

**IMPLEMENTACIÓN DE REQUISITOS DE SOFTWARE APLICANDO METODOLOGÍAS,
PROCEDIMIENTOS Y TECNOLOGÍAS CORPORATIVAS DE IBM.**

IBM DE COLOMBIA & CÍA S.C.A.

MICHAEL DANIEL ATUESTA CADENA

000199595

SUPERVISORES

JOHANNA MARCELA SUÁREZ PEDRAZA (UPB)

JESÚS ANTONIO GRATEROL (IBM)

FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

UNIVERSIDAD PONTIFICIA BOLIVARIANA SECCIONAL BUCARAMANGA

FLORIDABLANCA

2019

Tabla de contenidos

1	INTRODUCCIÓN	10
2	PRESENTACIÓN DE LA EMPRESA	11
2.1	GENERALIDADES DE LA EMPRESA	11
2.2	DESCRIPCIÓN DE LA EMPRESA IBM	11
2.2.1	PROPÓSITO	11
2.2.2	VALORES	11
2.2.3	PRÁCTICAS	12
3	JUSTIFICACIÓN	13
4	OBJETIVO DE LA PRACTICA	14
4.1	ACTIVIDADES	14
4.2	CRONOGRAMA DE ACTIVIDADES	14
5	DESARROLLO DE LA PRÁCTICA	15
5.1	RECONOCIMIENTOS DE LAS TECNOLOGÍAS Y METODOLOGÍAS DE IBM	15
5.1.1	Capacitaciones	15
5.1.2	Metodologías de desarrollo ágiles	15
5.1.3	Framework de aplicaciones web Angular	17
5.1.4	Enterprise design thinking	18
5.1.5	Personal branding	18
5.1.6	Arquitectura SOA y WebSphere Message Broker	19
5.1.7	Introducción de IBM Cloud	20
5.1.8	Servicio Chatbot de inteligencia artificial Watson	21
5.2	EJERCICIO SIMULADO DE PROPUESTA DE SOLUCIÓN TECNOLÓGICA	21
5.2.1	Desarrollo	21
5.2.2	Análisis del posible problema	21
5.2.3	Determinar posible solución	23
5.2.4	Definir herramienta de IBM para su solución	24
5.2.5	Implementar solución / desarrollo de DEMO	24
5.2.6	Resultados	32

5.2.7	Conclusiones.....	35
5.3	ASIGNACIÓN LABORAL PARA UN CLIENTE REAL	36
5.3.1	HERRAMIENTAS.....	36
5.3.2	Rational ClearQuest.....	61
5.3.3	Soporte Rational ClearQuest	65
5.3.4	Desarrollo con Rational ClearQuest	80
6	CONCLUSIONES	96
7	RECOMENDACIONES	97
8	BIBLIOGRAFIA	98

Índice de ilustraciones

Ilustración 1 Manifiesto ágil.....	16
Ilustración 2 Ejemplo ejercicio combinar objetos.....	18
Ilustración 3 Patron ESB. Tomado de Diapositivas WebSphere Message Broker	19
Ilustración 4 Ejemplo sin Service Bus. Tomado de Diapositivas WebSphere Message Broker	20
Ilustración 5 Ejemplo Service Bus. Tomado de Diapositivas WebSphere Message Broker	20
Ilustración 6 Queja redes sociales banco.....	22
Ilustración 7 Queja 2 redes sociales banco.....	22
Ilustración 8 Horario Chat en Línea Banco.....	23
Ilustración 9 Archivo. env	25
Ilustración 10 Archivo common.js.....	25
Ilustración 11 Archivo conversation.js	26
Ilustración 12 Archivo app.js	27
Ilustración 13 Intent Watson assistant.....	28
Ilustración 14 Ejemplos de entrenamiento de Intent #Agradecimientos	28
Ilustración 15 Estructura nodos de dialogo	29
Ilustración 16 Carpetas de nodos de dialogo	30
Ilustración 17 Contenido JSON del mensaje.....	30
Ilustración 18 Arquitectura basica de la app.....	31
Ilustración 19 Pantalla principal de la app	32
Ilustración 20 solicitud de ayuda en el chatbot.....	32
Ilustración 21 Selección de opciones del chatbot.....	33
Ilustración 22 Respuesta con hipervínculo del chatbot.....	34
Ilustración 23 Pagina destino del hipervínculo, ubicación bancos cercanos.....	34
Ilustración 24 Creación botón de recargar	35
Ilustración 25 Pantalla principal UCD web	36
Ilustración 26 Creación de usuario UCD	37
Ilustración 27 Creación de equipos UCD	37
Ilustración 28 Estado agentes UCD	38
Ilustración 29 Agente asignado a una aplicación	38
Ilustración 30 Aplicación con dos componentes.....	38
Ilustración 31 Versiones de componente	39
Ilustración 32 Procesos de componente	39
Ilustración 33 Proceso básico de editor de procesos	40
Ilustración 34 Plugin UCD	40
Ilustración 35 Pagina plugin UCD "https://developer.ibm.com/urbancode/plugins/ibm-urbancode-deploy/"	41
Ilustración 36 Descripción plugin 7zip	41
Ilustración 37 Continuar si el proceso es exitoso	42

Ilustración 38 Continuar si el proceso es fallido	42
Ilustración 39 Continuar sin importar el resultado	43
Ilustración 40 Proceso de una aplicación	43
Ilustración 41 Ambientes de una aplicación	44
Ilustración 42 Variables de ambiente	44
Ilustración 43 Pantalla inicial solicitud de despliegue	45
Ilustración 44 Definición de versiones de componentes a desplegar	45
Ilustración 45 Pantalla solicitud de despliegue cargada	46
Ilustración 46 Ejecución del despliegue solicitado.....	46
Ilustración 47 Resultado despliegue detallado con cada paso del proceso	47
Ilustración 48 Log detallado de ejecución de paso del proceso.....	47
Ilustración 49 Ambiente con información de los componentes actuales desplegados	47
Ilustración 50 Despliegue versión 1509.....	48
Ilustración 51 Artefactos versión 1509	48
Ilustración 52 Aprobación manual task versión 1509	49
Ilustración 53 Despliegue versión 1520.....	49
Ilustración 54 Artefactos versión 1520	49
Ilustración 55 Aprobación manual task versión 1520	50
Ilustración 56 Output log artefactos despliegue.....	51
Ilustración 57 Proceso de despliegue modificado	52
Ilustración 58 Paso Crear Carpeta	53
Ilustración 59 Script Crear Carpeta	53
Ilustración 60 Paso Download Artifacts.....	54
Ilustración 61 Paso Salida de versión	55
Ilustración 62 Script Salida de versión	55
Ilustración 63 Paso Validacion de versión.....	56
Ilustración 64 Paso Rechazo Borrar Carpeta	56
Ilustración 65 Script Rechazo Borrar Carpeta	57
Ilustración 66 Paso Borrar Carpeta	57
Ilustración 67 Script Borrar Carpeta	58
Ilustración 68 Ruta destino del despliegue.....	58
Ilustración 69 Inicio despliegue versión 1547	58
Ilustración 70 Creación carpeta versión en la ruta	59
Ilustración 71 Inicio despliegue versión 1549.....	59
Ilustración 72 Creación carpeta de dos versiones en la ruta	59
Ilustración 73 Finaliza despliegue versión 1549.....	60
Ilustración 74 Eliminación de carpeta al finalizar despliegue.....	60
Ilustración 75 Finalización despliegue versión 1547	61
Ilustración 76 Eliminación de carpeta versión 1547	61
Ilustración 77 Pantalla de bienvenida ClearQuest.....	62
Ilustración 78 Ejemplo Flujo	62

Ilustración 79 Ejemplo registro	63
Ilustración 80 Ejemplo campos obligatorios	64
Ilustración 81 Ejemplo mensaje de error para campos faltantes	64
Ilustración 82 Diagrama de transición del flujo	65
Ilustración 83 Flujo solicitud de usuarios.....	66
Ilustración 84 Pantalla administrador de usuarios	67
Ilustración 85 Acción agregar usuario	68
Ilustración 86 Ventana de creación de usuarios.....	68
Ilustración 87 Acción actualizar bases de datos.....	69
Ilustración 88 Selección de bases de datos a actualizar	69
Ilustración 89 Lista de IMs asignados	70
Ilustración 90 Consultas de importaciones.....	71
Ilustración 91 Acción de importación.....	71
Ilustración 92 Importación para registros de DatosUsuario	72
Ilustración 93 Importación para flujo de solicitud de usuarios.....	73
Ilustración 94 Mensaje de registros importados	74
Ilustración 95 ClearQuest Designer - carpetas.....	74
Ilustración 96 FlujoSolicitudUsuarios - Fields.....	74
Ilustración 97 Listado de campos para el flujo SolicitudUsuarios	75
Ilustración 98 FlujoSolicitudUsuarios - State Transition Matrix	75
Ilustración 99 Transición de estados con sus respectivas acciones	76
Ilustración 100 FlujoSolicitudUsuarios - Actions.....	76
Ilustración 101 Listado de acciones para el flujo SolicitudUsuarios.....	77
Ilustración 102 FlujoSolicitudUsuarios - Behaviors.....	77
Ilustración 103 Listado comportamientos de campos para cada estado	78
Ilustración 104 FlujoSolicitudUsuarios - Forms	78
Ilustración 105 GUI interfaz gráfica para el flujo SolicitudUsuarios.....	79
Ilustración 106 Cuadro de herramientas para interfaz grafica	79
Ilustración 107 Propiedades de los campos o cuadros de texto	80
Ilustración 108 Registro Desarrollo actualizado	82
Ilustración 109 Pestaña Pruebas actualizada	83
Ilustración 110 Pestaña Implementación actualizada.....	84
Ilustración 111 Creación cuadro de texto	85
Ilustración 112 Propiedades cuadro de texto	86
Ilustración 113 Pestaña Implementación final	86
Ilustración 114 Registro Desarrollo final.....	87
Ilustración 115 Asignación de campo para lista	87
Ilustración 116 Carpeta listas dinámicas	88
Ilustración 117 Asignación lista dinámica para el campo Causa_Falla.....	88
Ilustración 118 Selección de lista dinámica	88
Ilustración 119 Opción guardar y Test work	89

Ilustración 120 Selección de listas dinámicas creadas.....	89
Ilustración 121 Cague de lista dinámica con opciones.....	89
Ilustración 122 Validación de campo para modificar comportamiento.....	90
Ilustración 123 Creación de código Validation en la acción Solicitar Aprobación.....	91
Ilustración 124 Captura de datos de usuario a notificar.....	91
Ilustración 125 Creación registro Email Notification Rule.....	92
Ilustración 126 Asignación de parámetros en Main.....	92
Ilustración 127 Asignación de destinatarios en Header.....	93
Ilustración 128 Asignación de mensaje en Subject and Body.....	93

Índice de tablas

Tabla 1 Generalidades de la empresa.....	11
Tabla 2 Duración actividades en meses.....	14
Tabla 3 Estimación tiempos desarrollo.....	84
Tabla 4 Formato calendario propuesto.....	85
Tabla 5 Formato casos de prueba clearquest.....	94

RESUMEN GENERAL DE TRABAJO DE GRADO

TITULO: IMPLEMENTACIÓN DE REQUISITOS DE SOFTWARE APLICANDO METODOLOGÍAS, PROCEDIMIENTOS Y TECNOLOGÍAS CORPORATIVAS DE IBM.

AUTOR(ES): Michael Daniel Atuesta Cadena

PROGRAMA: Facultad de Ingeniería de Sistemas e Informática

DIRECTOR(A): Johanna Marcela Suarez Pedraza

RESUMEN

Este documento presenta el resultado del proceso de práctica empresarial, realizado para optar por el título de Ingeniero de sistemas e Informática. La práctica empresarial se realizó en la empresa IBM durante un periodo de seis meses, las principales actividades fueron dar soporte a las herramientas de Urbancode Deploy y Rational ClearQuest las cuales son herramientas centrales del banco de occidente para la automatización de los despliegues de sus aplicaciones internas y el control de los procesos del banco, estas herramientas tienen el fin de mejorar los procesos del banco, facilitando la implementación de metodologías agile para los desarrollos, lo que permite brindar una atención de clase mundial, con calidad y alta satisfacción para los clientes, logrando eficiencia operativa, auditable y financiera, permitiendo cumplir con los altos estándares de hoy en día y alcanzar la competitividad requerida para el mercado global. Como resultados de la práctica empresarial se dio solución a todo tipo de solicitudes del cliente y se hizo entrega de desarrollos que contaron con la aprobación del usuario, el desarrollo de casos de pruebas permitió la reducción de errores lo que agilizo los despliegues a producción de los requerimientos, la comunicación constante con el cliente permitió adaptarse a las indicaciones y dar respuesta a los fallos de manera oportuna, se logró experimentar la realidad del entorno laboral y los riesgos y deberes que acarrea la profesión de Ingeniería de Sistemas e Informática y la importancia del manejo apropiado de los datos e la información, a su vez se evidencio como los conocimientos adquiridos durante la carrera, como las asignaturas de programación y la línea de gestión de tecnologías de la información junto con teoría de la negociación facilitaron la comunicación con el cliente y la comprensión de sistemas, procesos y lenguajes de programación previamente desconocidos.

PALABRAS CLAVE:

IBM, Banco de Occidente, Aplicaciones, Flujos, Registros, Despliegues, Urbancode, Clearquest

Vº Bº DIRECTOR DE TRABAJO DE GRADO

GENERAL SUMMARY OF WORK OF GRADE

TITLE: IMPLEMENTATION OF SOFTWARE REQUIREMENTS
APPLYING IBM CORPORATE METHODOLOGIES,
PROCEDURES AND TECHNOLOGIES.

AUTHOR(S): Michael Daniel Atuesta Cadena

FACULTY: Facultad de Ingeniería de Sistemas e Informática

DIRECTOR: Johanna Marcela Suarez Pedraza

ABSTRACT

This document presents the result of the internship process, carried out to opt for the degree of Systems and Computer Engineer. The internship was carried out at the IBM company for a period of six months, the main activities were to support the database and deployment tools of Urbancode Deploy and Rational ClearQuest which are central tools of the financial entity Banco de Occidente for automating the deployment of its internal applications and to control the bank's processes, these tools have the purpose of improving the bank's processes, facilitating the implementation of agile methodologies for developments, which allows the bank to provide world class service, with quality and high customer satisfaction, achieving operational, auditable and financial efficiency, allowing compliance with today's high standards and achieving the competitiveness required for the global market. As a result of the internship, all types of client requests were solved and developments for the tools were submitted with the approval of the client, the correct establishment of test cases during development allowed the reduction of errors, which facilitated the deployment of the requirements, constant communication with the client allowed the developer to adapt quickly to their needs and respond to the errors in a timely manner, it was possible to experience the reality of a real work environment and the risks and duties that the profession of Systems and Computer Engineering entails and the importance of the appropriate management of data and information offered by the client, also it was evidenced how the knowledge acquired during the university courses, such as the programming course and the technology management courses together with the negotiation theory course facilitated the communication with the client and the understanding of systems, methodologies, software and programming languages that were previously unknown.

KEYWORDS:

IBM, Banco de Occidente, Applications, Flows, Records,
Deployments, Urbancode, Clearquest

V° B° DIRECTOR OF GRADUATE WORK

1 INTRODUCCIÓN

El propósito de este documento es presentar las experiencias adquiridas y actividades ejecutadas durante el desarrollo del proceso de práctica empresarial, las cuales fueron realizadas para optar por el título de Ingeniero de sistemas e Informática según los requerimientos de la universidad pontificia bolivariana.

La práctica empresarial fue llevada a cabo en IBM, una empresa de nivel mundial pionera en tecnología que cuenta con valores solidos que buscan ofrecer un servicio de alta calidad y satisfactorio para sus clientes, estos a su vez le permiten crear un ambiente laboral propicio para la continua adquisición y solución de retos que fomenta el crecimiento de sus empleados en una constante búsqueda de innovación, la práctica se ejecutó durante el periodo estipulado de seis meses, este periodo fue separado en dos etapas principales, una etapa de inducción o capacitación y otra para el desarrollo de actividades en un ambiente laboral real.

Durante el periodo de inducción se realizaron diversas capacitaciones en IBM por medio de videos o tutorías directas por parte del personal de IBM, lo que permitió la adquisición de conocimientos sobre diferentes herramientas y metodologías con las cuales IBM ofrece sus servicios y productos al cliente, esto con el fin de conocer más a fondo la organización y a su vez brindar un mejor servicio al cliente.

Finalmente, durante la etapa de ambiente laboral real, se garantizó el correcto funcionamiento de las herramientas Urbancode Deploy y Rational ClearQuest realizando labores de soporte para el cliente del banco de occidente, estos soportes incluyen la realización de procesos, atención a solicitudes, análisis de código, corrección de errores, administración de usuarios y desarrollo de requerimientos.

La ejecución de las actividades asignadas por cada etapa fueron documentadas en este informe final de practica empresarial siguiendo los lineamientos dados por el director de practica de la universidad pontificia bolivariana y el supervisor asignado por parte de IBM, completando de esta forma el requerimiento académico estipulado y permitiendo el desarrollo profesional del estudiante.

2 PRESENTACIÓN DE LA EMPRESA

2.1 GENERALIDADES DE LA EMPRESA

Nombre	IBM DE COLOMBIA & CÍA S.C.A.
Teléfono	Informaciones generales: • Bogotá: +57 1 390 1000 • Cali: +57 2 485 0495 • Barranquilla: +57 5 385 3064 • Medellín CEOH y Corfinsura: +57 4 604 6650
	Servicio técnico: • 57-1-390-2828 Bogotá • 57-2-660-4539 Cali • 57-4-370-7100 Medellín
Dirección	Carrera 53 # 100 - 25 Bogotá – Colombia

Tabla 1 Generalidades de la empresa

2.2 DESCRIPCIÓN DE LA EMPRESA IBM

En lugar de una misión y visión tradicional, IBM se rige bajo un propósito, tres valores y nueve practicas o “1-3-9” para abreviar.

2.2.1 PROPÓSITO

Ser esenciales: Desde nuestra fundación, los integrantes de IBM han sido guiados por un propósito que perdura. Es el porqué de IBM. Es el impacto que buscamos realizar en nuestros clientes, el mundo y cada uno de nosotros. Es lo que hemos aspirado cada día, durante los últimos 100 años. Esto es ser esencial.

2.2.2 VALORES

1. Dedicación para el éxito de cada cliente.
2. Innovación que importa para nuestra compañía y para el mundo.
3. Confianza y responsabilidad personal en todas las relaciones.

2.2.3 PRÁCTICAS

1. El cliente es primero.
2. Escuchar las necesidades, visualizar el futuro.
3. Compartir la experiencia.
4. Constantemente reinventar nuestra empresa y nosotros mismos.
5. Atreverse a crear ideas originales.
6. Valorar la originalidad.
7. Pensar. Preparar. Ensayar.
8. Unirse para hacer las cosas ahora.
9. Mostrar el interés personal.

3 JUSTIFICACIÓN

Hoy en día se observa como cada vez existe una mayor cantidad de información en el mundo, el manejo apropiado y oportuno de dicha información es vital para mantener un nivel alto de competitividad en el mercado, para esto a su vez se generan una gran cantidad de aplicaciones y procesos para poder manipular toda esta información, no siendo esto suficiente se requiere que al igual que la información estas aplicaciones se mantengan en constante evolución para no quedar obsoletas frente a un mundo que se encuentra en constante movimiento.

Por estos motivos el continuo despliegue de versiones en ambientes de metodologías ágiles presentan un problema para con los tiempos de productividad y para mantener el control auditivo de las aplicaciones que se poseen y los procesos que se realizan dentro de estas, la capacidad de poder encontrar los posibles errores a tiempo y mantener un control de versiones que de respaldo para mantener un servicio continuo es vital para todas las empresas, dado que cualquier momento que un proceso se bloquee a causa de una falla implica perdidas grandes para la misma.

Por lo tanto, para mantenerse la velocidad que se necesita para la empresas de hoy en día se requiere la automatización de los procesos que se ejecutan constantemente con el fin de mejorar los tiempos de productividad, a su vez se requiere el acceso oportuno a la información para los equipos de trabajo con el propósito de que todos puedan mantenerse actualizados en los estados actuales de los proyectos o procesos, con el fin de ofrecer acciones rápidas y eficaces que permitan dar un flujo continuo de los mismos, permitiendo también llevar un control de estos facilitando que facilite los trabajos de auditoria.

En torno a esto el banco de occidente busca mejorar sus servicios y el control de sus procesos por medio de las herramientas ofrecidas por IBM de Rational ClearQuest y UrbanCode Deploy, las cuales le permiten realizar la automatización de los despliegues de versiones de sus aplicaciones, permitiendo la salida de versiones continua a producción mejorando drásticamente los tiempos de los mismo y permitiendo a los desarrolladores enfocarse principalmente en el código, a su vez por medio de estas herramientas logran mantener un control de sus procesos, agilizando el flujo de los requerimientos, actividades y solicitudes, manteniendo a los respectivos equipos informados del estado actual de estos y las acciones que se requieren para dar continuidad a su ejecución. Todo esto manejando un correcto historial de estos que su vez facilita su seguimiento y manejo de los procesos de auditoria del banco.

4 OBJETIVO DE LA PRACTICA

Implementar un conjunto de requerimientos de software que respondan a las necesidades específicas de un cliente de IBM, alineando las metodologías y tecnologías de la organización durante las fases del ciclo de vida del software.

