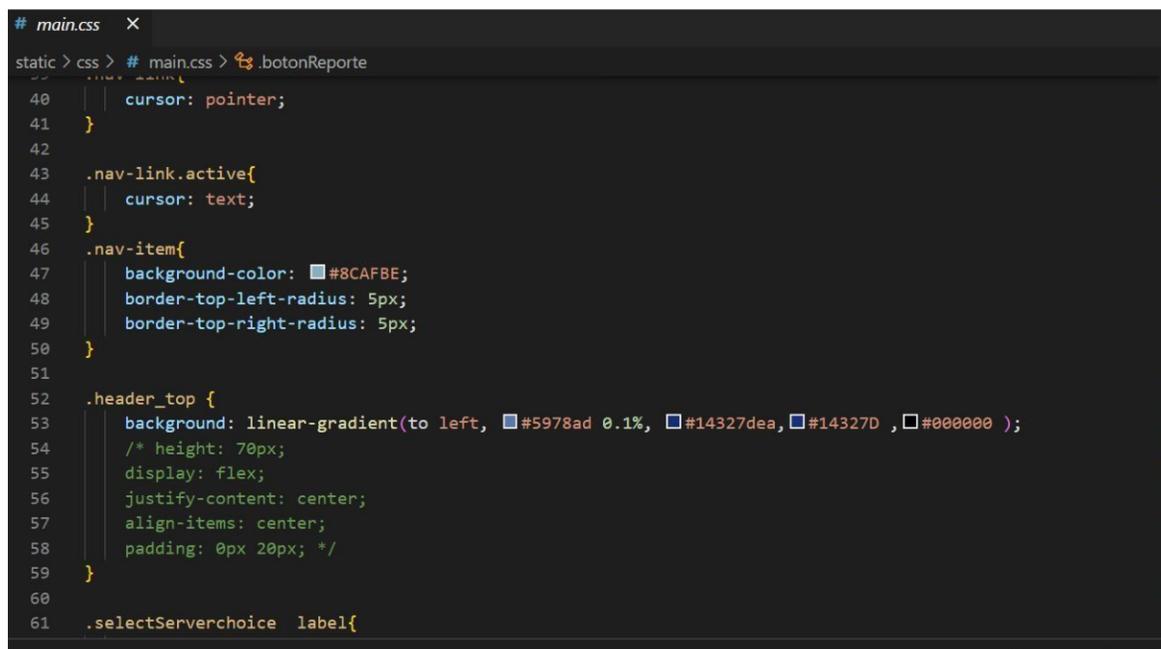
templates > <> base.html
1  {% load static %}
2  <!DOCTYPE html>
3  <html lang="en">
4
5  <head>
6      <meta charset="UTF-8">
7      <meta http-equiv="X-UA-Compatible" content="IE=edge">
8      <meta name="viewport" content="width=device-width, initial-scale=1.0">
9      <link rel="stylesheet" href="{% static 'css/main.css' %}">
10     {% block css %}{% endblock css %}
11     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"
12         integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC" crossorigin="a
13         rel="stylesheet">
14     <link href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.10.3/font/bootstrap-icons
15     </link>
16
17
18
19 </head>
20 <body>
21
22 <!--
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
```

Al tener las vistas creadas de HTML se inicia el siguiente objetivo para dar estilos.

### 3.6.3 Desarrollo estilos css y traemos las imágenes

Una vez creadas las vistas, se crean los estilos para que la aplicación sea visualmente atractiva

**Figura 37** Vista css

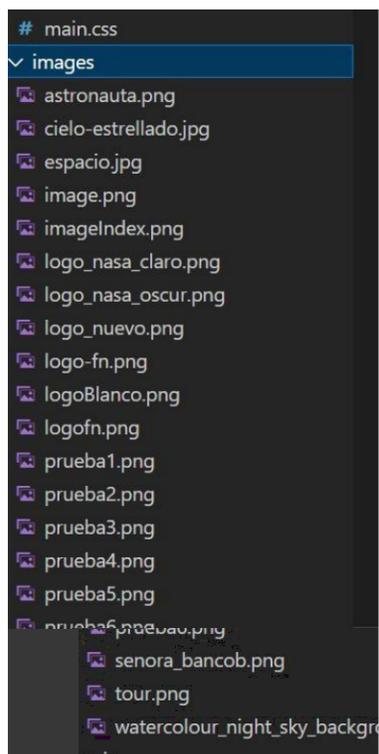
A screenshot of a code editor window titled "# main.css". The editor shows CSS code for styling a web application. The code includes rules for a button, navigation links, navigation items, a header, and a server choice label. The code is as follows:

```
# main.css
static > css > # main.css > botonReporte
40 |   cursor: pointer;
41 | }
42 |
43 | .nav-link.active{
44 |   cursor: text;
45 | }
46 | .nav-item{
47 |   background-color: #8CAFBE;
48 |   border-top-left-radius: 5px;
49 |   border-top-right-radius: 5px;
50 | }
51 |
52 | .header_top {
53 |   background: linear-gradient(to left, #5978ad 0.1%, #14327dea, #14327D, #000000 );
54 |   /* height: 70px;
55 |   display: flex;
56 |   justify-content: center;
57 |   align-items: center;
58 |   padding: 0px 20px; */
59 | }
60 |
61 | .selectServerchoice label{
```

Nota. Creación propia (2023)

Luego de tener css creado y los HTML, incorporamos las imágenes para darle vida a la

Página

**Figura 38** IMGS

Nota. Creación propia (2023)

Creado el estilo de la herramienta se continua con el siguiente objetivo dando funcionalidad

### ***3.6.4 desarrollo js para traer información de sitescope***

El api de sitescope nos permite traer información de los monitoreos que existen actualmente, se usara esta información para poder mostrar al usuario los monitores de los cuales puede generar reporte y de los cuales puede habilitar y deshabilitar, también se usara este api para automatizar los procesos ya entendiendone donde están las rutas se envían estas rutas para poder automatizar

Figura 39 JS