4.1 ACTIVIDADES

Dentro de la práctica se realizó investigación sobre tecnologías en tendencia, para luego ser parte de grupos de trabajo Ágiles, en los cuales se realizó la implementación o mantenimiento de funcionalidades, para diferentes cuentas y clientes, aplicando tecnologías actuales de los proyectos o nuevas tecnologías. Durante el proceso de igual manera se participó dentro de las funciones de levantamiento y pruebas.

Las actividades propuestas para la práctica empresarial fueron:

1. Elaborar un documento que presente las bases teóricas del proceso de capacitación recibido por parte de la empresa IBM, de modo que permita describir las nuevas tendencias tecnológicas y metodológicas que se están empleando en el sector productivo del área de la ingeniería del software.
2. Elaborar una propuesta de solución tecnológica a modo de ejercicio simulado que responda a las necesidades de un cliente, soportada en las tecnologías de IBM.
3. Implementar requerimientos de software asignados por IBM, para cumplir con las necesidades de un cliente real, aplicando las metodologías y tecnologías propias de la organización durante el ciclo de vida del software.

4.2 CRONOGRAMA DE ACTIVIDADES

Debido a la naturaleza de la práctica empresarial, el cronograma se dividió en dos etapas, la primera corresponde a la etapa de entrenamiento, que abarco un periodo de tiempo de dos meses, la segunda etapa corresponde a la consultoría con los diferentes clientes, la cual conto con un cronograma puntual según las características del proyecto asignado.

1. Capacitación: Reconocimiento de las tecnologías y metodologías de IBM.
2. Capacitación: Ejercicio simulado de propuesta de solución tecnológica.
3. Consultoría clientes: Asignación laboral para un cliente real.

Actividades/Mes	1	2	3	4	5	6
1.						
2.						
3.						

Tabla 2 Duración actividades en meses

5 DESARROLLO DE LA PRÁCTICA

5.1 RECONOCIMIENTOS DE LAS TECNOLOGÍAS Y METODOLOGÍAS DE IBM

Esta actividad busca presentar las bases teóricas del proceso de capacitación recibido por parte de la empresa IBM, de modo que permita describir las nuevas tendencias tecnológicas y metodológicas que se están empleando en el sector productivo del área de la ingeniería del software.

5.1.1 Capacitaciones

Al inicio de la práctica fue necesario conocer la estructura de la organización, así como las metodologías asociadas a los procesos técnicos, por tal razón se contó con acceso a la base de conocimiento técnico de IBM, por medio de las plataformas de aprendizaje web de “yourlearning”, también se realizaron capacitaciones de la metodología de desarrollo agile y algunas herramientas de desarrollo como lo es Angular. Todo esto con el fin de expandir el conocimiento del practicante y de adentrarlo a la metodología y tecnologías que utiliza la empresa para con sus clientes. A continuación, se presentarán las bases teóricas de las temáticas vistas.

5.1.2 Metodologías de desarrollo agiles

Durante el periodo de capacitaciones se participó en un taller para conocer las metodologías de desarrollo ágil, si bien IBM no realiza todos sus desarrollos con metodologías agile este se encuentra capacitando a sus empleados y clientes con el propósito de dar a conocer los beneficios y facilitar la comprensión de estas nuevas metodologías de desarrollo. Para comenzar se mencionó una definición de agile por parte del capacitador, la cual dice “un conjunto de valores y prácticas para el desarrollo de proyectos, algunos más restrictivos que otros sin embargo todos alineados alrededor de valores comunes y con el foco principal en las personas, todos los métodos agiles son fundamentados en el manifiesto ágil”.

Mayor importancia		Menor importancia
Individuos e interacciones	sobre	Procesos y herramientas
Software que funciona	sobre	Documentación extensiva
Colaboración con el cliente	sobre	Negociación contractual
Respuesta ante el cambio	sobre	Seguir un plan

Ilustración 1 Manifiesto ágil

Dentro de las metodologías ágiles se encuentra la metodología SCRUM el cual es un marco de trabajo iterativo e incremental para la gestión de proyectos que se encuentran dentro de contextos complejos, este cuenta con tres pilares que lo definen los cuales son la transparencia, la inspección y la adaptación. La transparencia hacer referencia al hecho de que todos los procesos, decisiones, acciones deben ser de carácter transparente tanto para el equipo como para el cliente, el cliente debe tener total claridad de lo que se está haciendo y porque se está haciendo, al igual que errores o situaciones que puedan generar cambios en los tiempos estipulados o las entregas. La inspección se refiere al proceso de analizar el estado del proyecto constantemente para identificar desviaciones que se están tomando o errores de concepción del fin último del proyecto y el alcance que este tiene con el fin de corregirlos para direccionar a todos los integrantes a la meta correcta. Por último, la adaptación se refiere a la capacidad del equipo para adaptarse al ambiente de cambio constante que se evidencia en esta metodología, dada la comunicación constante con el cliente, este generara constantemente cambios que se aproximen más al producto final que este desea, pero no visualizaba claramente desde el comienzo. [1]

La metodología cuenta con unos roles que permiten su correcto desarrollo:

- Scrum owner: es el encargado de maximizar el retorno de la inversión del proyecto, generalmente se espera que sea la persona con mayor claridad de el fin último del proyecto para así poder proyectar dicha visión al equipo de trabajo.
- Scrum master: Si bien la metodología no cuenta con una figura líder específica, este rol se encarga gestionar el proceso de Scrum y a las personas que lo ejecutan, busca remover impedimentos, establecer la comunicación entre el grupo y el product owner. Busca guiar a los integrantes del Scrum para que se pueda realizar la metodología eficientemente.

- Scrum team: Se recomienda que tenga un número de integrantes entre 3 y 9 personas, es un equipo auto organizado que gestiona y desarrolla el incremento del producto.

La metodología también presenta varias reuniones es su ejecución, estas son:

- Sprint planning: esta se realiza al iniciar el sprint y permite definir el avance que se le entregara al cliente o product owner y a su vez las tareas que esta entrega comprende para su finalización.
- Daily Scrum: esta reunión se realiza diariamente como su nombre implica, busca poner al tanto al equipo de las tareas que realizo cada integrante el día pasado, los posibles problemas que este presenta y las tareas que realizara el día actual, esto permite evitar el hecho de que los integrantes del grupo no sepan que se encuentra haciendo su compañero y por lo tanto realicen tareas iguales que se conviertan en tiempo perdido, como también permite identificar la existencia de posibles problemas que alente el desarrollo de las actividades y por lo tanto darle solución a estos rápidamente.
- Sprint review: Esta reunión se realiza al finalizar el sprint planeado, busca entregar el avance con el cual se comprometió el equipo con el product owner, este validara el correcto cumplimiento de las funcionalidades que se establecieron se entregarían en la reunión del sprint planning y aportara feedback de los mismos.
- Sprint retrospective: Esta reunión se realiza posterior al sprint review y busca generar retroalimentación del sprint que acabo de transcurrir, esta comprende analizar las cosas que se realizaron correctamente, los problemas que se encontraron junto a la solución que les dio a estos o las posibles mejorar que se pueden realizar para evitar su repetida ocurrencia.

5.1.3 Framework de aplicaciones web Angular

Framework de desarrollo de aplicaciones web, desarrollado utilizando el lenguaje typescript, este lenguaje es de código abierto y actualmente es mantenido por Google, enfocado al desarrollo de aplicaciones web de una sola página las cuales buscan mejorar la experiencia de usuario al ofrecer mayor fluidez similar a la experimentada en aplicaciones de escritorio. Su enfoque de clases de tipo componentes facilitan la reutilización del código realizado, la integración de este y el testing.

Tiene la capacidad de desarrollar para múltiples plataformas como lo son web, web en móvil, móvil local y escritorio nativo. Dada su característica de una sola página estas presentan una velocidad alta y constante, la estructuración de código a partir de componentes es fácil de entender y permite agregar componentes previamente desarrollados por otros usuarios, igualmente angular cuenta con una gran cantidad de documentación, para conocer sus funciones básicas se puede realizar el tutorial de "Tour

of Heroes” encontrado en su página web principal, este tutorial crea una aplicación básica de control y manejo del staff en una agencia de héroes con la cual se pueden aprender las funciones fundamentales del framework angular. [2]

5.1.4 Enterprise design thinking

Es un framework que busca cambiar la manera en la cual se atacan los problemas y las soluciones que se generan de estos para enfocarse principalmente en las experiencias y resultados que el cliente quiere y espera. Para esto es necesario primero identificar correctamente el usuario ya que el usuario es el que determinara el valor que se le está entregando con el producto, también se debe tener la capacidad de reinventar, dado que nada es creado y perfeccionado al mismo tiempo, la primera respuesta no necesariamente es la única posible ni la más optima, partiendo de una idea inicial se puede evolucionar a una idea más completa que brinde mejores resultados y satisfacción al usuario. También es necesario alinear los diferentes puntos de vista del grupo sobre la solución que se plantea para el cliente, para así poderse enfocar todos en la misma meta agilizando el proceso. Dada la importancia de la creatividad se realizaron actividades para la creación de ideas, una de estas consistía en anotar tres objetos aleatoriamente y luego mezclar dichos objetos con los tres objetos escritos por otra persona, esto con el fin de crear un producto o objeto nuevo que junte las características de los otros dos objetos. [3]



Ilustración 2 Ejemplo ejercicio combinar objetos

5.1.5 Personal branding

Al momento de trabajar para IBM no solo es importante representar la marca de manera acorde, sino que también se hace importante el personal branding o marca personal que cada uno tiene, en definición personal branding es el proceso en el cual nosotros nos vendemos a los demás, es la posición personal que identifica que es lo que me hace único y de valor para el cliente. pues esta define como los clientes me perciben como individuo y como empleado de una organización, dado esto nuestro personal Brand se encuentra en testeo constante en acciones como nuestra apariencia, la puntualidad, la actitud que se

tome frente a las diferentes situaciones, incluso desde la foto y organización de nuestra hoja de vida las personas pueden estar evaluando nuestro personal Brand y a partir de esto la actitud o posición que se tome frente a nosotros puede variar tanto para bien como para mal, de igual manera al tener un personal Brand positivo las posibilidades de crecer en lo laboral y como persona se incrementan drásticamente ya que las personas querrán trabajar con esta marca que yo represento, similar a la manera que funcionan las industrias grandes como coca-cola y sus clientes. [3]

5.1.6 Arquitectura SOA y WebSphere Message Broker

La arquitectura SOA es una arquitectura orientada a servicios, permite la creación de sistemas altamente escalables, aumentando el rendimiento y reduciendo costos de las organizaciones, esta arquitectura también cuenta con servicios de infraestructura que permiten realizar la autenticación, controlar el acceso y la realización del registro, también ofrece servicios CRUD para la creación, lectura y actualización de los datos de todos los sistemas back-end al igual que servicios de integración. Esta arquitectura puede comprender un ESB o Enterprise bus service, este es un patrón de arquitectura que propone un único sistema de comunicación para múltiples sistemas de software. [4]

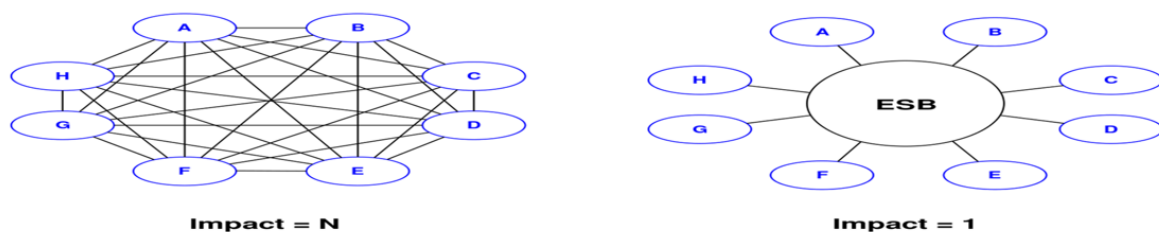


Ilustración 3 Patron ESB. Tomado de Diapositivas WebSphere Message Broker

Para cumplir con este patrón se utiliza el WebSphere Message Broker que permite la integración de los negocios, este cuenta con unos componentes principales que son una red de mensajería, núcleo middleware, servicios de integración, adaptadores y conectores. Dado que cuenta con un núcleo middleware el cual es una arquitectura orientada a servicios y que basa en la integración de aplicaciones, esta capa de middleware es la encargada de resolver las diferencias de las plataformas aplicativas para permitir la comunicación entre ellas, por medio de esto se busca establecer la mínima dependencia entre las aplicaciones, lo cual implica que las aplicaciones se pueden observar como servicios aparte y por lo tanto no son afectadas directamente por el funcionamiento de la otra dado a que todas se comunican por medio del bloque central o middleware, esto permite a su vez que las aplicaciones puedan actuar como clientes pues pueden solicitar un servicio de otra, como también servidoras que ofrecen un servicio a las demás.

Como conclusión el bus de servicios facilita la escalabilidad del sistema al ofrecer una manera de incorporar los servicios y aplicaciones de una forma más organizada, que facilita la depuración de errores, que permite múltiples protocolos de comunicación lo que facilita la integración y controla el redireccionamiento en un solo punto.

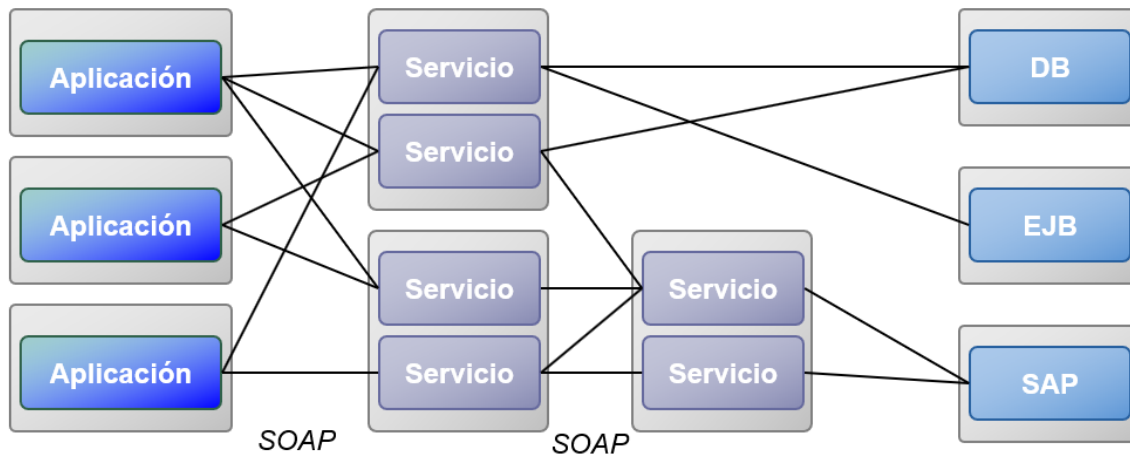


Ilustración 4 Ejemplo sin Service Bus. Tomado de Diapositivas WebSphere Message Broker

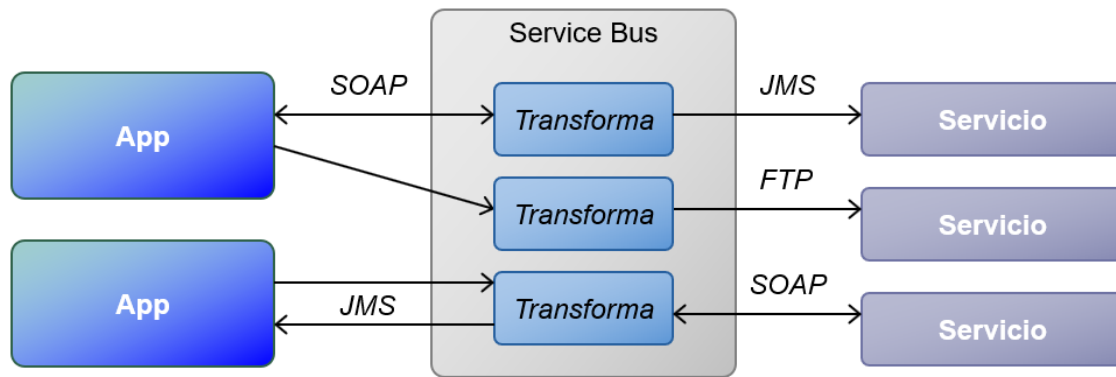


Ilustración 5 Ejemplo Service Bus. Tomado de Diapositivas WebSphere Message Broker

5.1.7 Introducción de IBM Cloud

La plataforma de nube de IBM la cual ofrece un set modular de servicios de infraestructura, permite la creación de entornos de ambiente públicos, privados o híbridos, también permite a las compañías de alta o baja escala construir y correr aplicaciones y servicios modernos altamente escalables y seguros, IBM cloud es impulsado por los proyectos open source más populares del mundo. Facilita el desarrollo e integración de aplicaciones al permitir al

usuario escoger el lenguaje, arquitectura y diseño deseado, posee más de cien APIs y servicios disponibles para el usuario, con una gran cantidad de documentación, tutoriales, templates entre otros para iniciar el proceso del usuario. También ofrece servicios de AI para robustecer las aplicaciones creadas. [3]

5.1.8 Servicio Chatbot de inteligencia artificial Watson

Uno de los servicios ofrecidos dentro del IBM cloud que utilizan la inteligencia artificial de Watson la cual utiliza aprendizaje cognitivo permitiendo que a partir de entrenamiento pueda realizar funciones de interpretación de imágenes, texto o análisis de datos en otras. El chatbot específicamente busca simular la interacción que se tiene con una persona durante un dialogo en un chat, este puede ser entrenado para cualquier tipo de temática, dado a que es impulsado por Watson, a diferencia de otras inteligencias artificiales no busca únicamente imitar la interacción humana sino que si la necesidad surge, posee la capacidad de buscar información por su cuenta en una base de conocimiento, pregunta por mayor claridad en la conversación si es necesario e incluso puede inferir cuando es necesario redirigir al usuario a un agente humano. [3]

5.2 EJERCICIO SIMULADO DE PROPUESTA DE SOLUCIÓN TECNOLÓGICA

Esta actividad busca presentar el desarrollo realizado para un ejercicio simulado realizado durante la práctica laboral de IBM, este ejercicio buscaba analizar el desempeño al momento de analizar posibles problemáticas presentadas por los clientes de IBM y la creatividad para desarrollar y vender soluciones que IBM podría ofrecerles para satisfacer estas necesidades encontradas.

5.2.1 Desarrollo

Se definió aleatoriamente un cliente del grupo de clientes de IBM para cada practicante, dado esto mi cliente asignado fue el “Banco de Bogotá”, posteriormente se debía realizar un análisis del cliente para encontrar posibles fallas o quejas que se presentaran en alguno de sus servicios, esto con el fin de plantear una solución haciendo uso de las tecnologías ofrecidas por IBM que pudiera ser vendida al cliente mencionado.

5.2.2 Análisis del posible problema

Dada la necesidad de encontrar problemas o fallos en los servicios del banco, se partió de la importancia que tienen los clientes para esta organización, ya que es de vital importancia

mantener a los clientes del banco conformes con sus servicios para que estos no vean la necesidad de buscar a la competencia sino que por el contrario estos recomienden el banco a sus conocidos para incrementar la clientela, bien se menciona que es más fácil mantener la confianza del cliente con el banco que recuperarla una vez esta se pierde, no solo perdiendo al cliente sino incrementando la posibilidad de que se esparza una mala imagen del banco. Por lo tanto, se optó por observar las redes sociales y foros del banco con el fin de encontrar la quejas o inquietudes que presentaban estos usuarios del banco.

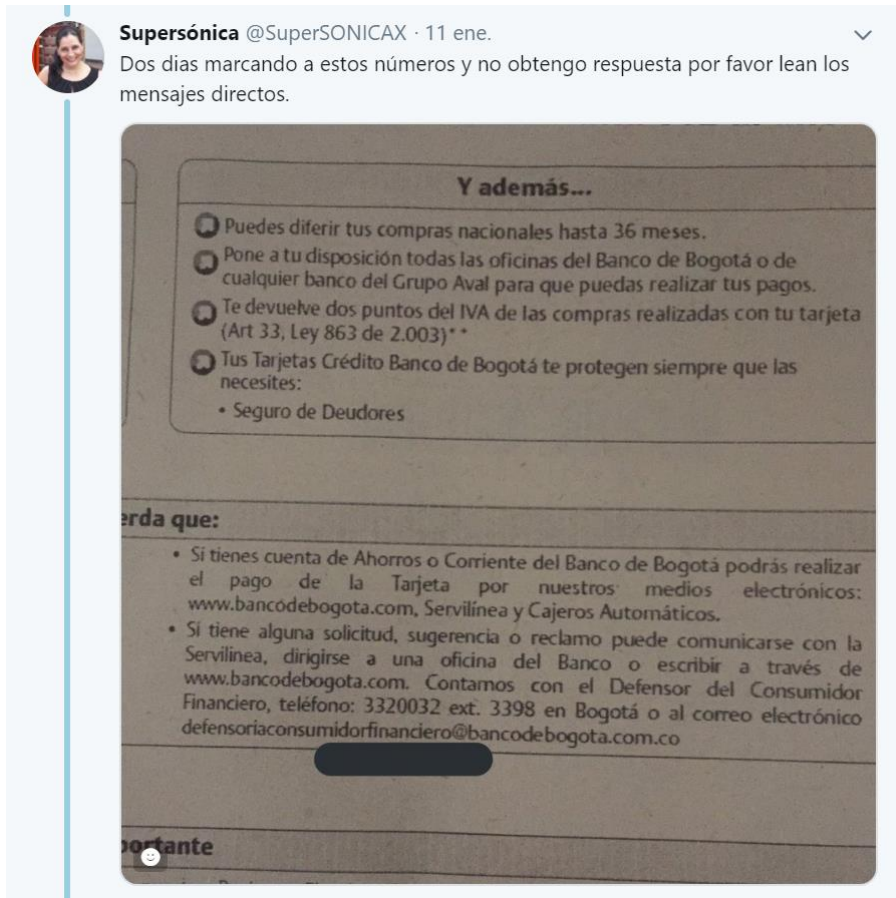


Ilustración 6 Queja redes sociales banco



Ilustración 7 Queja 2 redes sociales banco

Después de observar varios mensajes se logró observar que la atención al cliente no era del todo efectiva para varios usuarios, también varias de las preguntas que se tenían eran de fácil solución, como donde se podía enviar una solicitud, o los horarios de atención del banco con su ubicación, preguntas que se podrían atender rápidamente con el servicio adecuado.