```

JS index.js X
static > js > JS index.js > handleServerDisableEnable > $.each() callback > buttonMonitor
27
28
29 async function handleMonitors(event) {
30   console.log(event.target.value);
31   const response = await fetch(`${url_base}/get_url_server_report?servidor=${event.target.value}`);
32   const jsonData = await response.json();
33   console.log(jsonData)
34
35   var $el = $("#containerMonitorReport");
36
37   $el.empty(); // remove old options
38   $el.append("<h4 class='mb-4'>Tipo de servidores</h4>");
39   $.each(jsonData["servidores"], function(index, value) {
40     // $el.append("<option></option>")
41     // .attr("value", value).text(value));
42     console.log(value)
43     let elemento = $("<div></div>").addClass("row mb-3")
44     let titulo = $("<span></span>").text(value.celula + " " +value.serviceType + " " + value.adiciona
45     elemento.append(titulo)
46
47     let buttonMonitor = $("<button></button>").attr("value", value.valuePath).attr("name", "botonmonit
48     elemento.append(buttonMonitor)

```

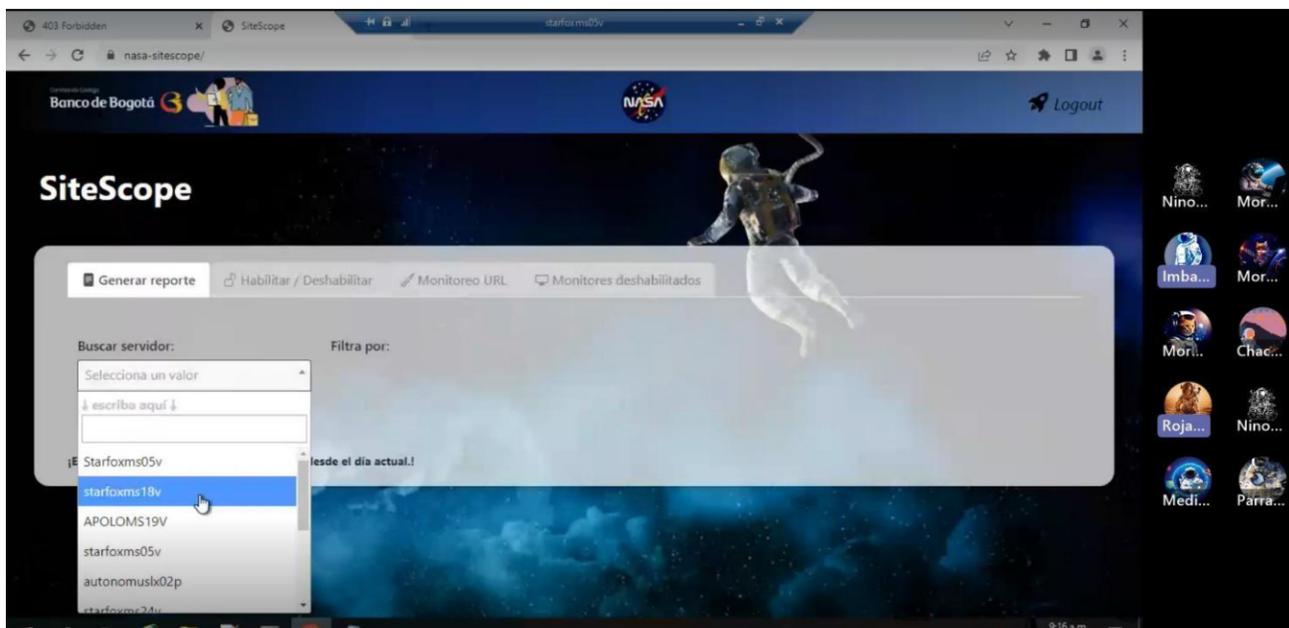
Nota. Creación propia (2023)

Ya con la primera mitad del software completado se da pie al siguiente objetivo

### ***3.6.5 hacemos primera entrega***

Se hace entrega de un adelanto del desarrollo de la aplicación sin el funcionamiento de Python solo la vista y la funcionalidad que se tiene actualmente

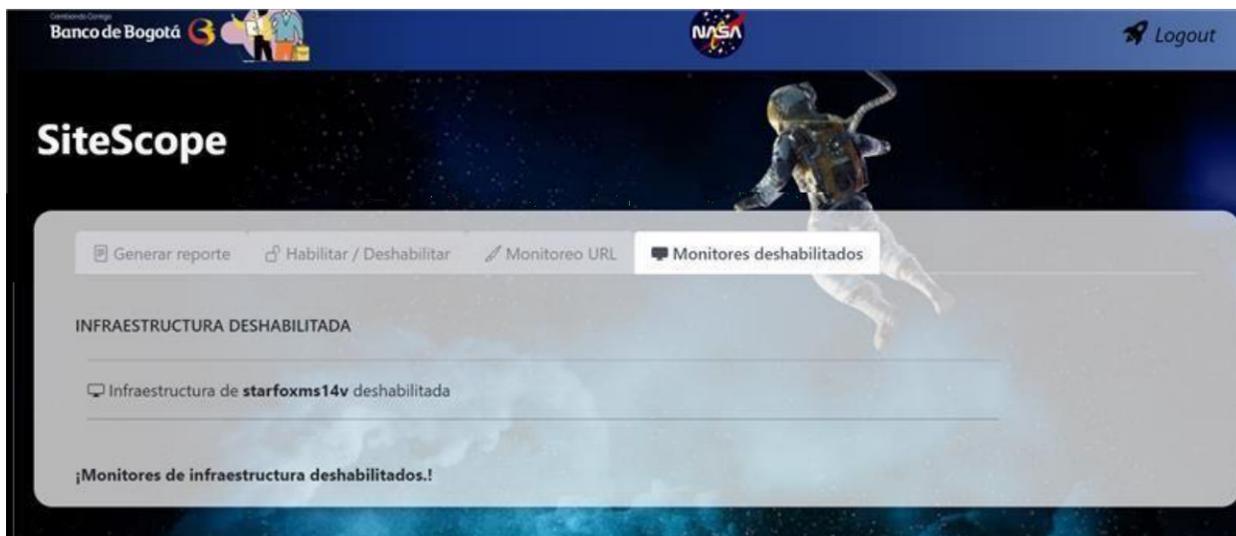
**Figura 40** Primera entrega herramienta



Nota. Creación propia (2023)

En esta primera entrega se pueden buscar los servidores para generar un reporte, se pueden buscar un servidor para habilitarlo y deshabilitarlo, se diseñó la página para crear un monitoreo de url y se diseñó la página que muestran los servidores que están deshabilitados.

**Figura 41** vista herramienta



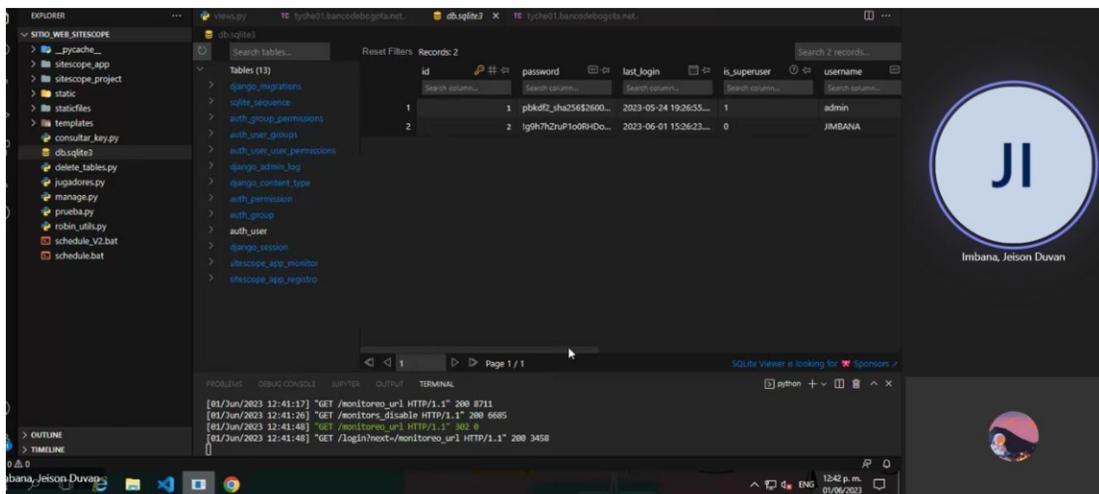
Nota. Creación propia (2023)

Luego de hacer la primera entrega y esta ser aprobada, se comienza a realizar el siguiente objetivo

### ***3.6.6 creación de login contra directorio activo del banco:***

antes de comenzar con la funcionalidad principal de la aplicación se debe crear un login que este autenticado contra el directorio activo del banco para que solo los usuarios del banco tengan acceso a la aplicación, Django crea una base de datos propia para la función de logeo.

Figura 42 creación del login

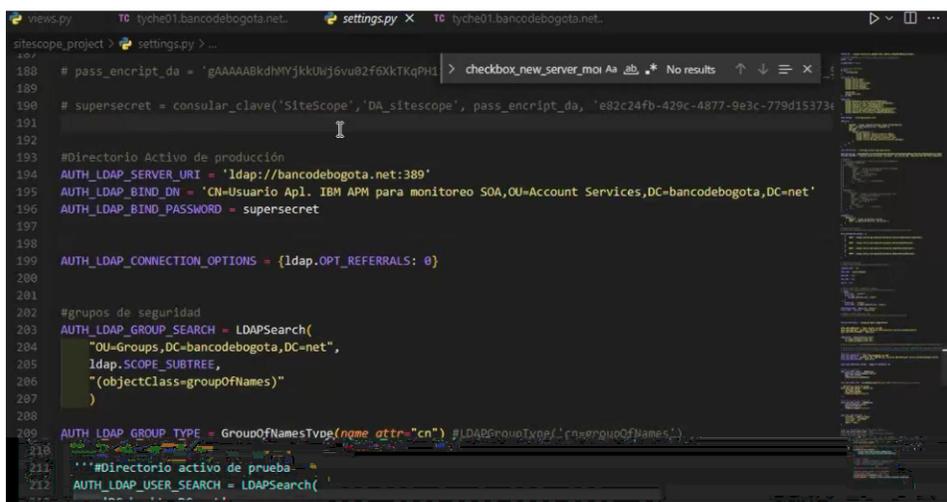


Nota. Creación propia (2023)

Se debe cambiar esa dirección de la base de datos actual por la del banco, en settings.py.

Aquí le damos la dirección el usuario y la contraseña de la base de datos para acceder a los usuarios que ya existen en el banco.

Figura 43 cambio de base de datos



Nota. Creación propia (2023)

Ahora se crea una vista en views.py para que obliguemos al usuario a logearse contra directorio activo y que de esta forma nadie externo al banco pueda ingresar a la herramienta y también obligamos al usuario a ingresar con un navegador actualizado. Para que no se genere ningún error a la hora de navegar en la herramienta.

**Figura 44** Vista.PY

```

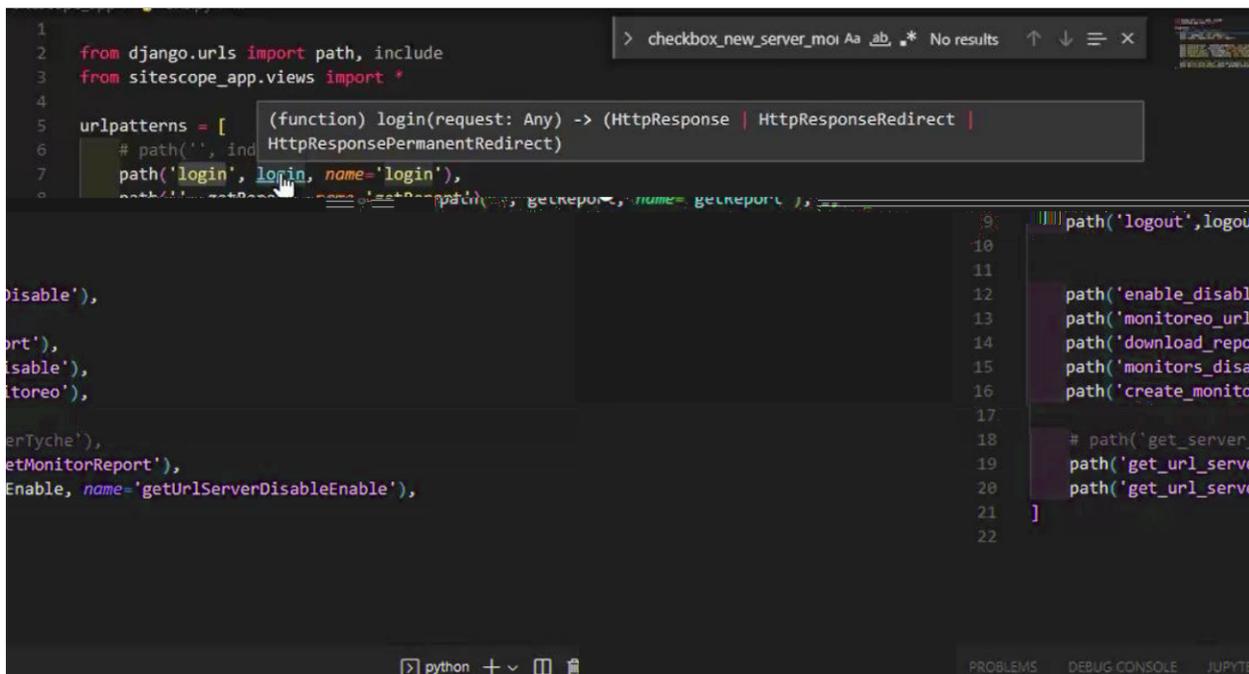
563     'form': CreateMonitorForm(ceLula=ceLula_pertene
564 }
565
566     return render(request, 'sitescope_app/createMonitoreo.html', context)
567
568
569     def login(request):
570         user_agent = request.META.get('HTTP_USER_AGENT', '')
571
572         supported_browsers = {
573             'Chrome': 94,
574             'Firefox': 96,
575             'Edg': 96,
576         }
577
578         browserEnable = False
579         for browser, min_version in supported_browsers.items():
580
581             if browser in user_agent:
582                 browserEnable = True
583                 version_start_index = user_agent.index(browser) + len(browser) + 1
584                 version = int(user_agent[version_start_index:].split('.')[0])
585
586                 if version < min_version:
587                     return render(request, 'update_browser.html')
588             if not browserEnable:
589
590                 if request.method == 'POST':
591                     ipv4_user = request.META.get('REMOTE_ADDR')
592                     usuario = request.POST.get("usuario")
593                     password = request.POST.get("password")
594
595                     # if usuario == 'usr_nasa':
596                     #     clave_usuario_produccion = encriptacion_clave(
597                     #         'D:/juegos/prodemail/ .key',
598                     #         'D:/juegos/prodemail/ .key'
599                     #     )
600                     #     password = clave_usuario_produccion
601                     user = authenticate(request, username=usuario, password=password)
602
603                     if user is not None:
604                         print("Autenticación Exitosa")
605                         auth_login(request, user)
606
607                         #authenticated_user = request.user
608
609                         request.session['usuario'] = usuario
610                         registro = Registro(usuario= usuario, consulta= "Inicio de sesión", ip
611                         registro.save()

```

Nota. Creación propia (2023)

Ahora nos dirigimos a url.py para poner nuestro login como página principal y que siempre nos redirija al login si no estamos logeados

**Figura 45** Redireccionamiento

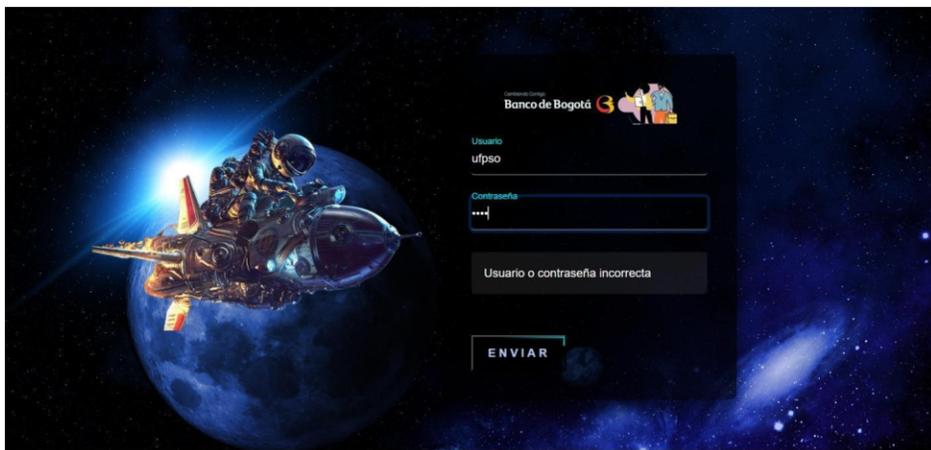


```
1 from django.urls import path, include
2 from sitescope_app.views import *
3
4 urlpatterns = [
5     # path('', include('sitescope_app.urls')),
6     path('login', login, name='login'),
7     path('getreport', getreport, name='getreport'),
8     path('logout', logout, name='logout'),
9     path('enable_disable', enable_disable, name='enable_disable'),
10    path('monitoreo_url', monitoreo_url, name='monitoreo_url'),
11    path('download_report', download_report, name='download_report'),
12    path('monitors_disable', monitors_disable, name='monitors_disable'),
13    path('create_monitor', create_monitor, name='create_monitor'),
14    # path('get_server_url', get_server_url, name='get_server_url'),
15    path('get_url_server', get_url_server, name='get_url_server'),
16    path('get_url_server_disable', get_url_server_disable, name='get_url_server_disable'),
17    ]
18
19
20
21
22
```

The screenshot shows a code editor with a Python file named `checkbox_new_server_moi`. The code defines Django URL patterns and a `login` function. A tooltip is visible over the `login` function definition, showing its signature: `(function) login(request: Any) -> (HttpResponse | HttpResponseRedirect | HttpResponsePermanentRedirect)`. The code includes imports for `path` and `include` from `django.urls` and `login`, `getreport`, `logout`, `enable_disable`, `monitoreo_url`, `download_report`, `monitors_disable`, and `create_monitor` from `sitescope_app.views`. The `urlpatterns` list includes `path('login', login, name='login')`, `path('getreport', getreport, name='getreport')`, `path('logout', logout, name='logout')`, `path('enable_disable', enable_disable, name='enable_disable')`, `path('monitoreo_url', monitoreo_url, name='monitoreo_url')`, `path('download_report', download_report, name='download_report')`, `path('monitors_disable', monitors_disable, name='monitors_disable')`, and `path('create_monitor', create_monitor, name='create_monitor')`. There are also commented-out paths for `get_server_url`, `get_url_server`, and `get_url_server_disable`. The editor interface includes a search bar at the top right, a status bar at the bottom, and a sidebar on the right with tabs for 'PROBLEMS', 'DEBUG CONSOLE', and 'JUPYTER'.

Nota. Creación propia (2023)

Para finalizar le damos estilos css y validamos el funcionamiento

**Figura 46** Login

Nota. Creación propia (2023)

Al crear el login los usuarios ya pueden ingresar con el usuario que tienen del banco, de esta forma se da paso al siguiente objetivo

### ***3.6.7 creación del funcionamiento traer reportes:***

Importamos urllib para poder acceder a la url de sitescope de forma segura ya que al solicitar un reporte por medio de la api lo que se devuelve es una url y lo que necesitamos es el informe, hacemos un get cuando los usuarios escojan el servidor al cual se le va a generar el reporte, ya cuando los usuarios escogen el servidor enviamos la información a la api de sitescope y esta nos devuelve una url, por lo tanto por medio de código iniciamos sesión en esta url con el usuario de sitescope, este inicio de sesión nos genera una nueva url con las seguridad y las cookies validadas, aquí se encuentra el reporte de sitescope.

**Figura 47** get traer reportes

```

137
138 def get_cookie_session(tyche):
139     headersList = {
140         "Accept": "*//*",
141         "Content-Type": "application/x-www-form-urlencoded"
142     }
143     base_URL_cookie = base_URL
144     if tyche != "Tyche01":
145         base_URL_cookie = base6_URL
146
147     respuestaLogin = requests.get(base_URL_cookie, headers=headersList, verify=False)
148
149     if respuestaLogin.status_code == 200 or respuestaLogin.status_code == 204:
150
151         cookiesLogin = respuestaLogin.cookies
152         CookiesessionLogin = cookiesLogin.get({"PEDROID"})
153     else:
154         raise Exception(f'Error al consultar eventos: {respuestaLogin.status_code}', respuestaLogin.text)
155
156
157     url_session = f'{base_URL_cookie}/j_security_check'
158

```

Nota. Creación propia (2023)

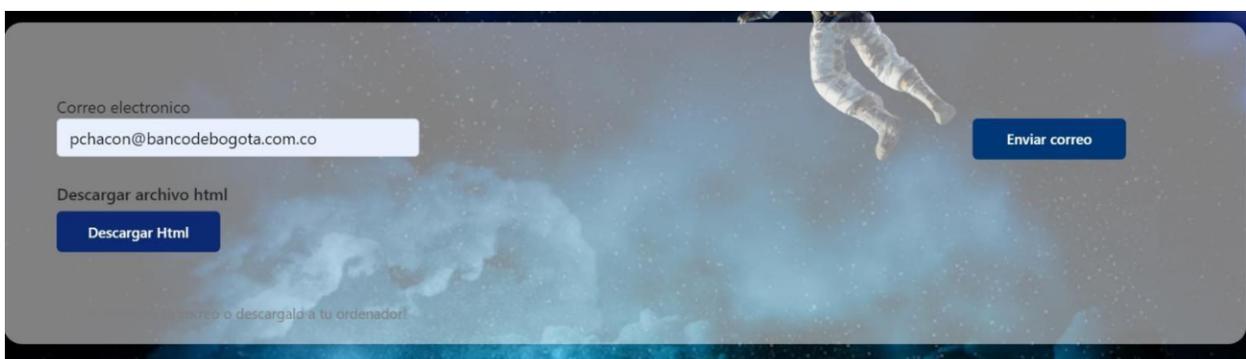
Luego de tener la url definitiva, usamos BeautifulSoup para extraer la información de las imágenes, ya que para poder enviar correctamente el HTML debemos transformar las imágenes en base 64 y de esta forma los usuarios finales podrán ver el reporte completo que se encuentra en esta url, luego de extraer la información obtenemos el reporte final, por lo tanto se crean dos botones, en la vista HTML para que al darle clic se pueda descargar esta información o la enviéal mismo correo del usuario que esta logeado en la aplicación.

**Figura 48** funcionalidad botones

```
reporteInforme = get_report_url_html(fecha_incial_new, fecha_final, report_servidor)

if(buttonEnviarCorreo):
    inputCorreo = request.POST.get('input_correo_report',None)
    print(inputCorreo)
    if inputCorreo:
        enviarMensaje(reporteInforme["html"], inputCorreo)
        return redirect("getReport")
        # enviarMensaje(htmlenviar, inputCorreo)
elif(buttonDescargar):
    archivoEnviar = BytesIO(reporteInforme["html"].encode("utf-8"))