5.2.3 Determinar posible solución

Frente a esta problemática el banco cuenta con una línea telefónica, un chat en línea y una sección de preguntas frecuentes en su página web. Sin embargo, la línea telefónica puede presentar alto tráfico, sumado a esto existe la posibilidad de que el operador no tenga la solución al problema solicitado y tenga que transferir la llamada a otro operador al cual el cliente tiene que mencionar nuevamente toda su problemática para contextualizarlo aumentando el estrés y la insatisfacción de este. Por otra parte, el chat en línea es un servicio de chat manejado por un operador como opción alterna a realizar las llamadas, este al requerir un agente tiene una limitante de tiempo, ya que el cliente tiene que esperar de igual forma a que un agente del banco responda su solicitud, sin mencionar que tiene un horario de atención limitado por lo que no estará siempre disponible al cliente.

Chat en Línea

Lunes a Viernes 7:00 am - 9:30 pm

Sábado 8:00 am - 5:00pm

INGRESAR

Ilustración 8 Horario Chat en Línea Banco

Por último, está la sección de preguntas frecuentes la cual al contener una gran cantidad de información puede no ser muy fácil de navegar para el cliente, especialmente para los que no están acostumbrados a navegar en una página web.

Partiendo de lo mencionado anteriormente surge la idea de un chat manejado por inteligencia artificial el cual se encuentre disponible siempre al cliente, dada su programación de las preguntas frecuentes que presentan los clientes puede responder fácil y rápidamente o incluso redireccionar correctamente al cliente al lugar que requiera o solicite. Permitiendo que siempre esté capacitado para responder las preguntas de un cliente sin importar además del tráfico de personas que requieran su servicio.

5.2.4 Definir herramienta de IBM para su solución

Partiendo de la propuesta de solución mencionada se observa la herramienta de Watson Assistant ofrecida por IBM, esta permite la creación de un chatbot que busca simular la interacción de chat en línea con un agente utilizando la inteligencia artificial de IBM Watson, en la documentación ofrecida por IBM se menciona los beneficios que presenta el uso de esta herramienta a partir de estudios realizados. Uno de estos análisis menciona que las personas en general prefieren el autoservicio sobre la necesidad de tener que hablar con un agente para obtener algún servicio, también se mencionan los siguientes porcentajes:

- El 50% de las llamadas no resuelven el problema o requieren escalar a otro agente.
- El 73% de las personas mencionan que “valorar sus tiempos” es lo más importante para ellos.
- El 70% de las personas prefieren enviar un mensaje sobre llamar.
- El 61% de las llamadas fallidas pudieron haber sido solucionadas con un mejor acceso a la información.
- Se calcula que 62 billones de dólares se pierden en los estados unidos a causa de la mala atención al cliente.
- El 91% de los clientes insatisfechos con un servicio no volverán a usar dicho servicio.
- Se reducen los costos de atención al cliente, ya que normalmente se invierte en la contratación de múltiples agentes los cuales adicionan un costo adicional de capacitación, el cual se pierde al momento de retirarse el agente de la empresa.
- Se disminuye el tiempo perdido de los agentes al responder llamadas, permitiendo así invertir su tiempo en otras labores.

Los servicios de Watson permiten también responder llamadas telefónicas, convertir texto a voz o inversamente. Sin embargo, por causas del demo únicamente se trabajó con el servicio de chatbot el cual realiza la conversación con el usuario por medio de una ventana de chat, el cual se programaría con algunas de las preguntas frecuentes de los clientes del banco.

5.2.5 Implementar solución / desarrollo de DEMO

La implementación se realizó por medio de un servidor desarrollado utilizando Nodejs para establecer la comunicación con Watson y un html que permitía la visualización del chat, se usó como base un template que se encargaba de establecer la comunicación con la Api de Watson assistant y se modificó para cumplir con las funciones deseadas.

En el código se encuentran las variables de entorno para comunicarse con la Api de Watson.



```

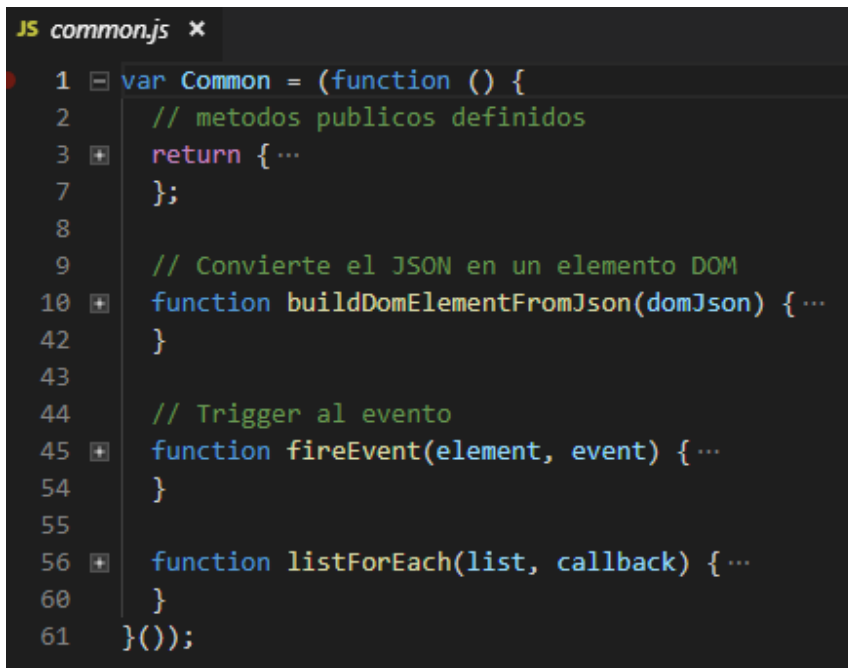
1  # Environment variables
2  WORKSPACE_ID=25bbc6ce-d764-4f5d-8ce6-2011b0ec7bb9
3  # You need to provide either username and password
4  ASSISTANT_USERNAME=apikey
5  ASSISTANT_PASSWORD=V1kIGoUqjPRDfQUEQJY4ovWGZP5U39EOt9pEMTUasws1
6  # OR IAM API key and URL
7  ASSISTANT_IAM_APIKEY=
8  ASSISTANT_IAM_URL=
9  ASSISTANT_URL=https://gateway.watsonplatform.net/assistant/api
10

```

Ilustración 9 Archivo. env

La ejecución de los procesos se realizaba a partir de archivos de javascript, el archivo “api.js” es el encargado de realizar la comunicación con la Api, esta comunicación se realiza por medio “request-response” o petición respuesta, los cuales envían el texto ingresado por el usuario y capturan los mensajes de respuesta de la Api.

Dado que los mensajes son enviados en formato JSON el archivo “common.js” se encarga de convertir este formato en un elemento DOM para facilitar su tratamiento con el html de visualización.



```

1  var Common = (function () {
2      // metodos publicos definidos
3      return { ...
7      };
8
9      // Convierte el JSON en un elemento DOM
10     function buildDomElementFromJson(domJson) { ...
42     }
43
44     // Trigger al evento
45     function fireEvent(element, event) { ...
54     }
55
56     function listForEach(list, callback) { ...
60     }
61     }());

```

Ilustración 10 Archivo common.js

El archivo “conversation.js” es el encargado de comunicarse con el html para modificar la parte visual del chat, se encarga de actualizar el chat con los mensajes ingresados por el usuario y recibidos de la Api de Watson, también de los efectos visuales que presenta el chat.

```
JS conversation.js ●
1  var ConversationPanel = (function () {
2    var settings = {...
13  };
14  // Definicion metodos publicos
15  return {...
19  };
20  // Iniciar el modulo
21  function init() {...
25  }
26  // actualiza utilizando la funcion displayMessage cuando se envian o reciben mensajes
27  function chatUpdateSetup() {...
39  }
40  // Crea un inputBox que subraye el texto conforme se escribe
41  function setupInputBox() {...
100 }
101 // Muestra el mensaje del usuario o watson que fue enviado o recibido
102 function displayMessage(newPayload, typeValue) {...
119 }
120 // Funcion recursiva para agregar respuestar al area del chat
121 function setResponse(responses, isUser, chatBoxElement, index, isTop) {...
141 }
142 // Construir nuevo DOM element del mensaje
143 function getDivObject(res, isUser, isTop) {...
162 }
163 // valida si el autor del mensaje, si es watson, el usuario o ninguno
164 function isUserMessage(typeValue) {...
171 }
172 function getOptions(optionsList, preference) {...
197 }
198 function getResponse(responses, gen) {...
232 }
233 // Construye nuevo elemento generico a partir del message payload
234 function buildMessageDomElements(newPayload, isUser) {...
269 }
270 // mueve al final de la ventana del chat
271 function scrollToChatBottom() {...
274 }
275 function sendMessage(text) {...
285 }
286 // Maneja las entradas del input Box
287 function inputKeyDown(event, inputBox) {...
294 }
295 }());
```

Ilustración 11 Archivo conversation.js

Por último, se encuentra el archivo “app.js” el cual es el encargado de establecer la comunicación con la Api del servicio a partir de las variables de entorno previamente ingresadas.

```
JS app.js
1  'use strict';
2
3  var express = require('express'); // app server
4  var bodyParser = require('body-parser'); // parser para los post requests
5  var AssistantV1 = require('watson-developer-cloud/assistant/v1'); // watson sdk
6
7  var app = express();
8
9  app.use(express.static('./public'));
10 app.use(bodyParser.json());
11
12 var assistant = new AssistantV1({
13   version: '2018-07-10'
14 });
15
16 app.post('/api/message', function (req, res) {
17   // ...
18 });
19
20 /**
21  * Updates the response text using the intent confidence
22  * @param {Object} input La petición para el servicio
23  * @param {Object} response La respuesta del servicio
24  * @return {Object} La respuesta con el mensaje actualizado
25  */
26 function updateMessage(input, response) {
27   // ...
28 }
29
30 module.exports = app;
```

Ilustración 12 Archivo app.js

Una vez establecida la comunicación se hace necesario programar ahora el servicio para que cumpla con la función deseada, que en este caso es responder las preguntas frecuentes que ingrese el usuario del banco.

El servicio de Watson assistant consta principalmente de “Intents” y “Entities” los cuales buscan comprender el mensaje enviado por el usuario y ayudar a interpretarlo, para efectos de este demo solo se utilizaron Intents por lo que no se entrara en detalle sobre las Entities. Los Intents buscan capturar la intención que tiene el usuario con el mensaje que ingreso, por ejemplo, este mensaje puede tener la intención de agradecer, despedirse, solicitar algo, saludar entre muchas otras intenciones.

☐ Intent (23) ▼

☐ #AbonosExtraordinarios_TC

☐ #Adquirir_tarjetaDebito

☐ #Agradecimiento

☐ #Billete_falso

☐ #CambiarClaveSecreta_TD

☐ #CausaRobo_TD

Ilustración 13 Intent Watson assistant

Para este demo se contó con 23 Intent para interpretar la intención del usuario, estas intenciones son alimentadas a través de machine learning por medio de ejemplos que se le ingresan a Watson para que este se entrene pueda realizar una correcta interpretación de los mensajes, dado esto entre más ejemplos se ingresen más acertado será Watson al interpretar, en caso de que Watson no logre interpretar el mensaje, este se define como “irrelevant”. En la siguiente imagen se observan los ejemplos que alimentan el Intent de #Agradecimientos.

☐ User examples (4) ▼

☐ Estoy agradecido ✎

☐ Gracias ✎

☐ Gracias por la ayuda prestada ✎

☐ Muchas gracias ✎

Ilustración 14 Ejemplos de entrenamiento de Intent #Agradecimientos

Por último, se define la estructura de dialogo que el chatbot tendrá para responder los mensajes del usuario, este utiliza los Intents previamente definidos para acceder la secuencia de dialogo relevante con el mismo.

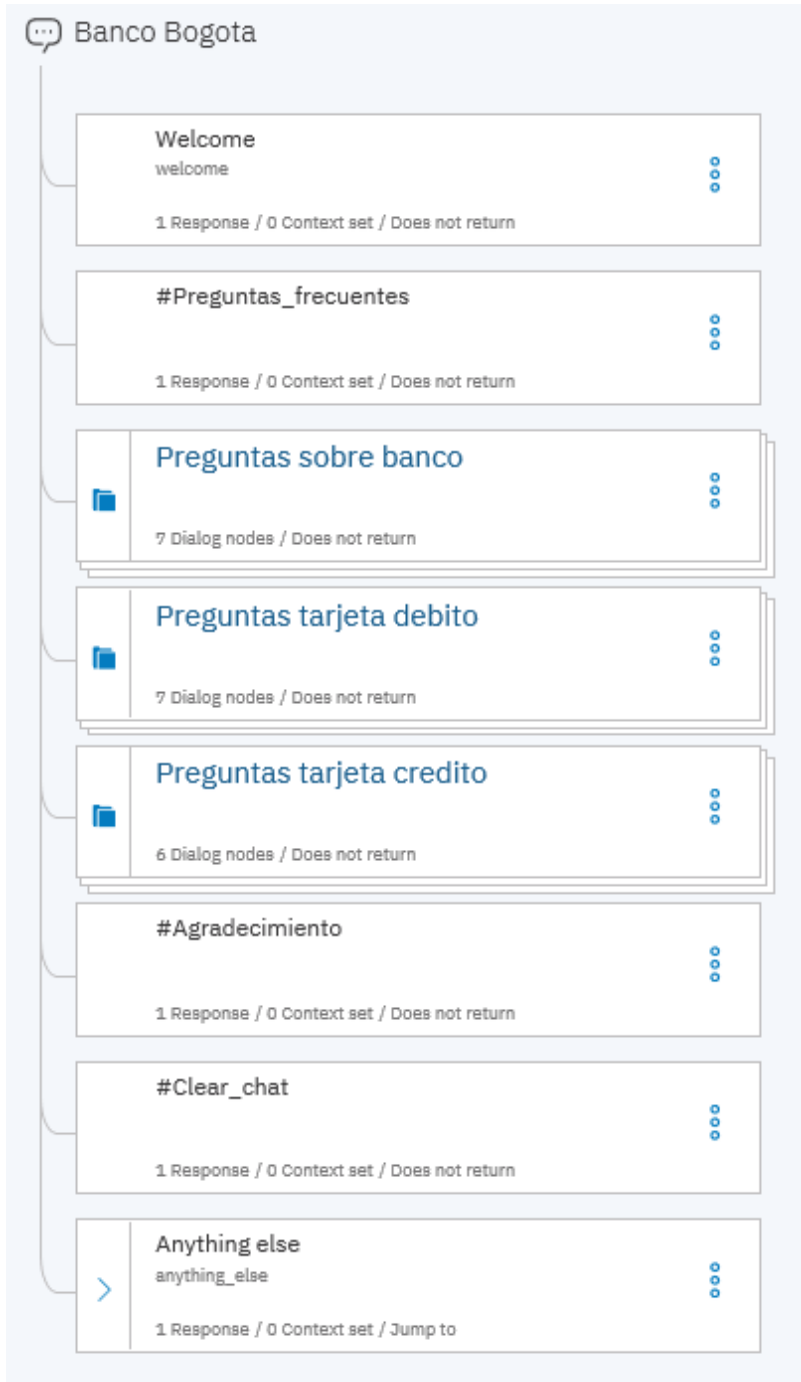


Ilustración 15 Estructura nodos de dialogo

Básicamente esta estructura sigue un orden desde arriba hacia abajo, pregunta en el nodo de dialogo si el mensaje ingresado cumple con el Intent solicitado para activar este dialogo, en caso de que cumpla mostrara el dialogo y ejecutara las acciones programadas para este nodo de dialogo, en caso de que no, realizara la validación con el siguiente nodo. Los diálogos también pueden presentar sub-diálogos o nodos hijos, los cuales implican que al ingresar a un nodo de dialogo que tenga estos, el siguiente mensaje ingresado por el usuario se validara a través de los sub-diálogos presentes y no para toda la estructura de diálogos. Por último, si el mensaje no cumple con los parámetros solicitados de alguno de los nodos de diálogo este ejecutara el nodo de “Anything else”.

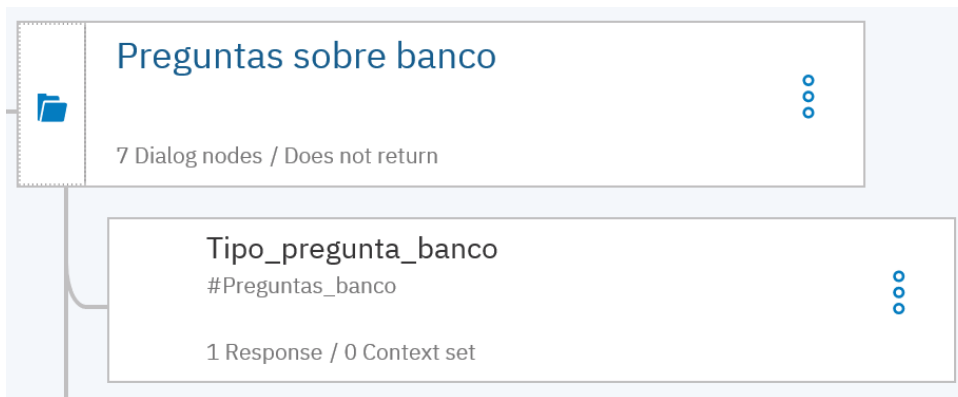


Ilustración 16 Carpetas de nodos de dialogo

También se pueden crear carpetas que contengan nodos de dialogo con el fin de organizar los nodos.

```
1 {
2   "output": {
3     "generic": [
4       {
5         "values": [
6           {
7             "text": "Hola soy el agente virtual del banco de Bogotá."
8           }
9         ],
10        "response_type": "text",
11        "selection_policy": "sequential"
12      },
13      {
14        "title": "Puedo ayudarte con preguntas respecto a los siguientes
15 temas.",
16        "options": [
```

Ilustración 17 Contenido JSON del mensaje

Dentro de los nodos también se puede observar y manipular directamente el JSON de respuesta que envía el nodo de dialogo.

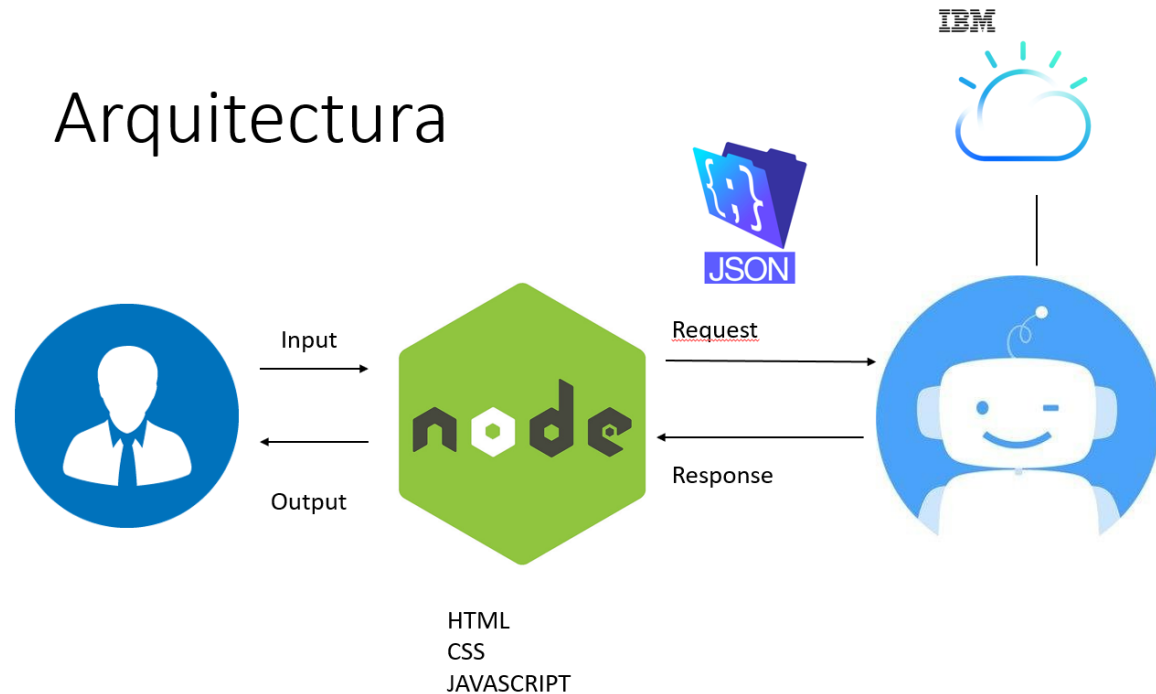


Ilustración 18 Arquitectura básica de la app

Por último, esta sería una arquitectura básica de la aplicación, esta cuenta con entradas de mensajes del usuario, estos son procesados por el servidor de nodejs el cual utiliza html, css y javascript, el servidor envía el mensaje al Watson assistant en formato JSON por medio de un request, este mensaje es procesado por Watson el cual se encuentra alojado en el IBM Cloud y retorna un response de formato JSON, nuevamente el servidor lo captura y procesa. Finalmente, este mensaje se muestra al usuario, el cual puede continuar con la operación.

5.2.6 Resultados

Como resultado se puede observar las siguientes capturas del demo de la aplicación:

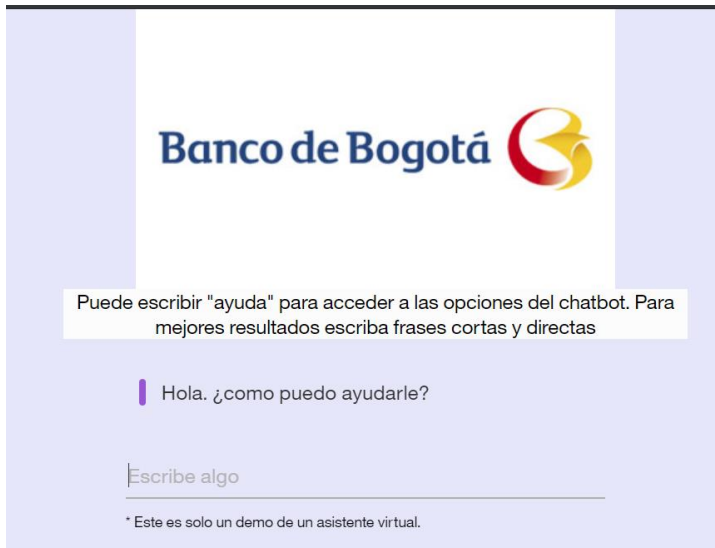


Ilustración 19 Pantalla principal de la app

Esta es la pantalla inicial del chatbot el cual realiza un saludo básico y presenta unas instrucciones de uso.

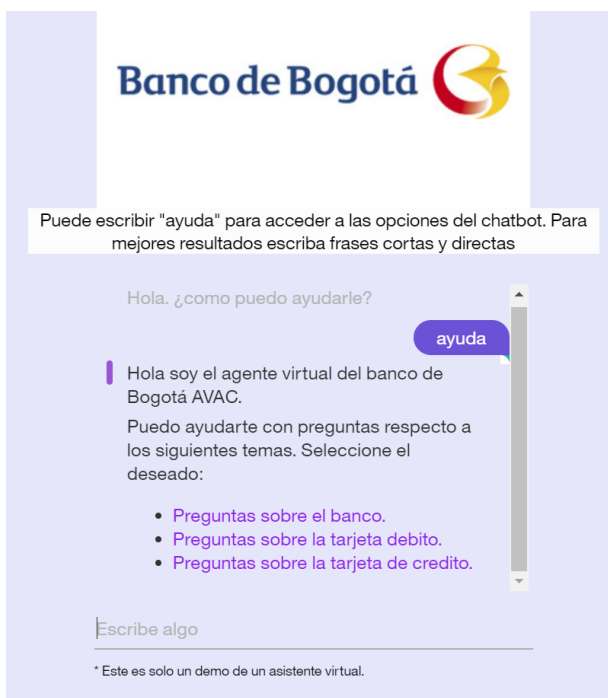


Ilustración 20 solicitud de ayuda en el chatbot

Si el usuario solicita ayuda el chatbot responderá mostrando las opciones de uso que este posee. Estas opciones se pueden acceder dándoles clic o de igual manera realizando la pregunta respectiva.

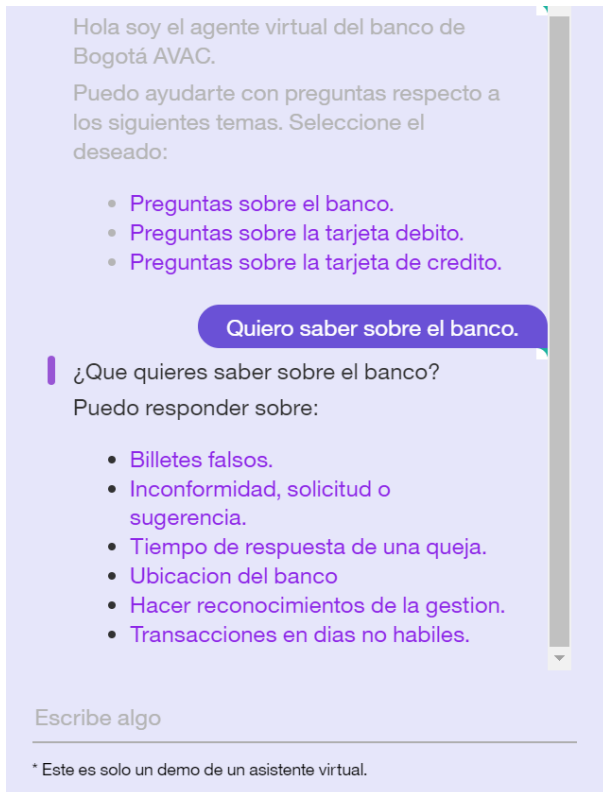


Ilustración 21 Selección de opciones del chatbot

En caso de tener una pregunta abierta el chatbot mostrara nuevamente opciones de las respuestas que puede ofrecer sobre el tema mencionado. Sin embargo, si el usuario pregunta directamente sobre una temática en específico el chatbot le indicara directamente la respuesta sin necesidad de realizar todo este procedimiento, reduciendo de esta manera el tiempo que gasta el usuario en responder su inquietud si este la define claramente.

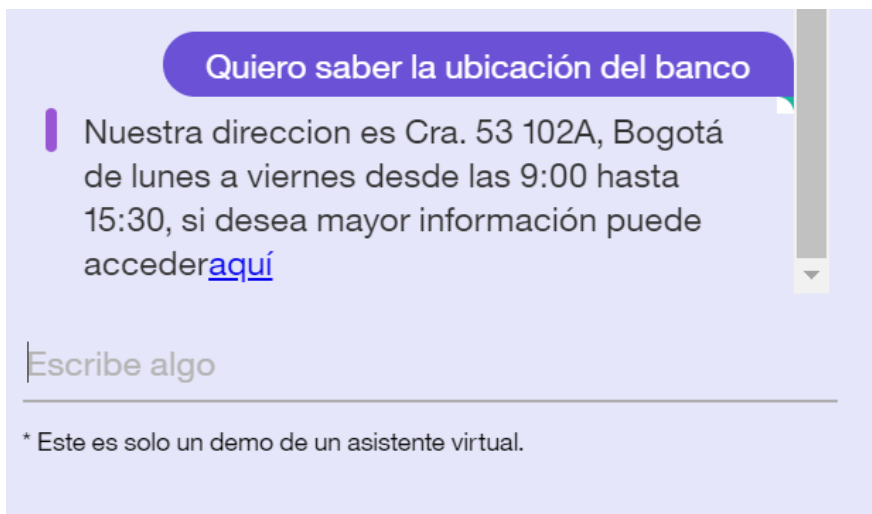


Ilustración 22 Respuesta con hipervínculo del chatbot

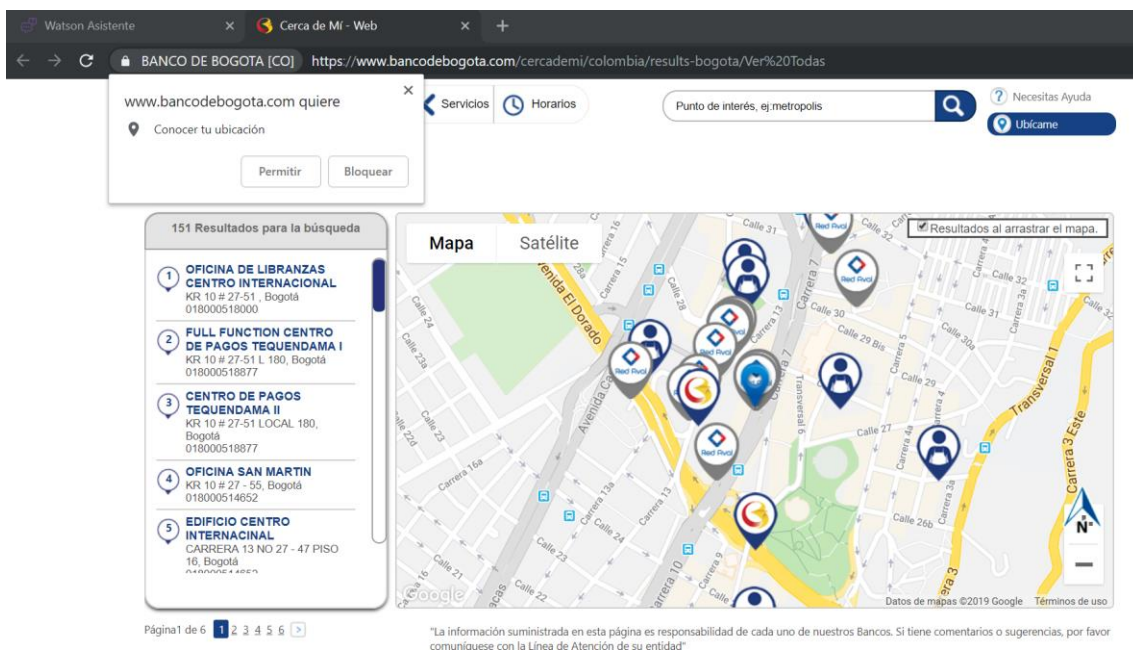


Ilustración 23 Pagina destino del hipervínculo, ubicación bancos cercanos

El chatbot también permite la creación de hipervínculos facilitando la redirección del usuario a lugares con la información que este requiera. En este caso en particular redirige al usuario a la página con las ubicaciones de los bancos de Bogotá cercanos a él.

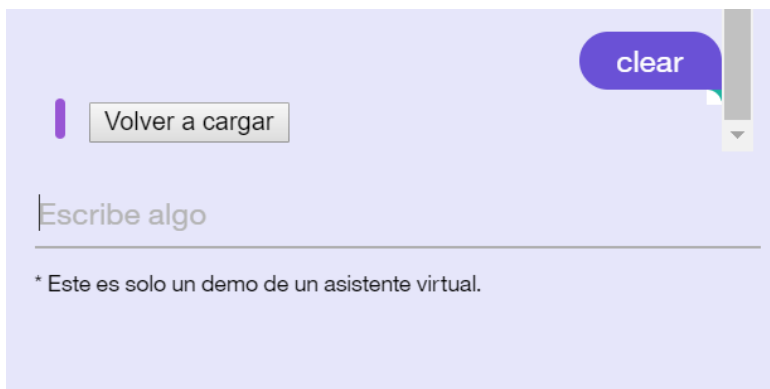


Ilustración 24 Creación botón de recargar

El chatbot también permite la creación de botones para ejecutar operaciones, en este caso crea un botón para actualizar la página y resetear el estado del chatbot.

5.2.7 Conclusiones

La aplicación de Watson assistant comprende diferentes mensajes escritos por el usuario y los responde acorde la interpretación de su intención. Los tiempos de espera de las respuestas son cortos y dada su naturaleza de bot se encuentra disponible para la cantidad de usuarios que accedan a su servicio sin límites horarios, esta permite a su vez el redireccionamiento de los usuarios, permitiendo así ubicarlos en la información que estos requieran o comunicarlos a el responsable de una temática si esta problemática presentada es de un nivel de complejidad mayor. La aplicación también a su vez es escalable permitiendo que se pueda realizar un mayor entrenamiento de las funciones actuales, como también el ingreso de nuevas intenciones o diálogos según se requiera, también existe la posibilidad de acoplarlo con los otros servicios de Watson permitiendo agregar funciones como traductor de texto, reconocimiento de imágenes, responder llamadas y convertir el texto a voz entre otras. Finalmente, el entrenamiento realizado para Watson no se perderá a diferencia de las capacitaciones de agentes normales que se retiran de la organización.

5.3 ASIGNACIÓN LABORAL PARA UN CLIENTE REAL

Esta actividad busca presentar el trabajo realizado para un cliente de IBM (Banco de Occidente) durante la práctica laboral de IBM, este trabajo consta de dar soporte de las herramientas de IBM implementadas por el banco “UrbanCode Deploy” y “Rational ClearQuest”, este soporte puede ser corrección de errores, desarrollo dentro de las aplicaciones, seguimientos de los procesos y validaciones entre otras situaciones que se puedan presentar con las herramientas.

5.3.1 HERRAMIENTAS

5.3.1.1 UrbanCode Deploy

Esta herramienta de IBM permite realizar la automatización de los procesos de despliegue de aplicaciones, permitiendo de esta manera mantener un mayor control de la aplicación durante su ciclo de vida, permitiendo la realización de despliegues más veloces y confiables, ayuda también a realizar la auditoria de los mismos facilitando la transparencia y orden de la ejecución. [5]

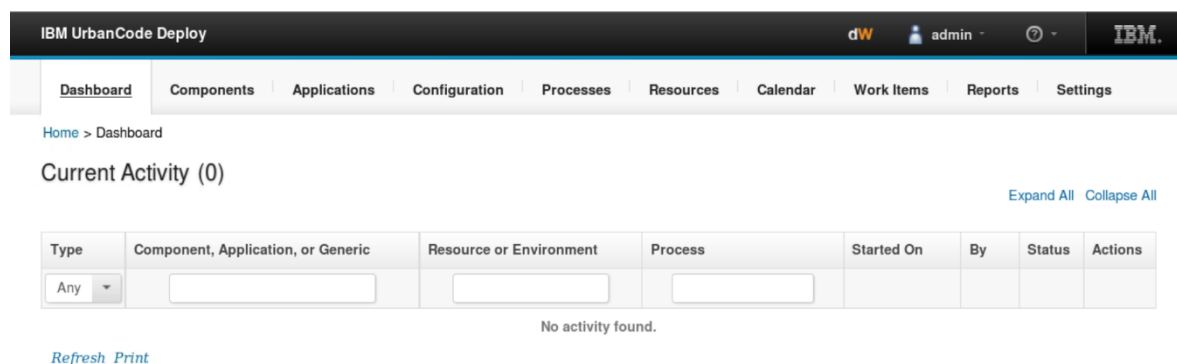


Ilustración 25 Pantalla principal UCD web

La herramienta permite la creación de usuarios a partir del servidor LDAP del banco, facilitando el proceso de autenticación, también permite la creación de grupos con el fin de estipular los permisos de acceso y modificaciones que cada usuario tenga bajo su jurisdicción, por ejemplo un usuario que solo requiera participar como un observador del proceso no necesita permiso para desplegar la aplicación, por lo tanto se crea un grupo de observadores para agregar a todos los usuarios que únicamente requieran observar el proceso pero no deban tener acceso a la creación o modificación del mismo. Por otra parte, existe el grupo de despliegue los cuales pueden además de observar el proceso realizar la operación del despliegue sobre una aplicación, de igual forma se pueden diferenciar los

tipos de acceso de diferentes usuarios para los múltiples roles que estos puedan ejercer sobre la aplicación. También es importante mencionar que la creación de estos grupos es diferente para cada aplicación permitiendo nuevamente un mayor control sobre el acceso a las aplicaciones pertinentes para cada usuario.

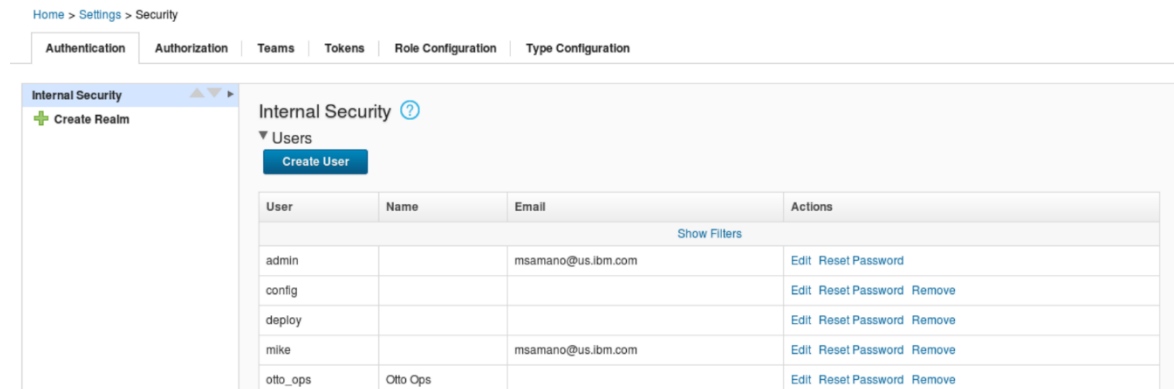


Ilustración 26 Creación de usuario UCD

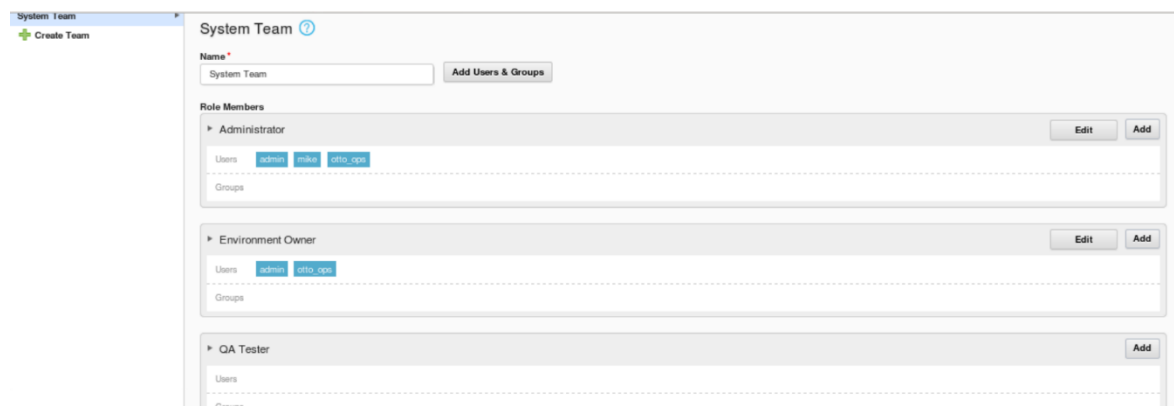


Ilustración 27 Creación de equipos UCD

Para llevar a cabo el proceso de despliegue de una aplicación se realiza por medio de un agente el cual la aplicación reconoce como un recurso, cada servidor de destino de la aplicación a desplegar requiere la existencia de un agente en el mismo ya que este agente es el encargado de realizar los procesos solicitados por el servidor principal de UrbanCode Deploy, básicamente un agente entonces es un proceso ligero que se encuentra alojado en el servidor destino y que se comunica directamente con el servidor principal para realzar las peticiones que este tenga para él, en caso de que el agente no se encuentre en uso, este se ejecutara como un proceso en segundo plano permitiendo el ahorro de los recursos del servidor. De esta manera es que las tareas normalmente realizas por una persona

pueden pasar a ser dirigidas por UrbanCode Deploy a través de un agente, permitiendo de esta manera el concepto de automatización.

Name	Description	Status	License	Version
oldagent		Offline	Unlicensed	6.1.0.0.520179
ucr.local		Online	Unlicensed	6.1.1.1.628129

Ilustración 28 Estado agentes UCD

Name	Inventory	Status	Description
ucr.local	(View Agent)	Online	
UCR Datacenter	(View Agent)	Online	

Ilustración 29 Agente asignado a una aplicación

Partiendo de los agentes explicados previamente, se hablaba de solicitudes que se les envían para que estos realicen ciertos procesos, UrbanCode Deploy define estos procesos a partir de componentes y aplicaciones. Una aplicación comprende diferentes componentes, como por ejemplo se puede pensar en una página web como una aplicación, pero para que esta funcione requiere a su vez un componente de base de datos, un componente que contenga todos los archivos de imágenes de la página y un componente que tenga el condigo fuente de la página. Por lo tanto, se tiene una aplicación que contiene tres componentes diferentes.

Component	Description	Actions
Pet Breeder Site-APP		Run Process Remove
Pet Breeder Site-DB		Run Process Remove

Ilustración 30 Aplicación con dos componentes

Los componentes también manejan un control de versiones a partir del tipo de control que se defina para estos, el más general es a partir de un repositorio el cual separa las carpetas

en versiones, las cuales defines los artefactos o archivos actuales que este componente contenga en definida versión.

Component: Pet Breeder Site-APP

Created By: admin
Created On: 5/29/2013, 6:06 PM
Template: Simple Template
Used By: Pet Breeder Site

Dashboard | Usage | Configuration | Calendar | **Versions** | Processes | Changes

Version	Statuses	Type	Created By	Date	Description	Actions
		Any				
1.0.126		Full	admin	12/4/2014, 3:07 PM	Created in UrbanCode Build	Compare Delete Copy
1.0.125		Full	admin	12/4/2014, 3:05 PM	Created in UrbanCode Build	Compare Delete Copy
\$(prstated/StampName)		Full	admin	12/4/2014, 3:02 PM	Created in UrbanCode Build	Compare Delete Copy
\$(prstampName)		Full	admin	12/4/2014, 3:00 PM	Created in UrbanCode Build	Compare Delete Copy
2.0.1		Full	admin	5/29/2013, 6:18 PM		Compare Delete Copy

Ilustración 31 Versiones de componente

Cada componente puede tener uno o más procesos propios de él, estos procesos pueden ser tanto un proceso de despliegue, como un proceso de pruebas del componente, o la realización de un despliegue con diferentes parámetros y operaciones al despliegue original, entre otras muchas opciones.