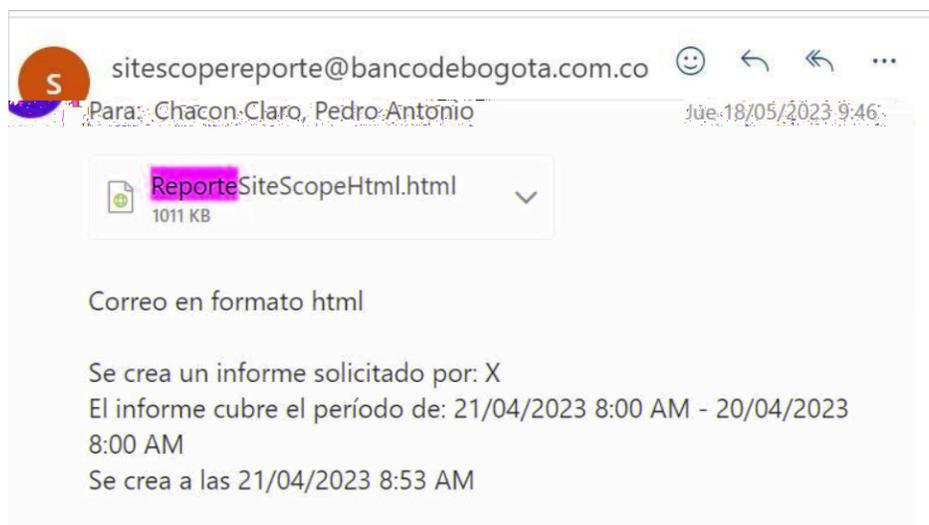
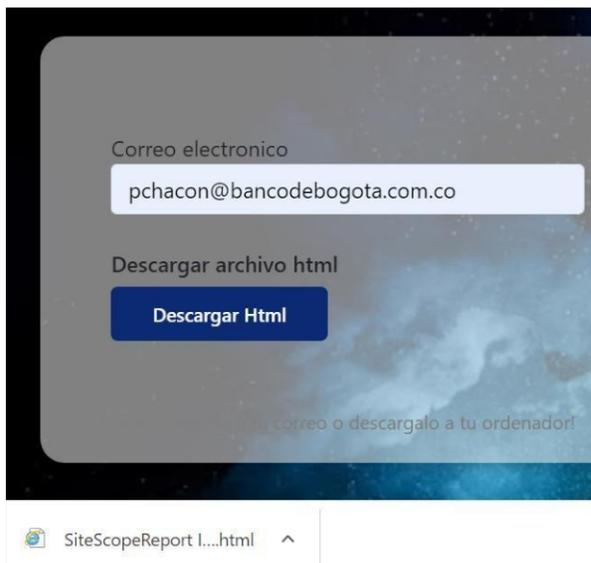
    nombreServidor = report_servidor.split("_sis_path_delimiter_")
    responseArchivo= FileResponse(archivoEnviar,as_attachment=True, filename=f'SiteScopeReport')
    return responseArchivo
```



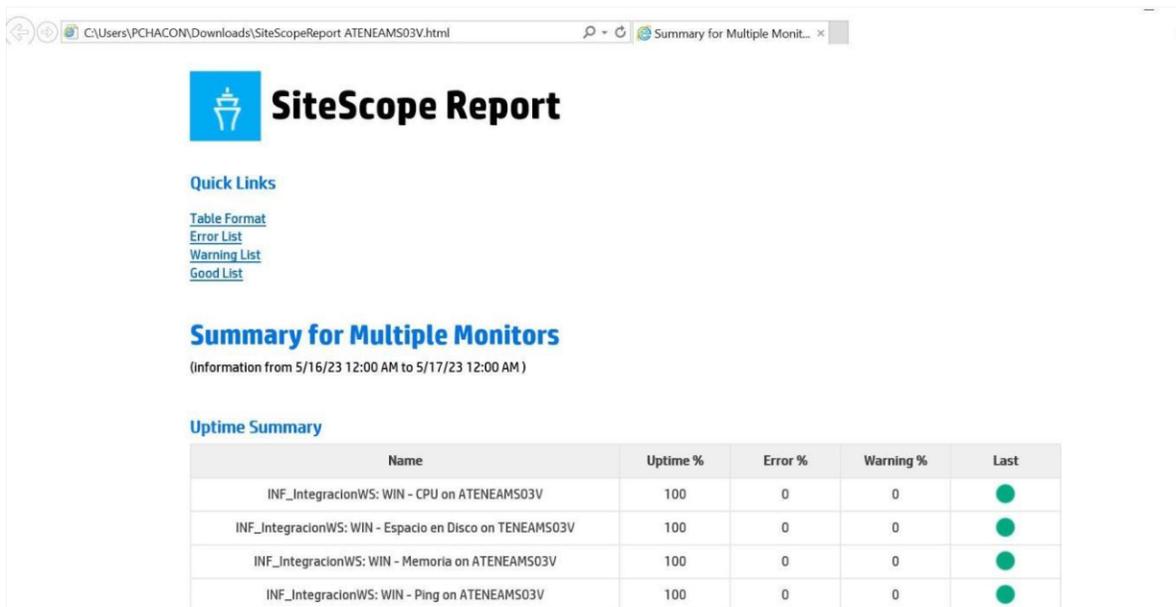
Nota. Creación propia (2023)

Si el usuario da clic sobre descargar HTML se descarga el documento directamente, si le da enviar correo se envía el HTML por correo

**Figura 49** acción clic en botón descargar



**Figura 50** reporte que entrega la herramienta



**SiteScope Report**

**Quick Links**

[Table Format](#)  
[Error List](#)  
[Warning List](#)  
[Good List](#)

**Summary for Multiple Monitors**  
 (information from 5/16/23 12:00 AM to 5/17/23 12:00 AM)

**Uptime Summary**

Name	Uptime %	Error %	Warning %	Last
INF_IntegracionWS: WIN - CPU on ATENEAMS03V	100	0	0	●
INF_IntegracionWS: WIN - Espacio en Disco on TENEAMS03V	100	0	0	●
INF_IntegracionWS: WIN - Memoria on ATENEAMS03V	100	0	0	●
INF_IntegracionWS: WIN - Ping on ATENEAMS03V	100	0	0	●

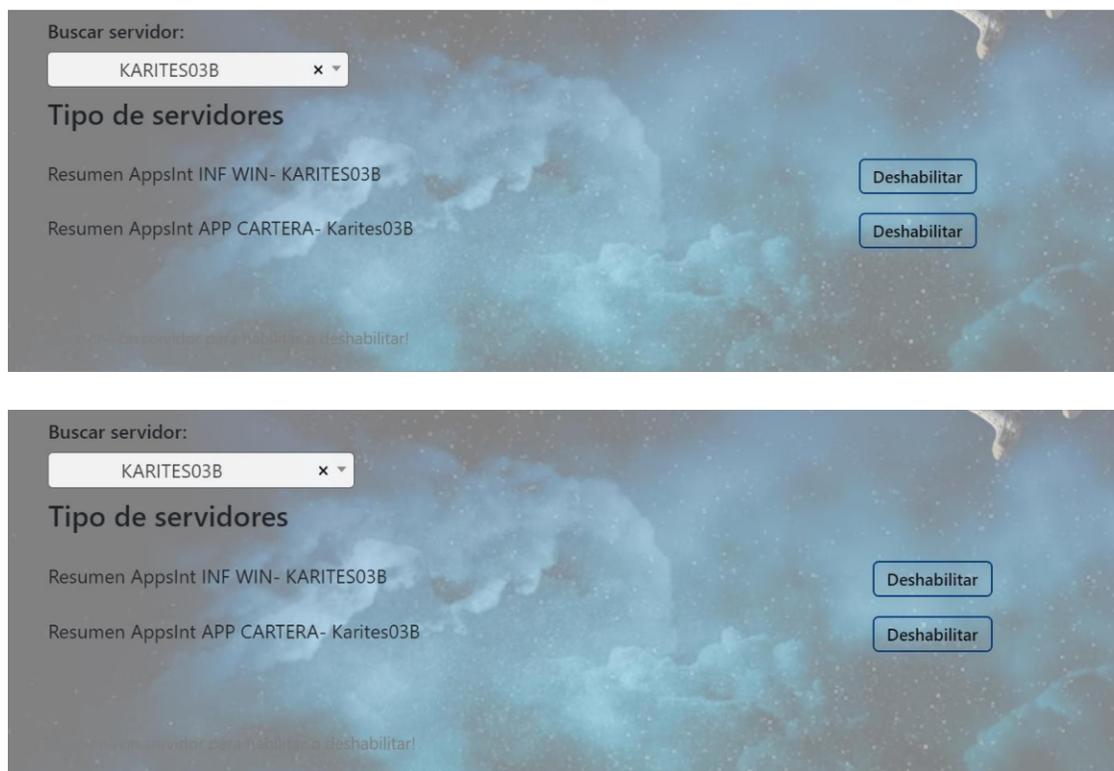
Nota. Creación propia (2023)

Ya creada la funcionalidad de traer reportes el usuario podrá crear reportes de los últimos 30 días de cada maquina monitoreada, por ende, se da comienzo al siguiente objetivo

### ***3.6.8 creación del funcionamiento habilitar y deshabilitar:***

Traemos los servidores y les asociamos un botón, si el servidor se encuentra deshabilitado le asociamos el botón de habilitar, si el botón se encuentra habilitado le asociamos el botón de deshabilitar

**Figura 51** visual habilitar/deshabilitar



Nota. Creación propia (2023)

hacemos un post cuando los usuarios escojan el servidor que previamente enviamos del api, ese post lleva el nombre del monitor y su estado si el usuario habilito el estado es false, si el usuario deshabilita, el estado que se manda es true.

**Figura 52** funcionalidad botón habilitar deshabilitar

```

if(request.method == "POST"):

    botonEnable = request.POST.get('botonEnable',None)
    botonDisable = request.POST.get('botonDisable',None)

    pathGroup = ""
    statusGroup = None

    if botonEnable:
        pathGroup = botonEnable
        statusGroup = True

    if botonDisable:
        pathGroup = botonDisable
        statusGroup = False

    parametrosDisableEnable = {
        "fullPathToGroup":pathGroup, #group/monitor
        "enable":statusGroup,
    }

    response = enable_disalbe_group(parametrosDisableEnable, pathGro
    print(response)

```

Nota. Creación propia (2023)

Ya creada la funcionalidad de deshabilitar/habilitar el usuario podrá habilitar y deshabilitar cada maquina monitoreada, por ende, se avanza con el siguiente objetivo.

### ***3.6.9 creación del funcionamiento crear monitoreo de url:***

Ahora para crear un monitoreo de url diseñamos una página HTML donde el usuario ingrese el servidor la url y la carpeta donde se guarda el monitoreo

**Figura 53** visual crear monitoreo

1. Verificar conexión del servidor

Buscar servidor:  
  
 Agrega un nuevo servidor. [Crear servidor](#)

2. Crear monitoreo de la url

Url:       Aplicación.:

\*\* Agrega la URL donde está alojado el sitio web. Es el nodo, no el de balanceo.

Nota. Creación propia (2023)

Cuando ingrese los valores solicitados y haga clic en crear monitoreo enviamos un post al api de SiteScope para crear el Monitoreo automáticamente por medio de una plantilla diseñada por el grupo nasa.

**Figura 54** funcionalidad api crear monitoreo

```
def create_server_template(data):
    url = f'{base_URL}/api/templates/templateDeployment'
    nombre_celula = data['Nombre_Celula']
    typo_servidor = data['Type_Server']
    path_to_template = 'Automatizar_sis_path_delimiter_Plantillas_sis_path_delimiter_Servidor_Win_sis_path'
    path_target_group = f'BANCO DE BOGOTA_sis_path_delimiter_{nombre_celula}_sis_path_delimiter_INF_sis_pa
```



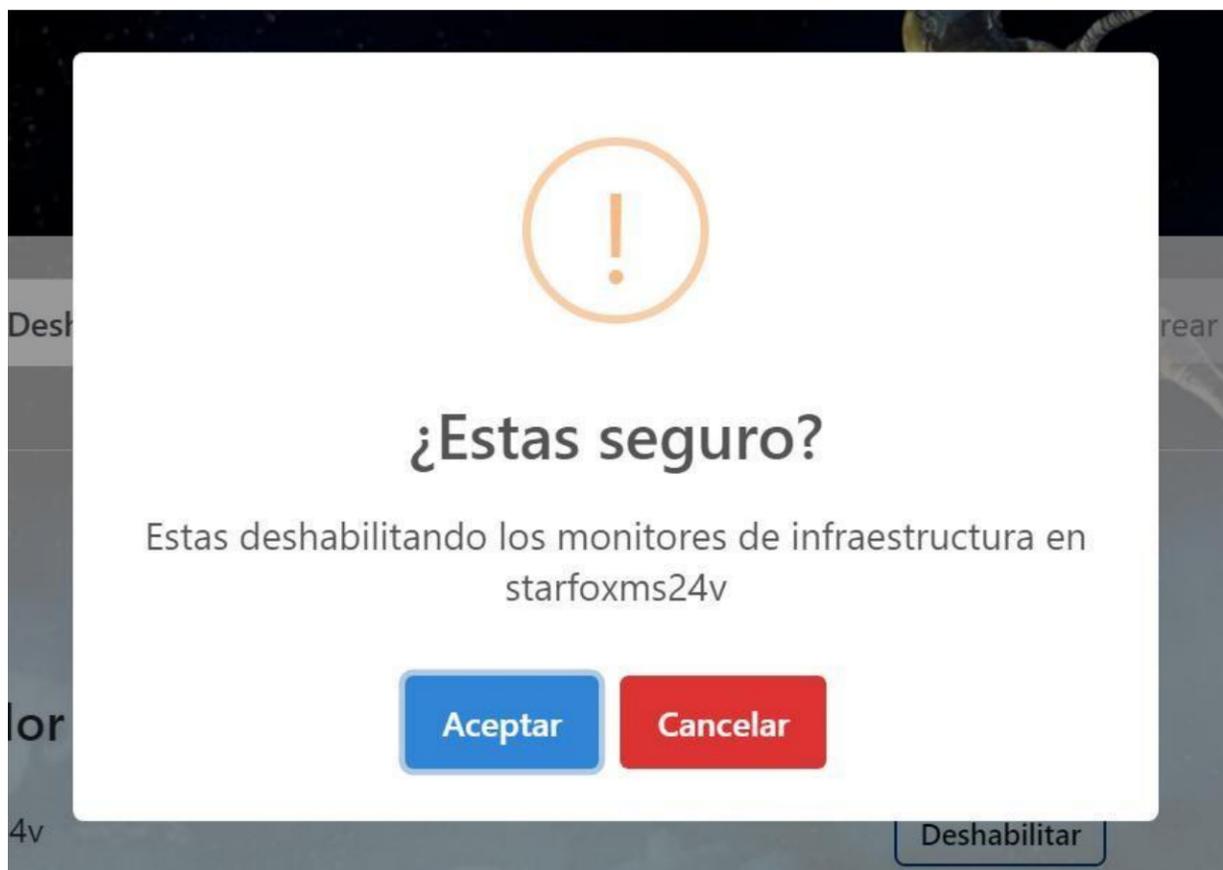
Nota. Creación propia (2023)

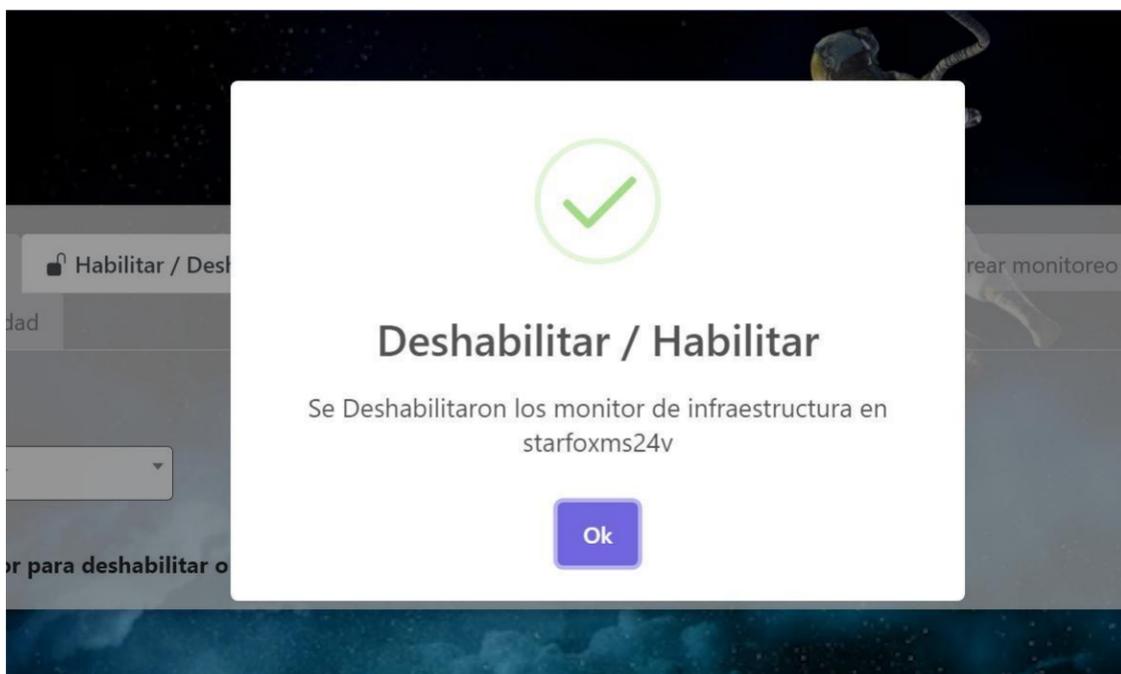
Ya creada la funcionalidad de crear monitoreos el usuario podrá crear monitoreos de url de cada maquina monitoreada, terminado esto se da comienzo al siguiente objetivo.

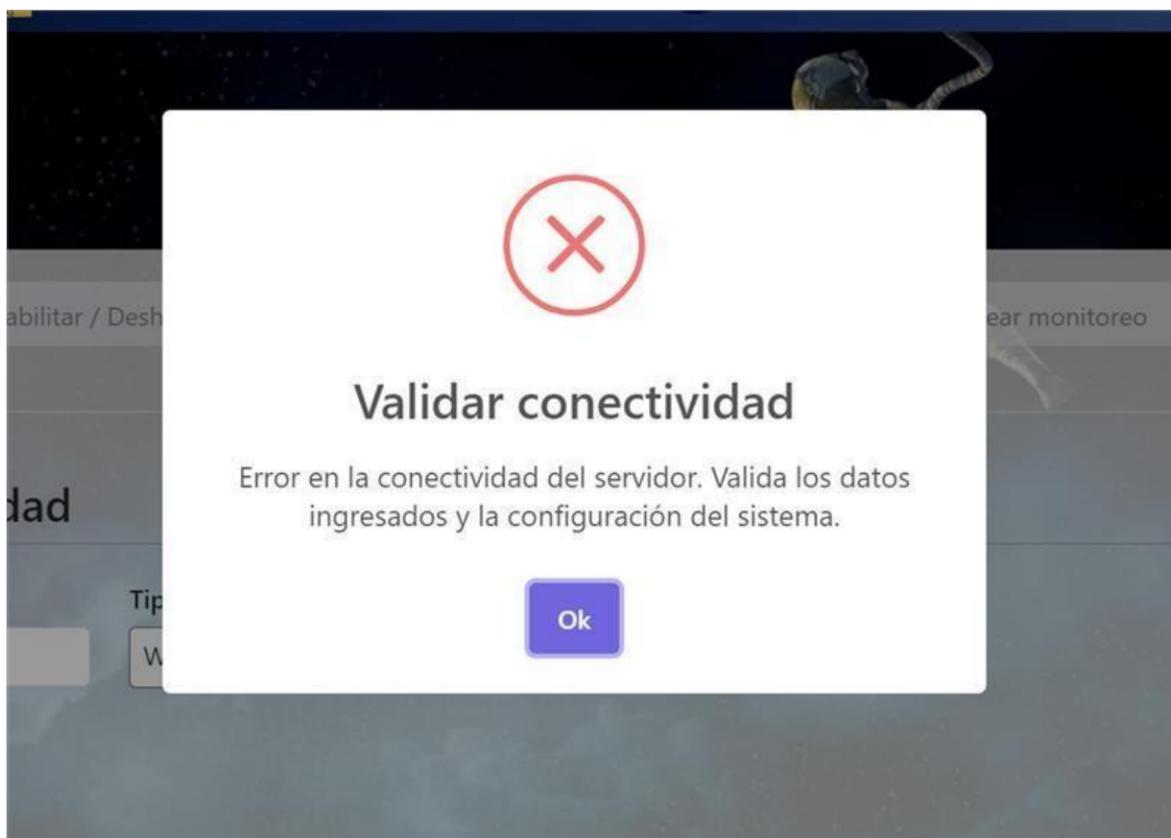
### ***3.6.10 creación de alertas:***

Se crean alertas que le notifiquen a los usuarios si los procesos que están realizando son satisfactorios o si ocurrió algún error.

**Figura 55** Alertas







Nota. Creación propia (2023)

Al crear alertas el usuario tendrá una mejor experiencia de navegación y entenderá de mejor manera que hace cada botón y cómo funciona la herramienta, con este último objetivo finalizamos podemos realizar el objetivo final de esta herramienta

### ***3.6.11 Hacemos entrega final de la primera herramienta:***

Entregamos la herramienta de automatización de proceso de sitescope y explicamos el funcionamiento completo, esta herramienta la subimos a un servidor apache con los certificados que nos entregó el banco.

Figura 56 subiendo herramienta a sitescope