Component: Pet Breeder Site-APP

Created By: admin
Created On: 5/29/2013, 6:06 PM
Template: Simple Template
Used By: Pet Breeder Site

Dashboard | Usage | Configuration | Calendar | Versions | **Processes** | Changes

[Create Process](#)

Process	Description	Actions
Deploy (Simple Template)		Edit Copy Delete
Run Tests (Simple Template)		Edit Copy Delete
Uninstall (Simple Template)		Edit Copy Delete

3 records - [Refresh](#) [Print](#) 1 / 1 Rows 10

Ilustración 32 Procesos de componente

Los procesos de un componente constan de un inicio, un fin y uno o más pasos entre estos que son definidos por el usuario para realizar las opciones que este desee, cada paso puede cumplir con diferentes operaciones simples o complejas, como puede ser la descarga de los artefactos o archivos que se desean desplegar al servidor destino, la modificación de algún archivo de configuración, apagar la aplicación para poder realizar posteriormente su actualización, el reinicio de una aplicación, o el movimiento y extracción de los artefactos en la carpeta de la aplicación en el servidor destino, entre muchas otras opciones.

Para facilitar la creación de los pasos previamente mencionados, UrbanCode Deploy permite la creación de estos en el editor de procesos, este es un editor visual que permite agregar los pasos a través de una función de arrastrar y soltar los pasos en un espacio de diseño, similar a la creación de diagramas de flujo.

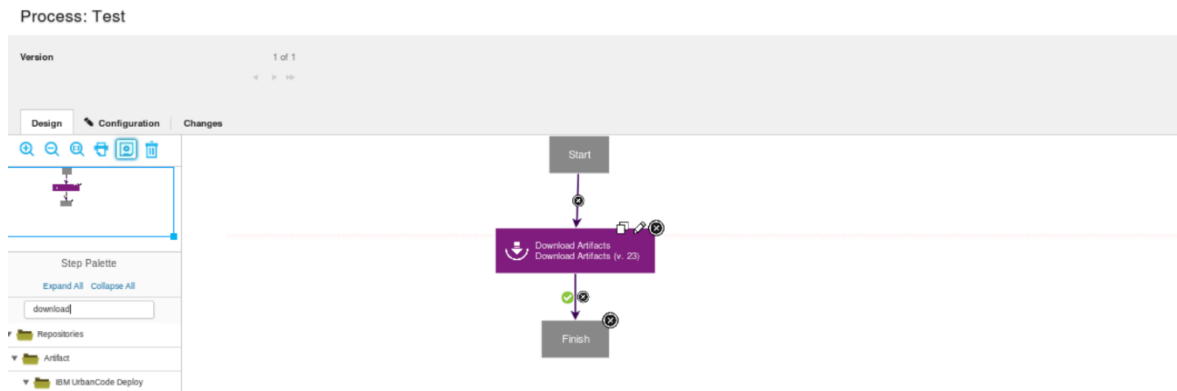


Ilustración 33 Proceso básico de editor de procesos

UrbanCode Deploy ofrece varios pasos generales que permiten descargar archivos, obtener información del sistema, entre otros muchos pasos de utilidad general. La herramienta también ofrece plugins los cuales agregan otros pasos que permiten la integración con otras herramientas de despliegue y múltiples servidores de aplicación, estos permiten el acceso a muchos procesos frecuentes que se realizan en estos como la descarga de artefactos, adquirir información pertinente de un entorno, o por ejemplo en casos de servidores Linux como lo es el servidor Redhat, UrbanCode Deploy permite el acceso a comandos Shell, lo cual permite la automatización de los procesos que se desean realizar en este servidor.

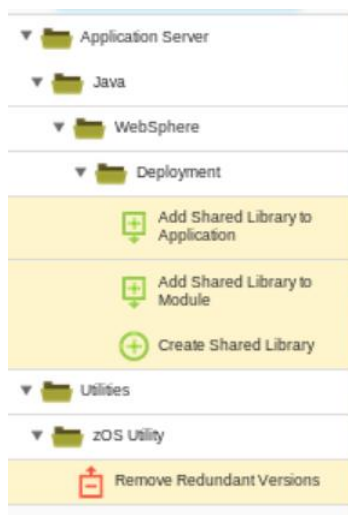


Ilustración 34 Plugin UCD

Estos se pueden adquirir directamente de la página oficial de UrbanCode y observar información detallada de cada uno de estos para encontrar de esta manera el plugin necesario para la organización. [5]

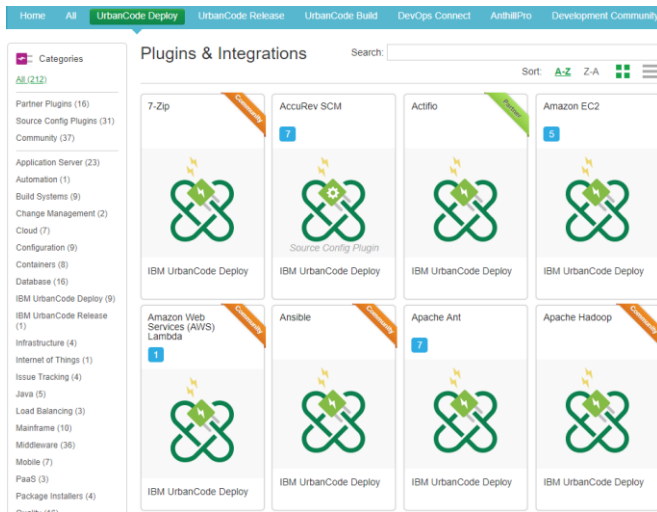


Ilustración 35 Pagina plugin UCD "https://developer.ibm.com/urbandcode/plugins/ibm-urbandcode-deploy/"

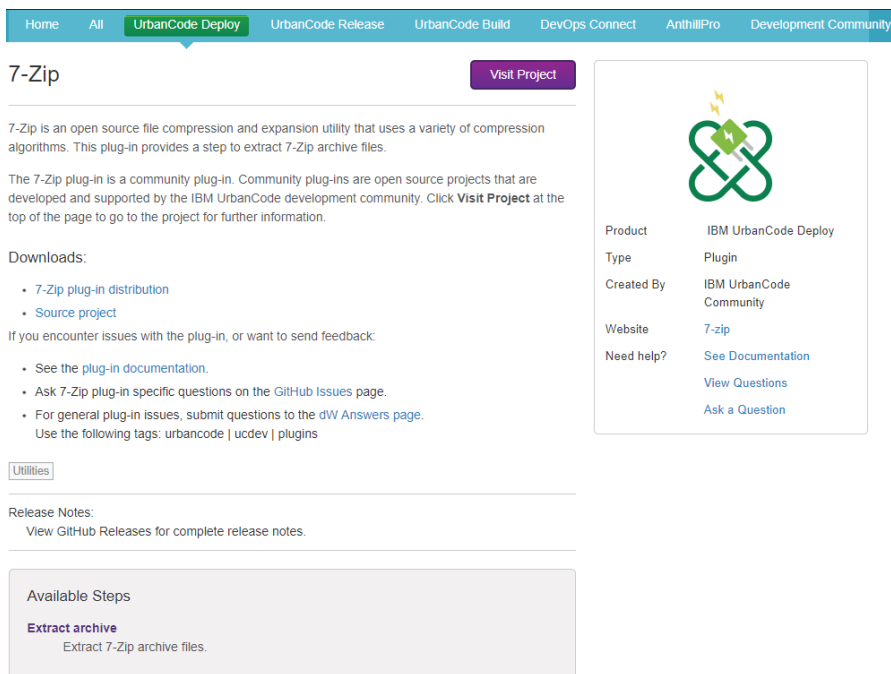


Ilustración 36 Descripción plugin 7zip

Como por ejemplo se observa en la ilustración 36 la descripción del plugin de 7zip el cual ofrece el paso para la extracción de un archivo que se encuentre comprimido en este formato.

Para finalizar el editor de procesos permite relacionar los pasos y definir su flujo al siguiente paso según cumplan con alguna de las tres condiciones predefinidas, si el paso anterior completa correctamente marcado con un círculo verde y un símbolo de visto bueno, si el paso anterior encuentra un error marcado un círculo rojo y un visto malo y si debe realizar el siguiente paso sin importar el estado del paso anterior marcado con un círculo en gris. Esto permite un mayor control sobre los procesos y definir acciones en caso de encontrar errores según se necesite.

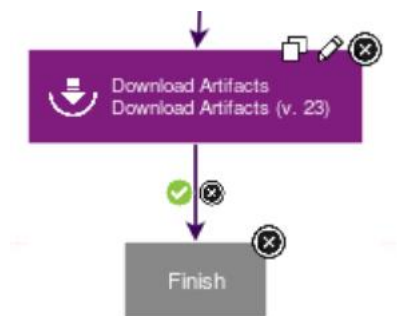


Ilustración 37 Continuar si el proceso es exitoso

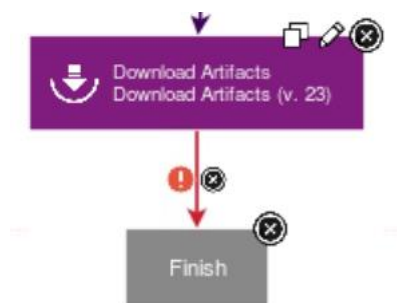


Ilustración 38 Continuar si el proceso es fallido

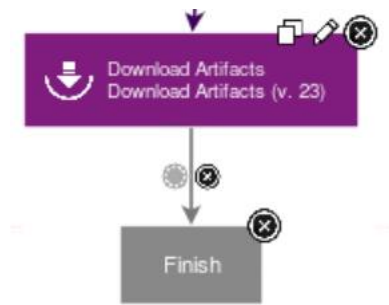


Ilustración 39 Continuar sin importar el resultado

Luego de definir los procesos de cada uno de los componentes es necesario realizar la creación del proceso de despliegue de la aplicación, este proceso generalmente indica los componentes que se desean desplegar de la aplicación junto con el proceso de este que se desea realizar, validando también el orden en el que se desea que estos se realicen. Por lo tanto, una aplicación también puede tener definidos múltiples procesos según el despliegue de los componentes que desea realizar.

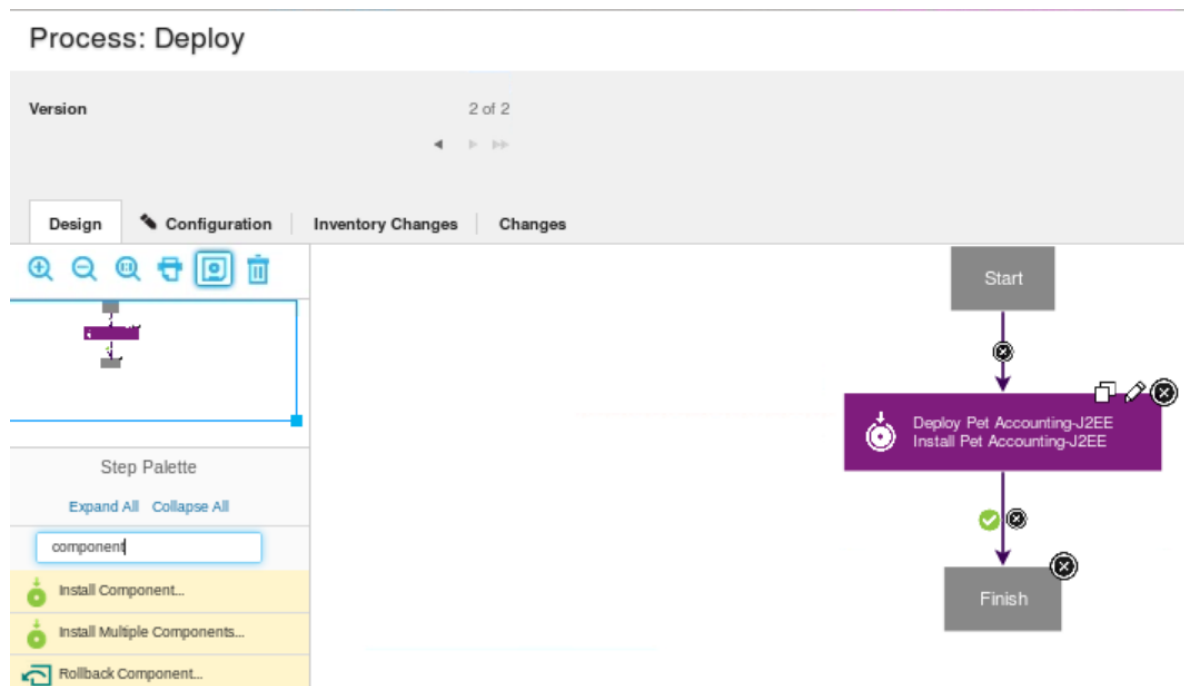


Ilustración 40 Proceso de una aplicación

Finalmente, la aplicación permite la creación de ambientes, estos permiten diferenciar entre ambientes de prueba que tienen diferentes rutas y se realizan diferentes despliegues que por ejemplo un ambiente de producción el cual solo se realizarían despliegues cuando esté completamente probado y sea seguro.

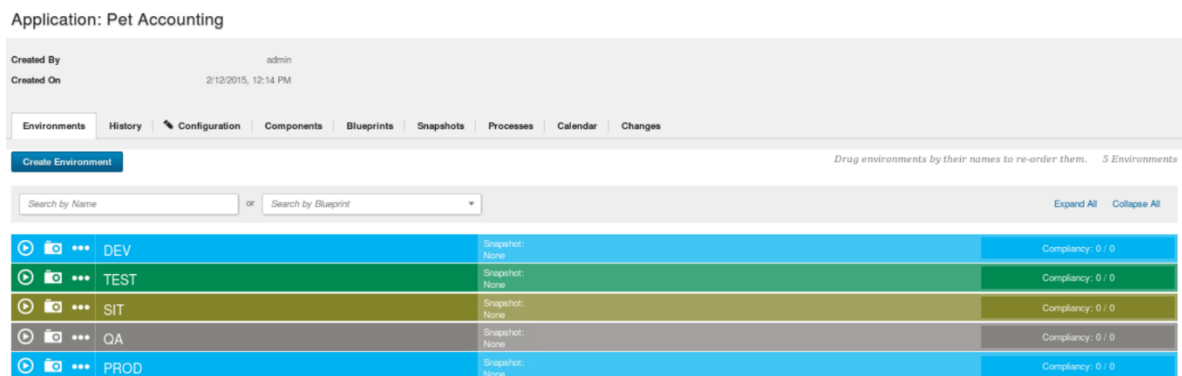


Ilustración 41 Ambientes de una aplicación

Los ambientes también permiten definir variables de ambiente las cuales pueden contener diferentes tipos de información, como lo es la ruta del servidor, o un puerto en específico el cual haga uso ese ambiente seleccionado entre muchas otras posibilidades, estas variables posteriormente pueden ser llamadas al momento de definir configuraciones o algún paso del proceso que las requiera, de esta forma evitando que en caso de ocurrir alguna modificación tener que cambiar este campo en cada proceso que lo requería sino que simplemente se modifica la variable de ambiente al valor nuevo deseado.

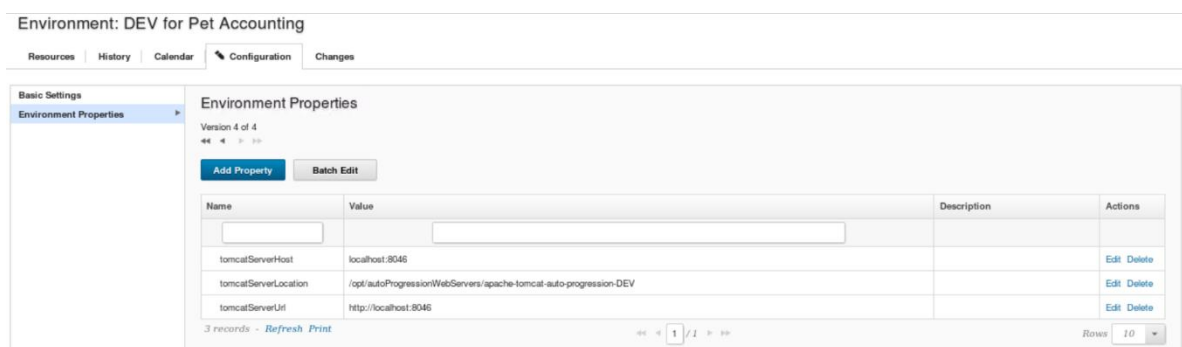


Ilustración 42 Variables de ambiente

Posteriormente a la configuración previamente realizada, se procede a solicitar el despliegue de una aplicación en un ambiente definido, este solicita las versiones de los componentes que se desean desplegar y el proceso de aplicación que se desea realizar.

Run Process on CERT-1



Only Changed Versions ☒

Process *

Select a snapshot, or choose versions for individual components.

Snapshot

Schedule Deployment? ☐

Description

Submit

Cancel

Ilustración 43 Pantalla inicial solicitud de despliegue

Component Versions



Select For All...

☐ Show only changed components ?

☐ Allow invalid versions ?

Component	Current Environment Inventory	Versions to Deploy
Pet Breeder Site-APP	1.0.2	1.0.126 x Add...
Pet Breeder Site-DB	1.0.2	2.0.1 x Add...

2 records

<< < 1 / 1 > >>

Rows 10

OK

Ilustración 44 Definición de versiones de componentes a desplegar

Run Process on CERT-1



Only Changed Versions ☒

Process *

Breeder Site - Deploy All

Select a snapshot, or choose versions for individual components.

Snapshot

Component Versions

Versions 2 selected ([Choose Versions](#))

Schedule Deployment? ☐

Description

Submit

Cancel

Ilustración 45 Pantalla solicitud de despliegue cargada

Posteriormente iniciara el proceso de despliegue en donde se puede observar el estado actual del despliegue, esto incluye el componente que se está desplegando, el proceso que se encuentra realizando y el paso actual que se encuentra realizando. Al finalizar se puede observar un reporte completo del proceso donde se puede observar el estado de todos los procesos y componentes que se desplegaron, indicando si este fue exitoso o si fallo en algún punto en específico, para esto la herramienta también permite observar los logs del proceso de despliegue en donde se analizar con mayor detalle los pasos realizados y los errores o resultados que se vieron en el servidor destino.

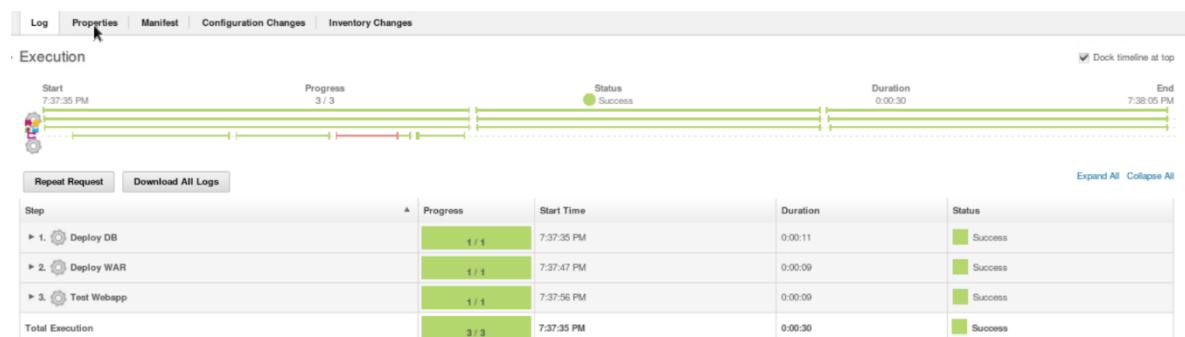


Ilustración 46 Ejecución del despliegue solicitado

▼ 1. Deploy DB	1 / 1	7:37:35 PM	0:00:11	Success
▼ Pet Breeder Site-DB	1 / 1	7:37:35 PM	0:00:11	Success
▼ Deploy (Pet Breeder Site-DB 2.0.1)		7:37:35 PM	0:00:11	Success
1. Download Artifacts		7:37:36 PM	0:00:04	Success
2. Get Component Short Name		7:37:40 PM	0:00:02	Success
3. Stop Application		7:37:43 PM	0:00:01	Failed
4. Create war file		7:37:43 PM	0:00:02	Success
5. Join		7:37:45 PM	0:00:00	Success
6. Copy war into Tomcat webapps directory		7:37:45 PM	0:00:01	Success

Ilustración 47 Resultado despliegue detallado con cada paso del proceso

Output Log

```

1 =====
2 plugin: Application Deployment for WebSphere, id: com.urbancode.air.plugin.WebSphere, version: 75
3 plugin command: '/opt/ibm-ucd/agent/opt/groovy-1.8.8/bin/groovy' '-cp' '/opt/ibm-ucd/agent/var/plugins/com.urbancode.air.plugin.WebSphere_75_90465f38
4 working directory: /opt/ibm-ucd/agent/var/work/CERT-1/Pet Breeder Site-DB
5 properties:
6 PLUGIN_INPUT_PROPS=/opt/ibm-ucd/agent/var/temp/logs7614942426340750139/input.props
7 PLUGIN_OUTPUT_PROPS=/opt/ibm-ucd/agent/var/temp/logs7614942426340750139/output.props
8 appEdition=
9 appName=
10 batch=
11 cell=
12 cluster=
13 commandPath=
14 contentType=
15 host=
16 node=
17 password=
18 port=
19 scriptFile=d4d06763-3b5c-4e0d-ab6b-76a4ea371968.py
20 server=
21 user=
22 environment:

```

Download Log

Ilustración 48 Log detallado de ejecución de paso del proceso

Al observar el ambiente también se indica el estado de los componentes actuales de la aplicación, la última versión desplegada satisfactoriamente y las versiones que no aún no se despliegan para el ambiente en específico.

CERT-1							
Snapshot: None				Compliance 2 / 5			
Component	Version	Date Created	Snapshot	Properties	Status	Compliance	Actions
Pet Breeder Site-APP	1.0.126	4/28/2019, 7:37 PM		Version 1	Active	Compliant (1/1)	View Request
Pet Breeder Site-DB	2.0.1	4/28/2019, 7:37 PM		Version 1	Active	Compliant (1/1)	View Request

Refresh Print

Ilustración 49 Ambiente con información de los componentes actuales desplegados

5.3.1.2 Desarrollo con Urbancode Deploy

Como parte del soporte es necesario realizar modificaciones a la herramienta según se presenten requerimientos de esta, durante el tiempo de practica se presentó una situación que dio origen a un requerimiento después de su respectivo análisis, posteriormente se realizó el respectivo desarrollo de este siguiendo los siguientes procedimientos.

5.3.1.3 Análisis del problema

Se presento una situación durante la ejecución de unos despliegues de un componente, en los cuales se informó por parte del cliente que al desplegar dos versiones diferentes la herramienta repitió el proceso dos veces con la misma versión del componente. Se procedió a realizar el análisis de los dos despliegues partiendo de lo mencionado por el cliente.

Se observo el primer despliegue que contenía el componente con la versión 1509, iniciado el 6/6/2019, 3:45 PM

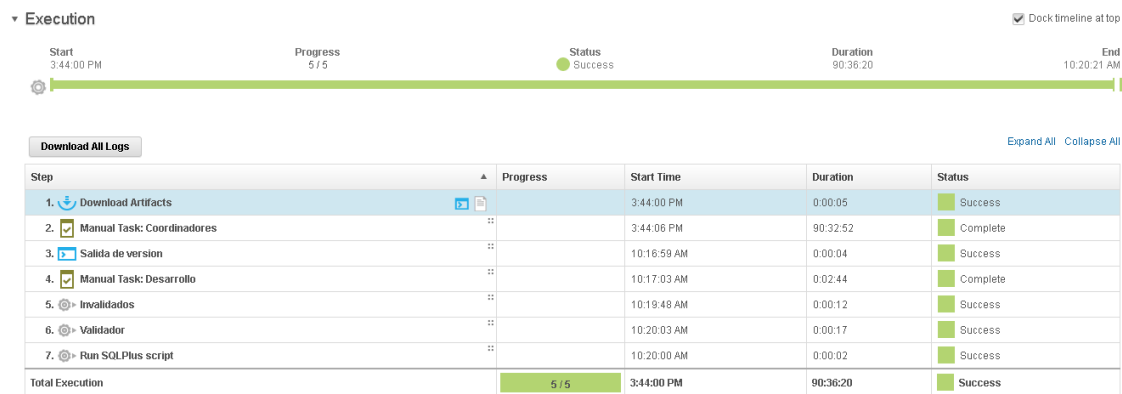


Ilustración 50 Despliegue versión 1509

Se observaron a su vez los artefactos o archivos que esta versión contiene:

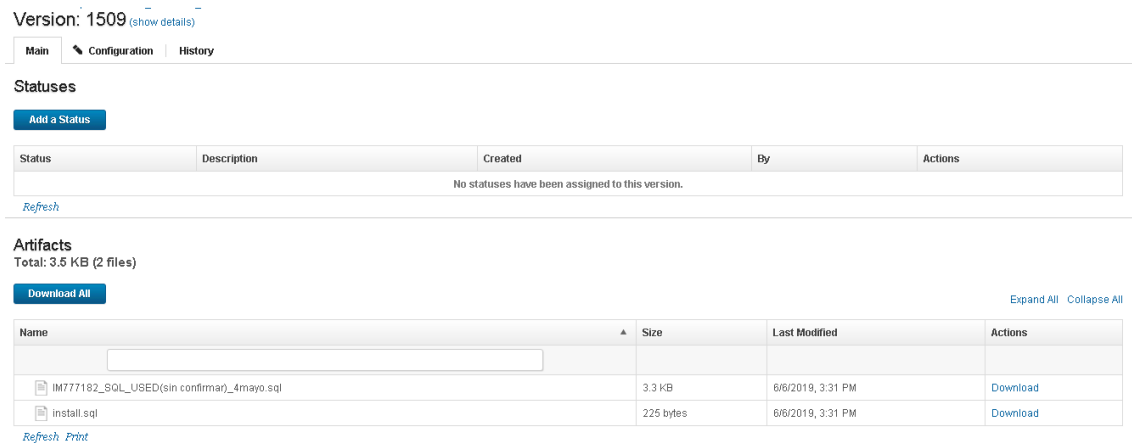


Ilustración 51 Artefactos versión 1509

Analizando el proceso de despliegue se observó que este realizo la descarga de los artefactos y quedo en espera de la validación requerida para dar continuación con el proceso, se demoró alrededor de 90 horas en realizar el manual task de coordinadores para la validación, fecha aprobación de la validación:



Por favor proceder.

by [REDACTED] on 6/10/2019, 10:16 AM

Ilustración 52 Aprobación manual task versión 1509

Por temas de seguridad se oculta el nombre del usuario encargado de realizar este proceso. Durante este periodo de tiempo el componente no se despliega esperando la validación.

Posteriormente los usuarios inician el proceso de despliegue de la versión 1520, iniciado el 6/7/2019, 5:29 PM

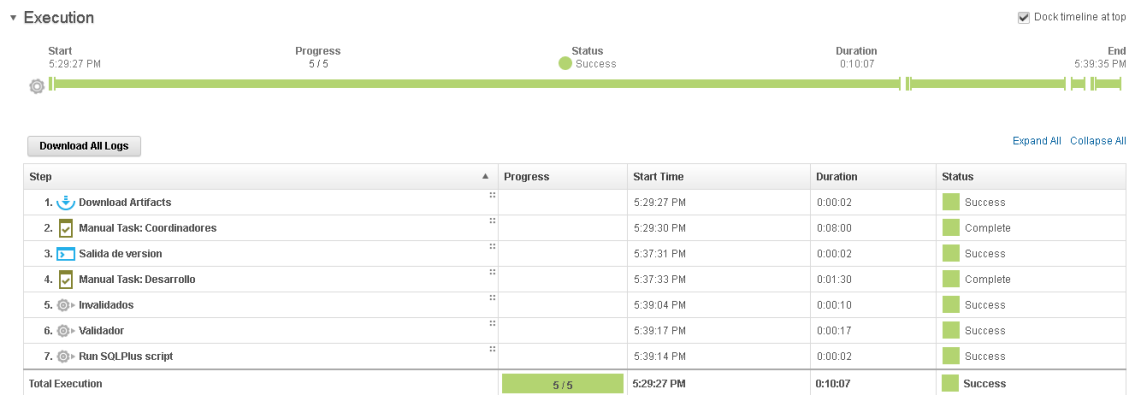


Ilustración 53 Despliegue versión 1520

Se observan los artefactos que contiene esta versión del componente:

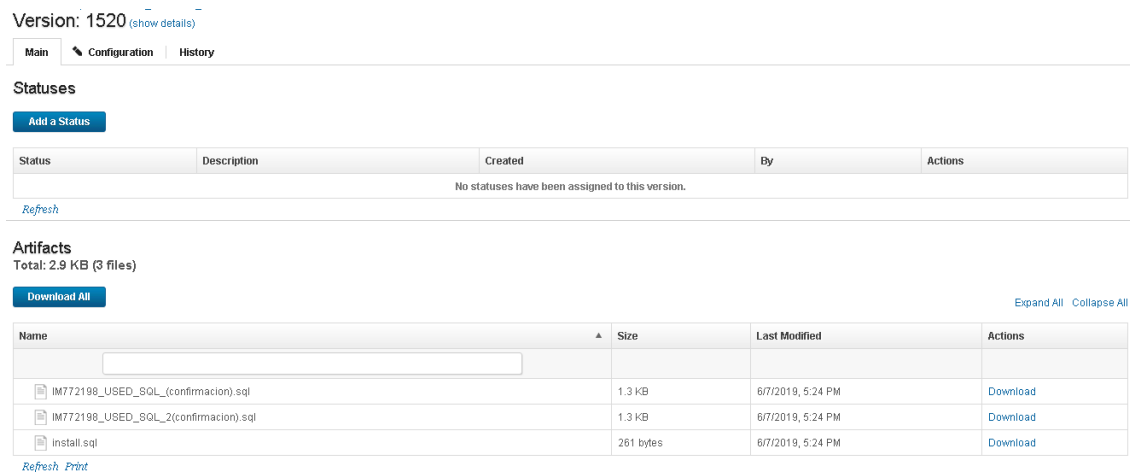


Ilustración 54 Artefactos versión 1520

Este realiza la descarga de sus artefactos (versión 1520), como parte del proceso al realizar la descarga de dichos artefactos modifica y reemplaza los artefactos previamente descargados en la versión 1509 en la ruta destino.

Se demora alrededor de 8 minutos en realizar el manual task de coordinadores para la validación y posteriormente dar continuidad al proceso de despliegue, fecha aprobación de la validación



Por favor proceder.
by [REDACTED] on 6/7/2019, 5:37 PM

Ilustración 55 Aprobación manual task versión 1520

Por lo que procede a realizar el despliegue de la versión 1520. (En este momento todavía no se despliega la versión 1509)

Partiendo de lo observado anteriormente, cuando finalmente se valida el despliegue de la versión 1509, el destino contiene ahora son los artefactos de la versión 1520, el proceso de despliegue ya había realizado previamente la descarga de los artefactos, por lo que inmediatamente procede a realizar el despliegue de los artefactos actuales que contiene la ruta del destino (Versión 1520).

Output Log

```
Working Directory /opt/IBM/ibm-ucd/agent6/var/work/BD_FlexCube_PRD
24 UD_DIALOGUE_ID=a61496e3-fd74-4d78-852d-ffc59feae8b
25 WE_ACTIVITY_ID=3af0e448-cc7d-436a-af3e-7654e6d05fa4
26 =====
27
28 command line: /bin/sh /tmp/shell_command_4773696316090890839.tmp
29 script content:
30 -----
31 ls -lt
32 more install.sql
33 -----
34 working directory: /opt/IBM/ibm-ucd/agent6/var/work/BD_FlexCube_PRD
35 =====
36 command output:
37 total 12
38 -rw-rw-r--. 1 ucdadm grpurban 1339 Jun  7 17:24 IM772198_USED_SQL_2(confirmacion).sql
39 -rw-rw-r--. 1 ucdadm grpurban 1341 Jun  7 17:24 IM772198_USED_SQL_(confirmacion).sql
40 -rw-rw-r--. 1 ucdadm grpurban 261 Jun  7 17:24 install.sql
41 ::::::::::::::
42 install.sql
43 ::::::::::::::
44 ALTER SESSION SET Current_Schema = FCJADM;
45 set define off
46 set feedback on
47 set termout on
48 set echo on
49 set linesize 32767
50 set pagesize 50000
51 set serveroutput on size 1000000
52 @"IM772198_USED_SQL_(confirmacion).sql"
53 @"IM772198_USED_SQL_2(confirmacion).sql"=====
54 command exit code: 0
55
```

Ilustración 56 Output log artefactos despliegue

Esto se logra evidenciar en el Output Log del proceso de despliegue de la versión 1509, el cual muestra que los artefactos utilizados para el despliegue son los de la versión 1520.

Finalmente se informa al cliente lo sucedido en detalle y se le indica que no fue un error de la herramienta, sino que el proceso funciona de esta manera en la actualidad, por lo que el cliente solicita modificar el proceso ya que requieren realizar el proceso de despliegue sin importar el orden y que la herramienta mantenga la versión hasta que se finalice el proceso de esta.

5.3.1.4 Solución propuesta

Como propuesta de solución se consideró incorporar la creación de una carpeta propia de la versión que se generara en la ruta del destino del despliegue, posterior a su creación los componentes de dicha versión se descargarían dentro de esta carpeta y se direccionaría el proceso de despliegue a la ruta dentro de la carpeta de la versión, con esto todas las versiones tendrían sus propias carpetas las cuales se eliminarían únicamente al finalizar el proceso de despliegue de la versión, corrigiendo de esta forma el hecho de que todas las versiones se ubicaran en la misma ruta y por ende procedieran a sobrescribirse.

Se informo la solución propuesta al cliente, posterior a recibir la aprobación de este se procedió a iniciar el desarrollo de la solución.

5.3.1.5 Desarrollo de la solución

Inicialmente se trabajó en la aplicación de pruebas de IBM, en este se realizaron las modificaciones respectivas y se realizaron las pruebas necesarias para validar el correcto funcionamiento del proceso de despliegue propuesto. Después de las modificaciones el nuevo proceso de despliegue fue el siguiente:

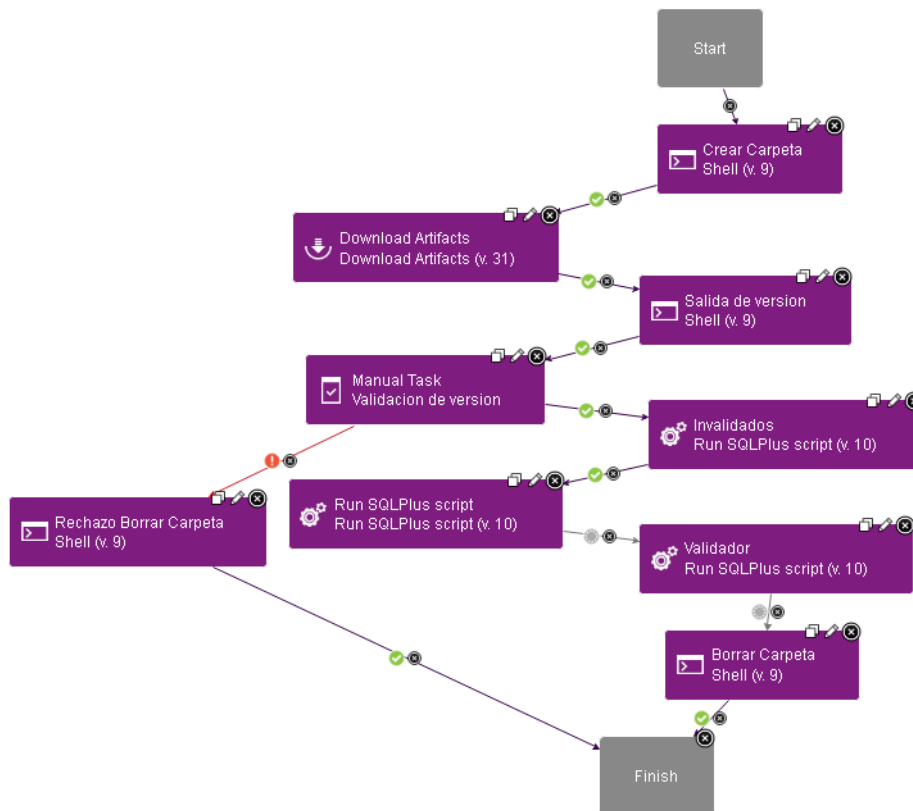


Ilustración 57 Proceso de despliegue modificado

Las modificaciones consistieron en agregar los pasos “Crear Carpeta”, “Rechazo Borrar Carpeta” y “Borrar Carpeta”, adicionalmente se modificaron las rutas de los demás pasos para que realizaran su proceso habitual dentro de la respectiva carpeta de la versión.

Edit Properties



Name *	Crear Carpeta
Directory Offset *	.
Interpreter	
Shell Script *	1 <code>mkdir "Version \${p:version.name}"</code>
Daemon	☐
Output File	
Working Directory	
Post Processing Script	Step Default New
Precondition	1
Use Impersonation	☐
Show Hidden Properties	☐

Ilustración 58 Paso Crear Carpeta

El paso “Crear Carpeta” realiza la creación de la carpeta por medio de la ejecución de un Shell script.

Edit Script

Go to line

1 `mkdir "Version ${p:version.name}"`

Ilustración 59 Script Crear Carpeta

Este utiliza el comando “mkdir” para crear la carpeta, como nombre se asigna la palabra “Version” y adicionalmente haciendo uso de las variables propias de la herramienta se trae la variable que contiene el nombre de la versión actual del componente (\${p:versión.name}).

Edit Properties

Name *

Download Artifacts

Directory Offset *

Version \${p:versión.name}

Artifact Directory Offset

Includes *

1

Excludes

1

Sync Mode

Sync

Full Verification

☒

Set File Execute Bits

☐

Verify File Integrity

☐

Charset

Working Directory

Post Processing Script

Step Default

New

Precondition

1

Use Impersonation

☐

Show Hidden Properties

☐

OK

Cancel

Ilustración 60 Paso Download Artifacts

El paso de descargar los artefactos o archivos de la versión se le agrego un Offset a la ruta para que descargara directamente en la carpeta previamente creada.

Edit Properties



Name *	Salida de version
Directory Offset *	.
Interpreter	
Shell Script *	<pre>1 cd ./"Version \${p:version.name}" 2 ls -lt 3 more install.sql</pre>
Daemon	☐
Output File	
Working Directory	
Post Processing Script	Step Default ▼ New
Precondition	1
Use Impersonation	☐
Show Hidden Properties	☐

OK Cancel

Ilustración 61 Paso Salida de versión

De igual forma al paso de “Salida de versión” se modificó para que accediera a la carpeta.

Edit Script

Save ↶ ↷ ✂ 📄 📄 Go to line 🔍 🔄 #

```
1 cd ./"Version ${p:version.name}"
2 ls -lt
3 more install.sql
```

Ilustración 62 Script Salida de versión

Para acceder a la carpeta se agregó el comando “cd” y la ruta de la carpeta.

Edit Properties

Name *
Validation de version

Notification Template *
TaskCreated

Add Property

Properties

Name	Label	Pattern	Required	Default Value	Description	Actions
No properties have been defined.						

Refresh
Print

Who can approve this task
Role Member By Application

Autorizador

for

Standard Application

Remove

Add Role...

OK

Cancel

Ilustración 63 Paso Validacion de versión

El paso “Validacion de version” se encarga únicamente de esperar la aprobación para seguir con el proceso de despliegue por lo que no fue necesario realizar ninguna modificación.

Edit Properties

Name *
Rechazo Borrar Carpeta

Directory Offset *
.

Interpreter

Shell Script *

1
rm -rf "version \${p.version.name}"

Daemon
☐

Output File

Working Directory

Step Default

New

Precondition

1

Use Impersonation
☐

Show Hidden Properties
☐

OK

Cancel

Ilustración 64 Paso Rechazo Borrar Carpeta

El paso “Rechazo Borrar Carpeta” es el encargado de borrar la carpeta y sus contenidos a través de un Shell script en caso de que no se apruebe realizar el despliegue.

Edit Script

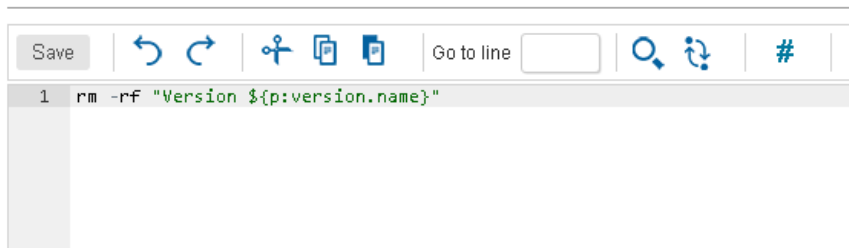


Ilustración 65 Script Rechazo Borrar Carpeta

El comando para borrar los archivos es “rm -rf” seguido por el nombre de la carpeta la cual es la que se desea borrar.

Los pasos “Invalidados”, “Run SQLPlus script” y “Validador” se encargan de correr cada uno un script estándar para todos los despliegues por lo que no es necesario modificarlos.

Edit Properties

The screenshot shows the 'Edit Properties' dialog box for a step named 'Borrar Carpeta'. The fields are as follows:

- Name ***: Borrar Carpeta
- Directory Offset ***: .
- Interpreter**: (empty)
- Shell Script ***: 1 rm -rf "Version \${p:version.name}"
- Daemon**: ☐
- Output File**: (empty)
- Working Directory**: (empty)
- Post Processing Script**: Step Default (dropdown menu) with a 'New' button below it.
- Precondition**: 1
- Use Impersonation**: ☐
- Show Hidden Properties**: ☐

At the bottom right are 'OK' and 'Cancel' buttons.

Ilustración 66 Paso Borrar Carpeta

El paso “Borrar Carpeta” cumple la misma función que “Rechazo Borrar Carpeta” solo que este paso se corre al momento de finalizar el despliegue de manera natural.