```
<VirtualHost nasa-sitescope:443>

# General setup for the virtual host
DocumentRoot "${SRVROOT}/htdocs"
ServerName nasa-sitescope:443
ServerAdmin hrojas2@bancodebogota.com.co
ErrorLog "${SRVROOT}/logs/error.log"
TransferLog "${SRVROOT}/logs/access.log"

Alias /static "D:/NASA/sitio_web_sitescope/static"
<Directory "D:/NASA/sitio_web_sitescope/static">
    Require all granted
</Directory>

WSGIScriptAlias / "D:/NASA/sitio_web_sitescope/sitescope_project/wsgi.py"

<Directory "D:/NASA/sitio_web_sitescope/sitescope_project">
    Require all granted
    <Files wsgi.py>
        Require all granted
    </Files>
</Directory>
```

view

D (D:) > Apl > Apache24 >

Name	Date modified	Type	Size
bin	18/05/2023 10:19 a...	File folder	
cgi-bin	18/05/2023 10:19 a...	File folder	
conf	24/05/2023 5:58 p...	File folder	
error	18/05/2023 10:19 a...	File folder	
htdocs	18/05/2023 10:19 a...	File folder	
icons	18/05/2023 10:19 a...	File folder	
include	18/05/2023 10:19 a...	File folder	
lib	18/05/2023 10:19 a...	File folder	
logs	30/05/2023 11:35 a...	File folder	
manual	18/05/2023 10:19 a...	File folder	
modules	18/05/2023 10:19 a...	File folder	
ABOUT_APACHE.txt	21/02/2020 1:33 a...	Text Document	14 KB
CHANGES.txt	13/04/2023 9:26 a...	Text Document	1 KB
INSTALL.txt	13/04/2023 9:26 a...	Text Document	1 KB
LICENSE.txt	13/04/2023 9:26 a...	Text Document	44 KB
NOTICE.txt	13/04/2023 9:26 a...	Text Document	3 KB
README.txt	16/02/2023 8:25 p...	Text Document	5 KB

Nota. Creación propia (2023)

Al hacer entrega de la herramienta, se da comienzo al desarrollo de la siguiente

herramienta

### 3.7 Desarrollar la herramienta de que permita la automatización de smax

Para desarrollar esta herramienta se crearán 3 funcionalidades principales, un Bot que solicite id del caso de smax, una IA que entienda los casos y pueda generar textos para los cierres y un script que permita a un Bot logearse en smax y buscar el id de cada caso

#### 3.7.1 Desarrollo de chatbot

Se desarrolla un chat Bot que solicita a los usuarios ingresar el número de id de smax para poder buscarlo, primero diseñamos un bucle infinito para que el Bot siempre este contestando las preguntas que le realizamos, luego se crea una función que nos permita limpiar el mensaje del usuario quitando los signos de puntuación y dejando todo el texto que ingrese el usuario en minúsculas.

**Figura 57**limpia de mensajes

```
import re
import random

def get_respuesta(user_input):
    split_message = re.split(r'\s|[,;.:?!-_\s*]', user_input.lower())
    respuesta = check_all_messages(split_message)
    return respuesta
```

Nota. Creación propia (2023)

Luego se crea una función que nos permita calcular las respuestas del Bot, según el

mensaje enviado por el usuario, por ende se crea un mensaje por defecto que permite la primera interacción con el usuario y le indica lo que debe enviar el id del smax para cerrar el caso

**Figura 58** respuesta predeterminada

```
        return unknown() if highest_prob[best_mat] < 0.5 else response

def unknown():
    response = ['Hola soy tu asistente digital, e
    return response

while True:
```

Nota. Creación propia (2023)

Si el usuario contesta correctamente este id se envía con un script de Python para cerrar un caso de smax y nos devuelve una respuesta

**Figura 59** funcionamiento Bot

```
You: hola
Bot: Hola soy tu asistente digital, estaré disponible las 24 hora
porfavor ingrese 1 si quiere cerrar un caso en smax
You: 1
Bot: Ingrese el id Del caso a cerrar
You: 17793219
Bot: se esta cerrando el smax porfavor espere la respuesta
Bot: se cerro el caso correctamente
```

Nota. Creación propia (2023)

Al finalizar este objetivo el usuario puede comenzar a realizar solicitudes por medio de WhatsApp, por ende, se da paso a la creación de la funcionalidad que son los siguientes objetivos

### ***3.7.2 Entrenamiento de IA***

Ahora comienza el desarrollo de una IA que pueda comprender texto y entender si este texto corresponde a un caso de deshabilitar, crear, habilitar, o generar y según la respuesta de la IA megenere otro texto, para esto tenemos que entrenar la IA con un documento csv que manejaremos con padas

Organizamos el archivo csv para que la IA asocie el contenido de cada smax a un label que indicara si el que tipo de solicitud es, deshabilitar, crear, habilitar, o generar y

limpiamos el contenido de cada smax para que sea más fácil para la IA poder procesarlos

**Figura 60** label y limpieza de datos

```

datos analizar

columnas = ["entrenarsmax", "id", "contenido", "fecha", "label"]
datos = pd.read_csv ("datosx.csv", header= none, names=columnas, engine="python", encoding="latin1")

metadatos = datos

metadatos.drop(["id","fecha"], axis = 1, inplace = true)

nombres_salida = metadatos.features['label'].names

['generar', 'habilitar', 'deshabilitar', 'crear']

```

```

def clean_datos(contenido):
    contenido = BeautifulSoup(contenido, "xml").get_text()
    contenido = re.sub(r"[A-Za-z0-9]+", " ", contenido)
    contenido = re.sub(r"[a-zA-Z? ]+", " ", contenido)
    contenido = re.sub(r" +", " ", contenido)

    return contenido

datos_limpios =[clean_datos(contenido) for contenido in metadat

```

Nota. Creación propia (2023)

Ahora se hace una tokenización para transformar nuestras oraciones en números asignándole unalongitud de máximo 100 palabras así generamos una red neuronal que tendrá 100 entradas y 4 salidas.

Si la oración es menor a 100 palabras las palabras faltantes se llenan con ceros



**Figura 62** clase conjunto de datos

```
class ConjuntoDatos(Dataset):  
  
    def __init__(self, contenido, labels, token, max_len):  
        self.contenido = contenido  
        self.labels = labels  
        self.token = token  
        self.max_len = max_len  
  
    def __len__(self):  
        return len(self.contenido)  
  
    def __getitem__(self, item):  
        contenido = str(self.contenido[item])  
        label = self.labels[item]  
        encoding = token.encode_plus(  
            contenido,  
            max_length = self.max_len,  
            truncation = True,  
            add_special_tokens = True,  
            return_token_type_ids = False,  
            pad_to_max_length = True,  
            return_attention_mask = True,  
            return_tensors = 'pt'
```

Nota. Creación propia (2023)

Luego se creó una nueva función que llame la clase conjunto Datos y que nos retorne un conjunto de datos con un batch\_size de 64 y un num worker de 4 que nos permite analizar los datos en paralelo.

**Figura 63** funcionalidad datos cargados

```
def datosCargados(Metadatos token, max_len, batch_size):
    dataset = ConjuntoDatos(
        contenido = metadatos.contenido.to_numpy(),
        labels = metadatos.label.to_numpy(),
        token = token,
        max_len = 100
    )

    return datosCargado(dataset, batch_size = 64, num_workers = 4)
```

Nota. Creación propia (2023)

Ahora dividimos nuestro set de datos para que nuestros datos sirvan para entrenar y testear la IA

**Figura 64** división de los datos prueba / entrenamiento

```
np.random.seed(ALEATORIO)

metadatos_entrenar, metadatos_prueba = train_test_split(metadatos, test_size = 0.1, random_state=ALEATORIO)

datos_entrenamiento_loader = data_loader(Metadatos_entrenar, token, MAX_LEN, BATCH_SIZE)
datos_prueba_loader = data_loader(Metadatos_prueba, token, MAX_LEN, BATCH_SIZE)
```

Nota. Creación propia (2023)

Ahora escribimos el código del modelo y para ponerlo a entrenar, para esto inicializamos una clase que tiene todas las características de una red neuronal, donde se encuentra un modelo reentrenado de una red neuronal e incorporamos una capa lineal de neuronas que tienen 768 neuronas y tienen como salida 4 neuronas que son las respuestas que tiene que predecir la IA, Luego se desarrolla otra función, una capa adicional de Dropout, para poder hacer más robusta la IA, de esta forma los datos de entrenamiento y de prueba generen una precisión similar.

Ahora se crea una función para definir como están conectadas las capas en esta función enviamos los datos de entrada, lo que nos devuelve toda la esencia de la frase y este es el vector que debemos llevar a la red neuronal.

**Figura 65** creación red neuronal

```

NumeroCtagori= 4

class ClasificacionSmx(nn.Module):

    def __init__(self, NumeroCtagori):
        super(ClasificacionSmx, self).__init__()
        self.bert = BertModel.from_pretrained(PRE_TRAINED_MODEL_NAME)
        self.drop = nn.Dropout(p=0.3)
        self.linear = nn.Linear(self.bert.config.hidden_size, NumeroCtagori)

    def forward(self, input_ids, attention_mask):
        _, cls_output = self.bert(
            input_ids = input_ids,
            attention_mask = attention_mask
        )
        drop_output = self.drop(cls_output)
        output = self.linear(drop_output)
        return output

model = BERTSentimentClassifier(NumeroCtagori)
model = model.to(device)

```

Nota. Creación propia (2023)

Ahora se crea el entrenamiento, este contiene 600 iteraciones de entrenamiento, el optimizador que en este caso usaremos Adam que contendrá los parámetros del modelo, la tasa de aprendizaje, por otra parte asignamos el número total iteraciones de entrenamiento que es el número de datos que tengo en batch que son 64 por el número de iteraciones que son 600, por otra parte se desarrolla un parámetro para optimizar el entrenamiento y la función de error que quiero minimizar, por último se crea la función que entrene nuestra inteligencia artificial e imprimimos este entrenamiento

**Figura 66** creación iteraciones

```
EPOCHS = 600
optimizer = AdamW(model.parameters(), lr=2e-5, correct_bias=
total_steps = len(datos_entrenamiento_loader) * EPOCHS
scheduler = get_linear_schedule_with_warmup(
    optimizer,
    num_warmup_steps = 0,
    num_training_steps = total_steps
)
loss_fn = nn.CrossEntropyLoss().to(device)

def train_model(model, datosCargados, loss_fn, optimizer, de
    model = model.train()
    losses = []
    correct_predictions = 0
    for batch in data_loader:
```

```
Epoch 1 de 600
-----
Entrenamiento: Loss: 0.2167752470637386, accuracy: 0.902567
Validación: Loss: 0.26672457740598, accuracy: 0.895673
```

Nota. Creación propia (2023)

Comprobamos que se estén haciendo las predicciones bien asignando un texto propio y viendo el resultado,

**Figura 67** prueba funcional IA

```
texto_prueba = se necesita crea unos nuevos monitoreos para los servidores apolo23v y apolo24v
classificadorsmx(texto_prueba)
necesita crea unos nuevos monitoreos para los servidores apolo23v y apolo24v Prediccion: crear
```

Nota. Creación propia (2023)

Ya comprobando que la IA hace las predicciones correctas entregamos el modelo para su implementación y comenzamos el ultimo objetivo de esta herramienta.

### ***3.7.3 Navegación a smax***

Para navegar a smax y poder cerrar los casos usamos selenium y accedemos por medio de Google Chrome, primero se desarrolla una clase con la instancia de Google Chrome

Figura 68 Instancia Chrome

```

def instanciaChrome():
    respuesta = False
    try:
        opts = Options()
        opts.add_argument('--ignore-ssl-errors=yes')
        opts.add_argument('--ignore-certificate-errors')
        opts.add_argument('--log-level=3')
        opts.add_experimental_option('detach', True)
        # opts.add_argument('--headless')
        opts.add_experimental_option('excludeSwitches', ['enable-

        opts.add_argument('--user-data-dir=C:\\Users\\KARGUEL\\App
        # opts.add_argument('--user-data-dir=C:\\Users\\HROJAS2\\J
        #opts.add_argument('--user-data-dir=C:\\Users\\JMART91\\
        browser = Chrome(executable_path=r'D:\NASA\cierre-smax=c
    except:
        print ("[ FAIL ] NO SE PUDO CREAR INSTANCIA DE UN NAVEGA
        return respuesta,0
    else:
        print ("[ OK ] CREA INSTANCIA DE NAVEGADOR")
        respuesta = True
        return respuesta, browser

```

Nota: Creación propia (2023)

Luego hacemos que el Bot se dirija a la url de la página de smax

Figura 69 abrir smax

```

def abrePagina(chrome):
    respuesta = False
    inicio = tim()
    urlSmax = "https://smax.bancodebogota.net/dashboard?TENANTID=6
    try:
        chrome.get(urlSmax)
        # time.sleep(2.5)
        WebDriverWait(chrome, 25).until(EC.element_to_be_clickable
    except:
        print ("[ FAIL ] NO SE PUDO ABRIR LA PAGINA WEB SMAX")
        chrome.get_screenshot_as_file("AbrePagina.png")
        fin = tim()
        total = fin - inicio
        return respuesta, total
    else:
        print ("[ OK ] ABRE PAGINA Smax")
        fin = tim()
        total = fin - inicio
        total1 = str(total)[:4].replace(".",",")
        print("Tiempo apertura pagina: %.2f segundos" % (total))
        respuesta = True
        return respuesta, total

```

Nota. Creación propia (2023)

Luego el Bot debe logearse en la página con un usuario y contraseña establecidos

Figura 70 login smax

```

def login(chrome):
    respuesta = False
    inicio = tim()
    try:
        WebDriverWait(chrome, 25).until(EC.element_to_be_clickable((By.XPA
        .send_keys('jvelez1')
        WebDriverWait(chrome, 25).until(EC.element_to_be_clickable((By.XPA
        .click()
        WebDriverWait(chrome, 25).until(EC.element_to_be_clickable((By.XPA
        .send_keys('encriptada')
        WebDriverWait(chrome, 25).until(EC.element_to_be_clickable((By.XPA
        .click()

    except Exception as e:
        print ("[ FAIL ] NO SE PUDO LOGEAR EN SMAX", e)
        # chrome.get_screenshot_as_file("AbrePagina.png")
        fin = tim()
        total = fin - inicio
        return respuesta, total
    else:
        print ("[ OK ] login Smax")
        fin = tim()
        total = fin - inicio
        total1 = str(total)[:4].replace(".",",")
        print("Tiempo apertura pagina: %.2f segundos" % (total))
        respuesta = True
        return respuesta, total

```

Nota. Creación propia (2023)

Por otra parte, tiene que buscar el smax a cerrar con el id que se le mando con el chat Bot

Figura 71 buscar id