Edit Script

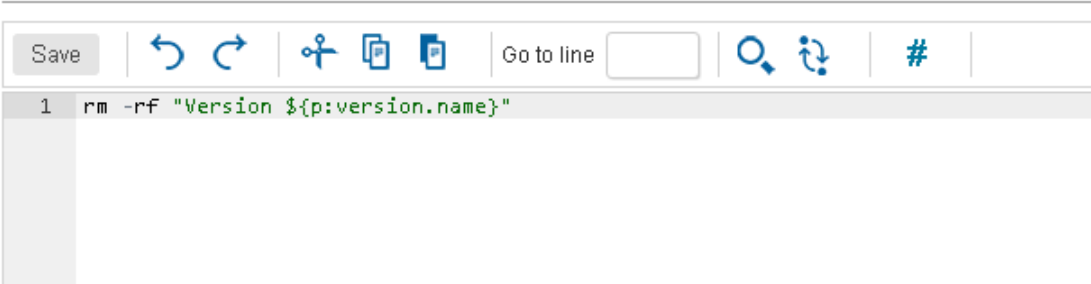


Ilustración 67 Script Borrar Carpeta

Nuevamente se utiliza el comando para borrar los archivos “rm -rf” seguido por el nombre de la carpeta la cual es la que se desea borrar.

5.3.1.6 Pruebas y Entrega

Posterior a validar que no se encuentran errores en el nuevo proceso, se solicitan dos versiones de prueba para replicar la situación que dio origen al problema y validar su corrección, por lo que se crean la versión 1547 y 1549.

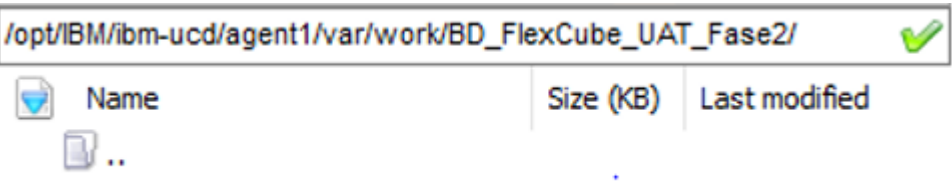


Ilustración 68 Ruta destino del despliegue

Se valida que la ruta del destino se encuentra vacía para dar inicio al despliegue.

Deployment of Component: BD_FlexCube_UAT_Fase2 [\(show details\)](#)

Log Properties

Execution

Pause Cancel Download All Logs Expand All Collapse All

Step	Progress	Start Time	Duration	Status
1. Crear Carpeta	100%	3:23:55 PM	0:00:02	Success
2. Download Artifacts	100%	3:23:58 PM	0:00:03	Success
3. Salida de version	100%	3:24:01 PM	0:00:02	Success
4. Manual Task: Validacion de version	Respond	3:24:04 PM	0:00:00	Open
5. Invalidados				Not Started
6. Run SQLPlus script				Not Started
7. Error Borrar Carpeta				Not Started
8. Validador				Not Started
9. Borrar Carpeta				Not Started
Total Execution	3 / 8	3:23:55 PM	0:00:09	Running

Ilustración 69 Inicio despliegue versión 1547

Se deja el proceso en el paso de esperar la validación para seguir el despliegue.

/opt/IBM/ibm-ucd/agent1/var/work/BD_FlexCube_UAT_Fase2/ 		
 Name	Size (KB)	Last modified
 ..		
 Version 1547	4	2019-06-13 15:24

Ilustración 70 Creación carpeta versión en la ruta

Se observa mientras que en la ruta efectivamente se creó la carpeta de la versión y contiene los archivos de prueba.

Deployment of Component: BD_FlexCube_UAT_Fase2 [\(show details\)](#)

Log

Properties

Execution

Pause

Cancel

Download All Logs

Expand All

Collapse All

Step	Progress	Start Time	Duration	Status
1.  Crear Carpeta	::	3:25:37 PM	0:00:05	Success
2.  Download Artifacts	::	3:25:43 PM	0:00:07	Success
3.  Salida de version	::	3:25:50 PM	0:00:05	Success
4.  Manual Task: Validacion de version	Respond	3:25:55 PM	0:00:06	Open
5.  Invalidados				Not Started
6.  Run SQLPlus script				Not Started
7.  Error Borrar Carpeta				Not Started
8.  Validador				Not Started
9.  Borrar Carpeta				Not Started
Total Execution	3 / 8	3:25:37 PM	0:00:24	Running

Ilustración 71 Inicio despliegue versión 1549

Se inicia el despliegue de la siguiente versión y se deja de igual forma en espera por aprobación.

/opt/IBM/ibm-ucd/agent1/var/work/BD_FlexCube_UAT_Fase2/ 		
 Name	Size (KB)	Last modified
 ..		
 Version 1547	4	2019-06-13 15:24
 Version 1549	4	2019-06-13 15:25

Ilustración 72 Creación carpeta de dos versiones en la ruta

Se observa que efectivamente creo otra carpeta para la versión con sus propios artefactos, por lo que no modifica nada de la versión previa.

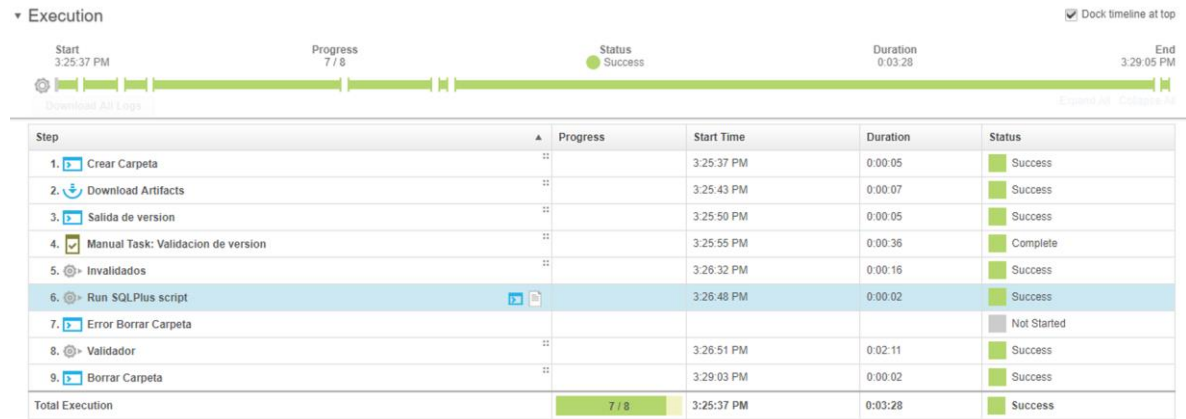


Ilustración 73 Finaliza despliegue versión 1549

Se aprueba le versión 1549 y esta finaliza el proceso de despliegue satisfactoriamente.

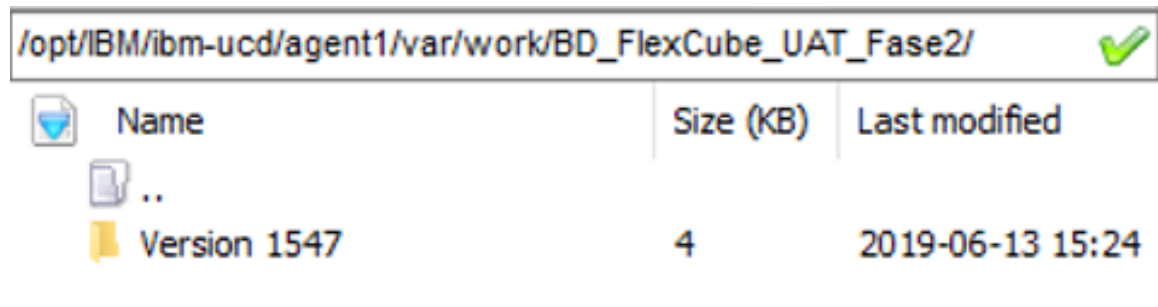


Ilustración 74 Eliminación de carpeta al finalizar despliegue

Se observa que efectivamente al terminar el despliegue se elimina únicamente la carpeta de la versión correspondiente.

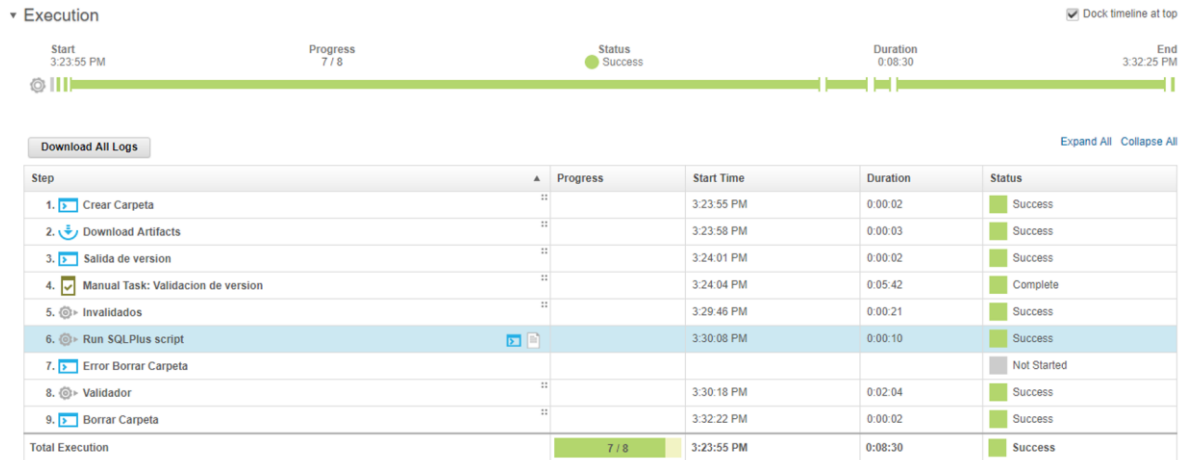


Ilustración 75 Finalización despliegue versión 1547

Finalmente se aprueba la versión 1547 y termina su proceso de despliegue de manera correcta.

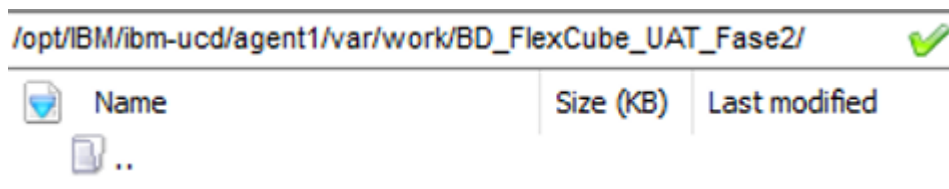


Ilustración 76 Eliminación de carpeta versión 1547

Por último, se confirma que la carpeta de la versión 1547 también se eliminó correctamente en la ruta del destino.

Se entregan los resultados de estas pruebas al cliente, los cuales dan el visto bueno para replicar el proceso de pruebas en producción y de esta forma dar cierre al caso.

5.3.2 Rational ClearQuest

Esta herramienta funciona como una base de datos modificable para cumplir con los requerimientos o necesidades del cliente, esta permite crear y mantener un control de los flujos de trabajo de los diferentes desarrollos, aplicaciones, actividades y sistemas de producción de la organización entre otros que esta requiera, ofreciendo mayor visibilidad y control sobre el ciclo de vida del desarrollo de software. Facilitando a su vez el trabajo de auditoria del mismo. [7]

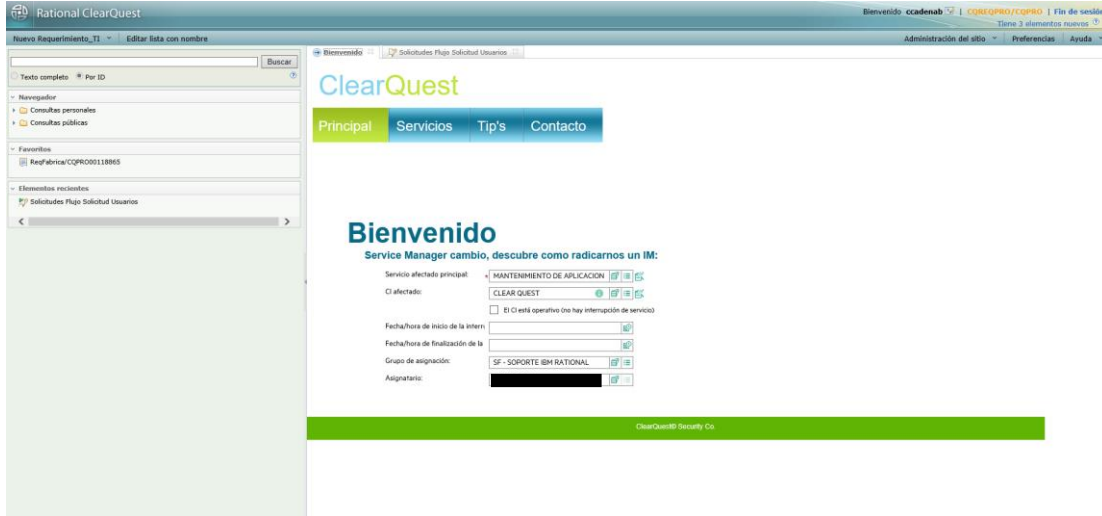


Ilustración 77 Pantalla de bienvenida ClearQuest

Esta herramienta define principalmente flujos y registros, los cuales se diferencian en el hecho de que un flujo puede cambiar de estado de acuerdo con la transición que tenga definida para dicho flujo mientras que los registros no pueden cambiar de estado, a su vez, un flujo puede contener en él muchos registros e incluso otros flujos, pero un registro únicamente puede contener más registros dentro de él.

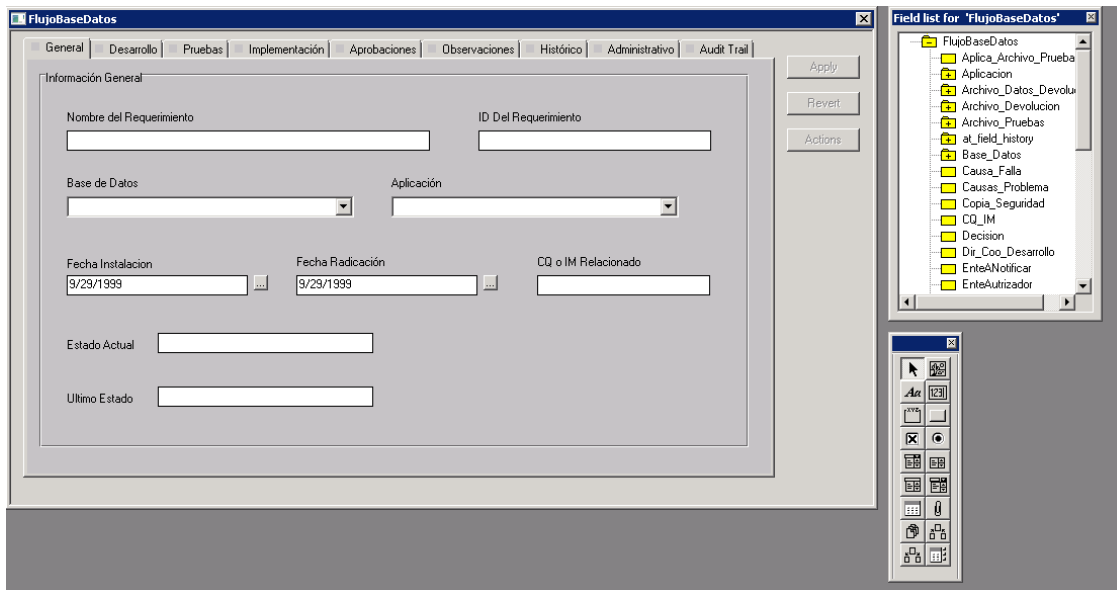


Ilustración 78 Ejemplo Flujo

The screenshot shows a software application window titled "Registro". It has three tabs: "Registro Base de Datos", "Observaciones", and "Histórico". The "Registro Base de Datos" tab is selected, displaying a form with the following fields:

- ID Del Requerimiento**: A text input field.
- Tipo de Archivo**: A text input field.
- Código de Revisión**: A text input field.
- Descripción**: A dropdown menu.
- URL**: A large text area.
- URL Instructivo de Instalación**: A large text area.

To the right of the main form is a panel titled "Field list for 'Registro_Desarrollo'". It contains a list of fields with checkboxes:

- ☐ Registro_Desarrollo
- ☐ CodigoRevision
- ☐ Descripcion
- ☐ Fecha_Radicacion
- ☒ FlujoBaseDatos
- ☐ history
- ☐ locked_by
- ☐ Observacion
- ☐ ratl_keysite
- ☐ ratl_mastership
- ☐ record_type
- ☐ Tipo_Archivo
- ☐ URL
- ☐ URL_Instructivo
- ☐ Usuario_Radicacion

Below the field list is a toolbar with various icons for editing and viewing data.

Ilustración 79 Ejemplo registro

Los estados de los flujos permiten mantener informados al equipo de trabajo sobre el estado actual en el que se encuentra el desarrollo de una actividad o proceso entre otros, solo basta con acceder al flujo para validar el estado actual y de esta forma identificar cual es la siguiente acción que se requiere para dar continuidad al flujo y a su vez quien es el encargado de realizar dicha acción, por ejemplo, la creación de un flujo de base de datos el cual se utiliza para modificar, crear o realizar alguna acción sobre una base de datos del banco. Primero se realiza la creación del flujo, esta marca los campos mandatorios con rojo para indicarle al usuario cual es la información que debe agregar en este estado, estos campos contienen principalmente los datos de que se realizara y sobre qué base de datos.

Crear FlujoBaseDatos CQPRU00008091

General Desarrollo Pruebas Implementación Aprobaciones Observaciones Histórico Administrativo Audit Trail

Información General

Nombre del Requerimiento ID Del Requerimiento
 Base de Datos Aplicación
 Fecha Instalación Fecha Radicación CQ o IM Relacionado
 Estado Actual 10_Radicado
 Ultimo Estado

OK Cancel Values

Ilustración 80 Ejemplo campos obligatorios

En caso de intentar guardar el flujo o registro sin tener todos los campos mandatorios la herramienta le informara a través de una pantalla de error indicando cual campo debe ser completado.

Crear FlujoBaseDatos CQPRU00008091

General Desarrollo Pruebas Implementación Aprobaciones Observaciones Histórico Administrativo Audit Trail

Información General

Nombre del Requerimiento
 Base de Datos
 Fecha Instalación
 Estado Actual 10_Radicado
 Ultimo Estado

IBM Rational ClearQuest

These fields have invalid values:

- The field "Aplicación" labelled "Aplicación" on tab "General" is mandatory; a value must be specified.
- The field "Base_Datos" labelled "Base de Datos" on tab "General" is mandatory; a value must be specified.
- The field "Causa_Falla" labelled "Causa de Falla" on tab "Implementación" is mandatory; a value must be specified.
- The field "Causas_Problema" labelled "Causas del Problema" on tab "Implementación" is mandatory; a value must be specified.
- The field "Copia_Seguridad" labelled "¿Existe copia de seguridad?" on tab "Implementación" is mandatory; a value must be specified.
- The field "CQ_IM" labelled "CQ o IM Relacionado" on tab "General" is mandatory; a value must be specified.
- The field "Fecha_Instalacion" labelled "Fecha Instalacion" on tab "General" is mandatory; a value must be specified.
- The field "Impacto" labelled "Impacto" on tab "Implementación" is mandatory; a value must be specified.
- The field "Nombre_Requerimiento" labelled "Nombre del Requerimiento" on tab "General" is mandatory; a value must be specified.
- The field "Razones_Tarea" labelled "Razones para Realizar la Tarea" on tab "Implementación" is mandatory; a value must be specified.
- The field "Registro_BD" labelled "Registro de Base de Datos" on tab "Desarrollo" is mandatory; a value must be specified.
- The field "Riesgo_Imp" labelled "Riesgo" on tab "Implementación" is mandatory; a value must be specified.

OK

Ilustración 81 Ejemplo mensaje de error para campos faltantes

Posteriormente para este flujo se procede a ejecutar la acción de solicitar aprobación, esta indica nuevamente los campos requeridos para ejecutar dicha acción, en este caso requiere asignar al director encargado de realizar la aprobación para la acción que se desea ejecutar en la base de datos.

Una vez asignado se le enviara un correo directo al usuario indicando que se solicita su aprobación para dar continuidad al flujo, dado que únicamente este usuario es el encargado de aprobar para seguir con el siguiente estado.

Finalmente, al dar la aprobación se procederá a realizar la acción solicitada sobre la base de datos llenando a su vez los campos que requiera el flujo para dar cierre.

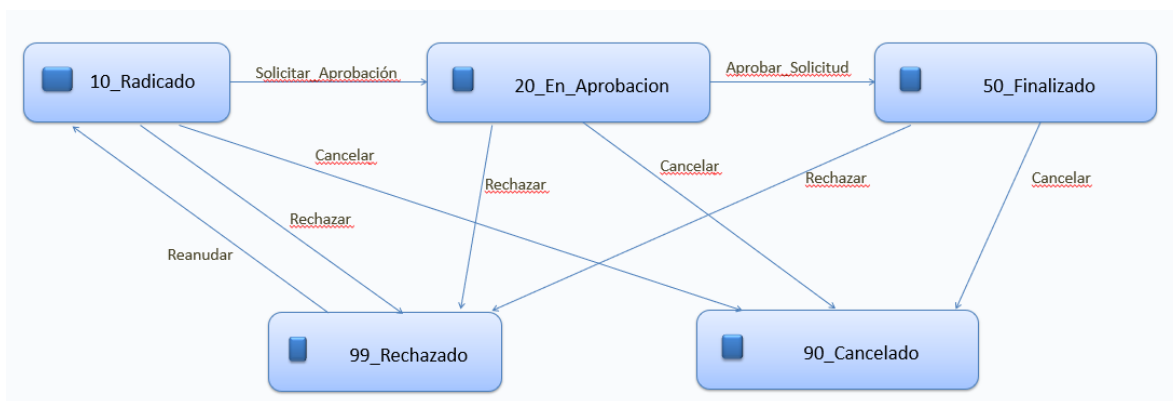


Ilustración 82 Diagrama de transición del flujo

Este proceso se realizará de igual manera con los demás estados que contenga los otros flujos en específico, solicitando la ejecución de acciones y modificaciones según se requiera hasta llegar a su estado final, el cual dará cierre al flujo y con él al procedimiento realizado en el banco.