```

def filtro(chrome):
    respuesta = False
    inicio = tim()
    try:
        time.sleep(5)
        try:
            WebDriverWait(chrome, 5).until(EC.element_to_be_clickable((By.XPATH,
                '/html/body/div[1]/div/div[2]/saw-grid-container/div/pl-grid-contain
            ).click()
        except:
            print("[ OK ] NO HAY FILTRO")
        time.sleep(5)
        WebDriverWait(chrome, 25).until(EC.element_to_be_clickable((By.XPATH,
            '//*[@id="mainView"]/div/div[2]/saw-grid-container/div/pl-grid-containe
        ).click()
        WebDriverWait(chrome, 25).until(EC.element_to_be_clickable((By.XPATH,
        ).send_keys('ID')
        WebDriverWait(chrome, 25).until(EC.element_to_be_clickable((By.XPATH,
        ).send_keys(Keys.DOWN)
        WebDriverWait(chrome, 25).until(EC.element_to_be_clickable((By.XPATH,
        ).send_keys(Keys.TAB)
        WebDriverWait(chrome, 25).until(EC.element_to_be_clickable((By.XPATH,
            '/html/body/div[1]/div/div[2]/saw-grid-container/div/pl-grid-container/
        ).send_keys('9364527')
        WebDriverWait(chrome, 25).until(EC.element_to_be_clickable((By.XPATH,
            '/html/body/div[1]/div/div[2]/saw-grid-container/div/pl-grid-container/
        ).send_keys(Keys.RETURN)

```

Nota. Creación propia (2023)

Por último, debe entrar al smax entender que tipo de petición es por medio de la IA,

contestar un mensaje dependiendo del tipo de petición y cerrar el caso



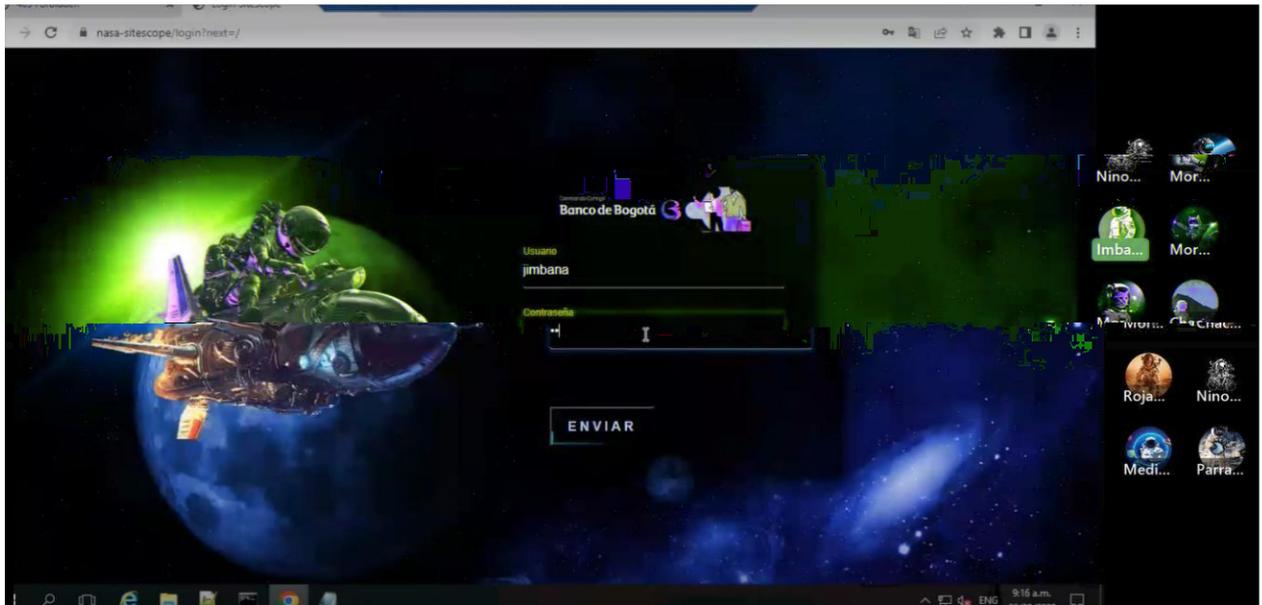
Al mejorar la herramienta se da comienzo al siguiente objetivo

### 3.9 Planificación y diseño de pruebas.

Se planifica y diseñan como van a ser las pruebas de las dos herramientas esta validación tiene que comprobar que todo lo solicitado se cumpla y funcione como es debido esto va a tomar una semana en la cual los usuarios de las herramientas harán pruebas constantes y de todo tipo,

reportando como se generan los errores para poder replicarlos y darles corrección, esta misma se intentara violar la seguridad de las herramientas y ver cómo se comportan

**Figura 73**pruebas



Nota. Creación propia (2023)

Ya planificada cada prueba se empieza el siguiente objetivo.

### ~~3.10 Aplicación de pruebas funcionales.~~

que:

Se aplicaron pruebas funcionales y se comprobó.

go , se comprobó que no se pudiera pasar

Se verifico la Seguridad en la herramienta Django

uenta que este usuario debe estar

del login sin autenticarse primero, teniendo en cu

autenticado contra el directorio activo del banco

gara las predicciones adecuadas tratando

se hicieron pruebas sobre la IA para que no entre

deconfundirla con documentos irónicos

rectamente y guie adecuadamente al

Se hicieron pruebas que el chat Bot funcioné cor

usuario.

reada en Django, se comprobó que él

Se probó las funcionalidades de la herramienta c

ran ningún bug

todas sus pestañas con funcionalidades no genera

cionara correctamente y contestara lo

Se probó que el chat que se logeaba en smax fun

que la IA predecía .

caso en smax funcionara correctamente

Se comprobó que la herramienta para cierres de

cuando se juntaron las 3 funcionalidades .

### **3.11 Corregir errores y entregar las nuevas versiones de las herramientas**

Se hicieron las correcciones de los bugs encontrados y se lanzaron nuevas versiones de lasherramientas

### **3.12 Crear manual de instrucciones de las herramientas**

Se crearon los distintos manuales de uso de las herramientas y como fueron programadas para que el siguiente practicante retome el trabajo realizado

#### 4. Diagnostico final

Una vez finalizada mi practica el banco se crea una auditoria continua del trabajo generado por el grupo “Célula” NASA y ofrece una retroalimentación de los puntos buenos y malos de mi trabajo esto me permitió entender mejor mis habilidades y debilidades como programador y desarrollador.

De igual manera la herramienta desarrollada logro solucionar muchos problemas de tiempo que estaba manejando el banco esto agilizo los procesos de monitoreo del banco y agilizo el tiempo de reacción sobre un problema de monitoreo ya que estas herramientas permitieron que los usuarios tuvieran acceso más rápido a información requiera y de igual manera pudieron enviar y cerrar información de forma más adecuada

Por otra parte, estas prácticas me permitieron adquirir grandes conocimientos del sector bancario y grandes conocimientos de Python, teniendo en cuenta que el banco me capacito en todas las áreas en las que veía una debilidad

La empresa es un escenario el cual me permitió crecer como profesional y como persona agradezco y recomiendo como empleador al banco de Bogotá

## 5. Conclusiones

Para concluir en este proyecto se llevaron a cabo las actividades y objetivos establecidos durante las fases de validaciones. Al enfocarse en la mejora de calidad y eficiencia de los propios sistemas del banco se logró contribuir significativamente al fortalecimiento de la empresa.

Por otra parte, para el desarrollo del proyecto se cumplieron los objetivos específicos, para esto la célula nasa logro entender el funcionamiento de los procesos de monitoreo en el banco para poder diseñar transformaciones tecnológicas en automatización, esto permitió que el desarrollo de la herramienta fuera más eficiente y eficaz, del mismo modo se logró definir un modelo de trabajo eficiente este modelo de trabajo fue aceptado por el banco y permitió evitar retrasos y hacer entrega de todo lo solicitado en los tiempos requeridos.

Por ende, gracias al cumplimiento de los objetivos específicos I y II se logró desarrollar una herramienta que cumplía todas las necesidades especificadas por el banco, lo que permitió que el banco agilizará sus procesos de monitoreo.

Por otro lado, se evaluaron todos los funcionamientos de la herramienta lo que permitió entregar una herramienta sin bugs y con alta escalabilidad, de esta forma la herramienta se implementó en el banco.

Para concluir se realizó una herramienta que permita la automatización de SITESCOP y otra que permita la automatización de SMAX junto a todos los objetivos planteados en el proyecto, lo que permitió mejorar el monitoreo en el banco y obtuvo una gran aceptación por parte de las células (grupos de trabajo), se obtuvo una aceptación tan grande que se espera continuar con la mejora continua de las entregas.

## 6. Recomendaciones

Se recomienda que el proyecto se le dé una mejora continua ya que se puede implementarmuchas más cosas en las dos herramientas.

Se recomienda dar acceso a la herramienta de cierres de caso smax a todas las células y entrenarla IA para que me permita cualquier tipo de respuesta

Por último, Se recomienda mejorar los canales de notificaciones porque muchas veces se solicitan cambios que ya se habían realizado pero que no se tenía constancia de ellos

## Referencias

Abougamea, A. K. E., & Lim, S. C. (2023). Smart power switch using internet of things.

Indonesian Journal of Electrical Engineering and Computer Science, 29(3), 1809.

<https://doi.org/10.11591/ijeecs.v29.i3.pp1809-1816>

Al-Badi, A. H., Khan, A. I., & Eid-Alotaibi. (2022). Perceptions of learners and

instructors towards artificial intelligence in personalized learning. Procedia

Computer Science, 201, 445-451. <https://doi.org/10.1016/j.procs.2022.03.058>

Convergence of cloud with AI for big data analytics. (2023). En Wiley eBooks.

<https://doi.org/10.1002/9781119905233>

Iskandar, K., Widiyanto, B. A., Kuntjoro, M. A., Dwantara, R. A., & Herlina, M. G.

(2023). Vehicle service reservation system and crowd-prediction feature using

ARIMA method. International journal of data and network science, 7(2), 873-882.

<https://doi.org/10.5267/j.ijdns.2023.1.001>

Rave, J. I. P., Correa, J. C., & Echavarría, F. G. (2022). Organizational social

commitment and employee well-being: Illustrating a construct mining approach in R.

Dyna-colombia, 89(223), 27-35. <https://doi.org/10.15446/dyna.v89n223.99230>

Svetlana, N., Norkina, A., Makar, S., Tatiana, G., & Medvedeva, O. E. (2022). Artificial intelligence as a driver of business process transformation. *Procedia Computer Science*, 213, 276-284. <https://doi.org/10.1016/j.procs.2022.11.067>

De Bogotá, B. (s. f.). Un Banco que hace historia <https://www.bancodebogota.com/wps/portal/banco-de-bogota/bogota/nuestra-organizacion/sala-de-prensa/2015/un-banco-que-hace-histori>

Python, R. (2014, September 22). Django Rest Framework – an introduction. *Realpython.com*; Real Python. <https://realpython.com/django-rest-framework-quick-start/>

Ramachandran, N. P., Kurup, R. R., Alphonse, J., Chellaiah, P., Nair, M., Nair, B., & Diwakar, S. (2018). Design and implementation of an open-source browser-based laboratory platform for EEG data analysis. 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI).

Proceedings of the 31st annual ACM SIGUCCS fall conference. (2003). ACM.

Gov.co. Retrieved June 7, 2023, from <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=3431>.

Ley 1341 de 2009 - Gestor Normativo. (n.d.). Gov.co. Retrieved June 7, 2023, from <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=36913>.

Ley 1581 de 2012 - Gestor Normativo. (n.d.). Gov.co. Retrieved June 7, 2023, from <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=49981>.

HDC, S. (2020) Metodología scrum, Una herramienta útil para agilizar tus Proyectos, Honduras Digital Challenge. Available at: <https://hondurasdigitalchallenge.com/2020/05/21/metodologia-scrum-una-herramienta-util-para-agilizar-tus-proyectos/> (Accessed: 31 August 2023).

Redhat.com. Retrieved June 7, 2023, from <https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces>.

SMAX: Solución de gestión de servicios de TI (ITSM). (n.d.). Microfocus.com. Retrieved June 7, 2023, from <https://www.microfocus.com/es-es/products/service-management-automation-suite/overview>.

Welcome to The Apache Software Foundation! (s.f.). Welcome to The Apache Software Foundation! <http://www.apache.org/>