Ya que la herramienta principalmente se utiliza para mantener un control de los procesos del banco, el flujo también cuenta con un historial o Audit Trail el cual indica todo el ciclo de vida del flujo, permitiendo de esta manera observar quien ha realizado modificaciones sobre el flujo, como también los usuarios que le dieron continuidad al avanzar cada estado hasta su cierre. Todo esto con el fin de agilizar los procesos internos de la organización y su vez mantener un control de auditoria de los mismos.

5.3.3 Soporte Rational ClearQuest

En la herramienta se realizan diferentes acciones de soporte al cliente, de estas las más importantes realizadas durante el proceso fueron las siguientes.

5.3.3.1 Creación Usuarios y modificación permisos

Dentro de las acciones de soporte de la herramienta se encuentra la tarea de crear los usuarios solicitados para la misma, esto se realizaba a través de un flujo de solicitud de usuarios el cual contiene toda la información necesaria para realizar la creación.

The screenshot shows the 'FlujoSolicitudUsuarios' application window. The 'Solicitud' tab is active, displaying a form with the following fields:

- Solicitud**
 - Ambiente: [Dropdown]
 - Tipo de Usuario: [Dropdown]
 - Tipo de Solicitud: [Dropdown]
 - Valida: [Text]
 - Aplicación: [Dropdown]
 - Sistema Operativo: [Dropdown]
 - Base de Datos: [Dropdown]
 - Manejo de Datos Enmascarados y/o Transformados: [Dropdown]
 - Fecha de Solicitud: 9/29/1999 9:39:00 AM [Text]
 - Modifica/Altera Estados Financieros: [Dropdown]
 - Estado Actual: [Text]
 - Número de la solicitud: [Text]
 - Último Estado: [Text]
- Solicitud Usuarios**
 - Lista de Usuarios Solicitados**

dbid	Nombre	Estado de la Solicitud	E1	E2	E3	E4
Right ...						

Buttons: Remover, Nuevo, Agregar
 - Lista de Usuarios Rechazados**

dbid	Nombre	Estado de la Solicitud	E1	E2	E3	E4	E5
Right ...							

Ilustración 83 Flujo solicitud de usuarios

Como mínimo para realizar esta acción se requiere el nombre completo del usuario, su usuario de banco, el correo del banco, el rol que este va a cumplir y la unidad o división a la que pertenece, en caso de que el registro del usuario no contenga toda la información es necesario regresar el flujo al estado de solicitar información e indicar específicamente el dato faltante, posteriormente se debe esperar a que el usuario complete el campo y mueva nuevamente el flujo de estado para continuar con la petición. Una vez se validan todos los datos se procede a realizar la solicitud en el administrador de usuarios de ClearQuest.

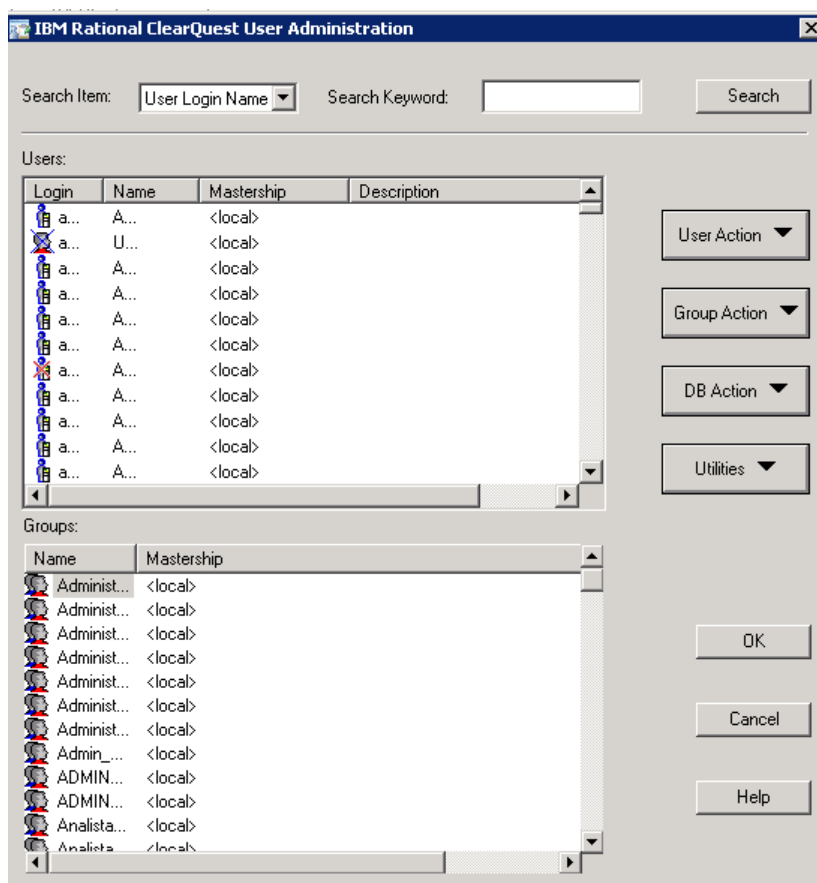


Ilustración 84 Pantalla administrador de usuarios

El administrador de usuarios además de permitir la creación de nuevos usuarios es también el que permite modificar los permisos o grupos a los que pertenecen los usuarios actuales, estos se encuentran a partir del campo “Search Keyword” y ingresar el usuario del banco que se requiere modificar.

Para la creación de un nuevo usuario se ingresa al botón “User Action” en la acción “Add user”

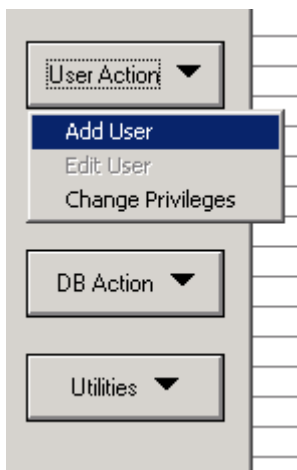


Ilustración 85 Acción agregar usuario

Esto abre la ventana para ingresar los datos del usuario.

 A screenshot of the 'Add User' dialog box. The dialog has a title bar 'Add User' with a close button. It contains several input fields: 'Login:', 'Password:', 'Confirm Password:', 'Name:', 'E-mail:', 'Phone:', 'Mastership:' (with a dropdown menu showing '<local>'), and 'Description:'. There is a 'Groups:' section with a list of checkboxes and group names: 'DIV_OPERACIONES', 'DIV_OPE_CARTERA_INT_TESO', 'DIV_ANALISIS_Y_PRESUPUEST', 'DIV_RECURSOS_ADMINISTRAT', and 'DIV CONTABILIDAD'. Below this is a checkbox for 'LDAP Authenticated' and an 'LDAP Login:' field. On the right side, there is a 'Privileges' section with a list of checkboxes: 'Active User' (checked), 'Dynamic List Administrator', 'Public Folder Administrator', 'SQL Editor', 'User Administrator', 'Schema Designer', 'All Users/Groups Visible' (checked), 'Security Administrator', and 'Super User'. Below the privileges is a 'Clear All' button. There is also a 'Subscribe' section with two radio buttons: 'All existing and future databases' and 'Select databases' (selected). Below this is a list of checkboxes for databases: 'CQPRO', 'SwF_P', 'SwF_T', and 'CQPRU'. Below the databases is a 'Clear All' button and a 'Check All' button. At the bottom of the dialog are five buttons: 'Add User', 'Clear All', 'OK', 'Cancel', and 'Help'.

Ilustración 86 Ventana de creación de usuarios

Todos los usuarios del banco se autentican a través del LDAP por lo que se selecciona el cuadro “LDAP Authenticated” y se ingresan los datos previamente mencionados, por último, se selecciona “Add User” para guardar el registro.

Para que se refleje el cambio realizado se deben actualizar las bases de datos del banco con el nuevo usuario, para esto se accede al botón “DB Action” y la acción “Upgrade”.

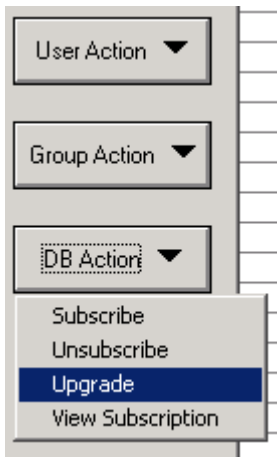


Ilustración 87 Acción actualizar bases de datos

Esta muestra los esquemas actuales de las bases de datos, dado que se requiere actualizar en todos, se dejan todos los campos como están y se selecciona OK.

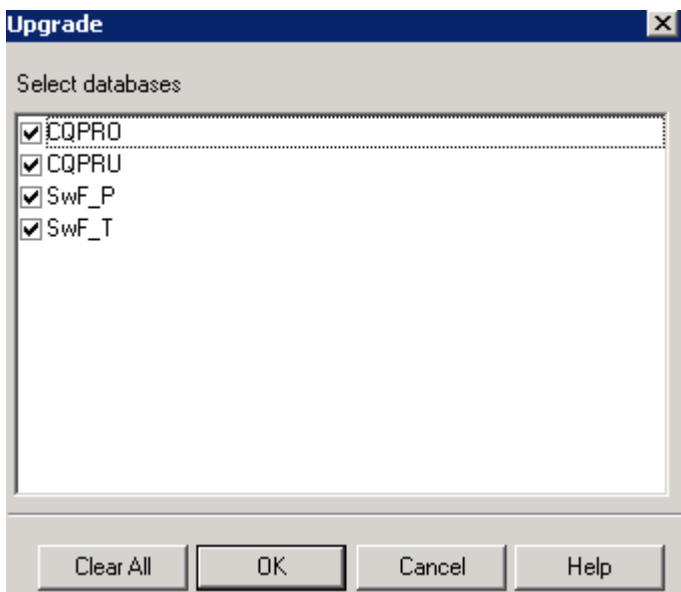


Ilustración 88 Selección de bases de datos a actualizar

Una vez actualizadas las bases de datos se procede a cerrar el flujo de la solicitud del

usuario, el cual indica que ya se realizó la creación del usuario y envía un correo al mismo mencionando el proceso para iniciar sesión en la herramienta.

5.3.3.2 Solución de incidentes – IM

Dentro de las labores de soporte de la herramienta se encuentra también la solución de Incidentes, estos se crean a partir del flujo de incidentes en el cual los usuarios solicitan la corrección o colaboración frente a diferentes situaciones del banco dentro de los cuales se encuentran los problemas de la herramienta, estas pueden incluir cambios de permisos en la herramienta, ayudar con la continuidad del algún flujo, fallos generales de la herramienta, modificaciones a algún flujo o registro, como por ejemplo el cambio del líder de un requerimiento, o los administradores de los ambientes del alguna aplicación entre otras. Se pueden considerar como el medio por el cual los usuarios solicitan soporte directo de la herramienta en sus diferentes funciones.

Cola: Vista:

ID de incidente	Estado	Estado de alerta	Grupo de asigna...	Asignatario	Título	Fecha/hora de a...	Fecha/hora de a...	Emisor de la pet...	Prioridad	CI afectado	Fecha/hora de r...
Grupo de asignación: CAMBIOS - DS - SOLUCIONES FUNCIONALES (2 Elementos)											
Grupo de asignación: SF - SOPORTE IBM RATIONAL (7 Elementos)											
Asignatario: [REDACTED] (6 Elementos)											
☐ IM786504	Pendiente de clien...	open	SF - SOPORTE IB...	CCADENAB	Cargue masivo us...	18/07/19 15:50:32	24/07/19 11:13:56	[REDACTED]	4 - Baja		
☐ IM786509	Pendiente de clien...	open	SF - SOPORTE IB...	CCADENAB	Cargue masivo us...	18/07/19 15:55:14	24/07/19 11:18:05	[REDACTED]	4 - Baja		
☐ IM786512	Pendiente de clien...	open	SF - SOPORTE IB...	CCADENAB	Cargue masivo us...	18/07/19 15:57:43	24/07/19 11:21:14	[REDACTED]	4 - Baja		
☐ IM786578	Categorizado	open	SF - SOPORTE IB...	CCADENAB	JREYESM - Solicit...	19/07/19 11:19:29	24/07/19 09:41:04	[REDACTED]	4 - Baja		
☒ IM787317	Categorizado	open	SF - SOPORTE IB...	CCADENAB	FALLA CLEAR QU...	23/07/19 10:54:42	23/07/19 11:42:15	[REDACTED]	4 - Baja		
☐ IM787686	Pendiente de clien...	open	SF - SOPORTE IB...	CCADENAB	CAMBIO ANALIST...	24/07/19 11:48:57	24/07/19 16:09:08	[REDACTED]	4 - Baja		
Asignatario: NULL (1 Elemento)											

Ilustración 89 Lista de IMs asignados

5.3.3.3 Cargue masivos

Dentro de los incidentes se encuentran las solicitudes de cargues masivos, estos se realizan cuando el cliente requiere crear un flujo de solicitud de usuarios que contenga como mínimo más de 20 registros. Estos registros son enviados en un formato de documento Excel que contiene todos los campos necesarios para realizar su carga, adicionalmente, dado que los usuarios en un flujo normal requieren la aprobación de un director y un subgerente del banco, para este cargue se requiere que ambos personajes den previamente su visto bueno en un mensaje de correo el cual se utiliza para validar el cargue.

Una vez se validen los campos, se procede a ejecutar un macro incluido en el documento Excel, este lo almacenara en un archivo de texto el cual se utiliza para realizar la importación en la herramienta.

Dentro de la herramienta se encuentran almacenadas consultas para traer los flujos y registros que se están creando a partir de los archivos, estas son de utilidad para identificar específicamente cual fue el flujo creado y añadirle a este los registros correspondientes.

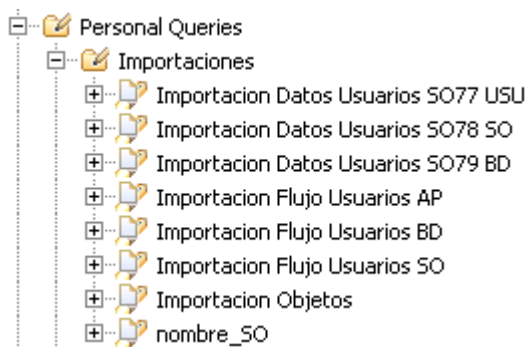


Ilustración 90 Consultas de importaciones

Para realizar la importación se accede a la opción “Import” dentro de la herramienta y adicionalmente se ingresa a “Records”.

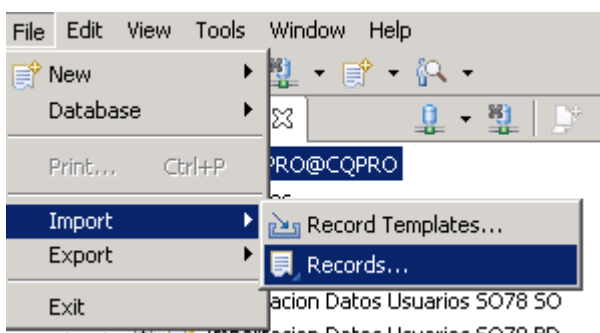


Ilustración 91 Acción de importación

Este abrirá una ventana la cual ira indicando la información que necesita para realizar la acción, similar al proceso de instalación de una herramienta.

The screenshot shows a Windows-style dialog box titled "ClearQuest Import". Below the title bar, the text "Select record type and data type" is displayed, followed by a smaller instruction: "(ccadenab,CQREQPRO@CQPRO) Please provide the information regarding the data you will be importing." The dialog is divided into three steps:

- Step 1:** "Please select Record Type to which you will be importing." It features a dropdown menu labeled "Record Type:" with "DatosUsuario" selected.
- Step 2:** "Please specify the type of data you will be importing. Select all that apply." It contains three checkboxes: "Record" (checked), "History" (unchecked), and "Attachments" (unchecked).
- Step 3:** It contains a single checkbox labeled "Importing updates to existing data records in your ClearQuest user database", which is unchecked.

At the bottom of the dialog, there are four buttons: "< Back", "Next >", "Finish", and "Cancel".

Ilustración 92 Importación para registros de DatosUsuario

Dentro de estos parámetros lo más importante es indicar el "Record Type" que para el caso de los registros es "DatosUsuario", de lo contrario para el flujo el tipo es "FlujoSolicitudUsuarios".

The screenshot shows a Windows-style dialog box titled "ClearQuest Import". The main heading is "Select record type and data type". Below this, a smaller text line reads "(ccadenab,CQREQPRO@CQPRO) Please provide the information regarding the data you will be importing." In the top right corner, there is a small icon of a calendar and a globe. The dialog is divided into three sections: "Step 1" with the instruction "Please select Record Type to which you will be importing." and a dropdown menu showing "FlujoSolicitudUsuarios"; "Step 2" with the instruction "Please specify the type of data you will be importing. Select all that apply." and three checkboxes: "Record" (checked), "History", and "Attachments"; and "Step 3" with a single checkbox "Importing updates to existing data records in your ClearQuest user database" which is unchecked. At the bottom, there are four buttons: "< Back", "Next >", "Finish", and "Cancel".

Ilustración 93 Importación para flujo de solicitud de usuarios

Posteriormente se ingresa el archivo respectivo creado previamente por el macro y se selecciona continuar hasta finalizar la solicitud de importación, la herramienta iniciara el proceso el cual varia en su duración dependiendo de la cantidad de registros cargados, en caso de presentar un error en algún campo este mostrara un mensaje de error indicando específicamente cual campo fue y procederá a no cargar dicho registro, en caso de presentarse esto es necesario realizar el borrado de los demás registros y reiniciar la importación posterior a modificar el campo del error.

Una vez se termine el proceso de importación se mostrará un mensaje de cargue satisfactorio y la cantidad de registros que se crearon a partir de este.

Action: Import records
Result:
Success import 280
Failed import 0

Ilustración 94 Mensaje de registros importados

Para finalizar el proceso se procede a cargar el flujo de solicitud de usuarios con los registros previamente importados, se agrega a su vez a dicho flujo el mensaje de aprobación del director y subgerente del banco.

5.3.3.4 Análisis de código

Una de las labores más recurrentes es el análisis del código para validar las funciones de la herramienta en sus flujos y registros, la herramienta utiliza principalmente el lenguaje de programación de Basic con opción de realizar algunas acciones con el lenguaje Perl, este análisis se realiza principalmente por medio de la herramienta de Rational ClearQuest Designer, este es el que permite observar los campos, comportamientos, acciones, código y ciclo de vida de un flujo, a su vez en este es donde se realizan los desarrollos en caso de aparecer un nuevo requerimiento para la herramienta.

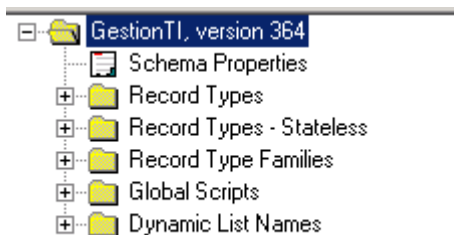


Ilustración 95 ClearQuest Designer - carpetas

Dentro de las carpetas se puede acceder a todos los flujos y registros actuales de la herramienta, en la carpeta “Record Types” se encuentran específicamente los flujos, mientras que en la carpeta “Record Types Stateless” se encuentran todos los registros.

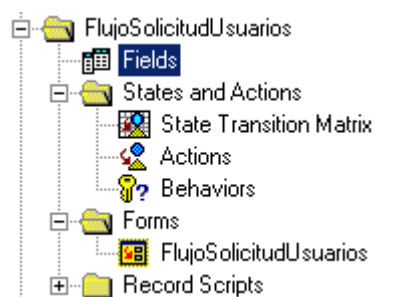


Ilustración 96 FlujoSolicitudUsuarios - Fields

Al acceder a la carpeta de un flujo en específico se observa todo lo que lo compone, en “Fields” se encuentran todos los campos que este flujo contiene, estos funcionan de manera similar a una variable global para el flujo, pueden ser asignados directamente a los campos de la forma del flujo y contener diferentes tipos de información como listas, enteros o caracteres alfanuméricos, a su vez presentan campos en los cuales se puede realizar su manipulación directamente por código.

	Field Name	Type	Default Value	Permission	Value Changed	Validation	
►	dbid	DBID					
	is_active	INT					
	id	ID					
	State	STATE					
	version	INT					
	lock_version	INT					
	locked_by	REFERENCE					DEFAULT
	ratl_mastership	REFERENCE					DEFAULT
	history	JOURNAL					
	is_duplicate	INT					
	unduplicate_state	SHORT_STRING					
	record_type	RECORDTYPE					
	tipo_solicitud	SHORT_STRING			BASIC	BASIC	CONSTANT_LIST
	tipo_usuario	SHORT_STRING			BASIC	BASIC	CONSTANT_LIST
	ultimo_estado	SHORT_STRING					
	at_field_history	REFERENCE					DEFAULT
	usuario_radicaion	REFERENCE	BASIC		BASIC		DEFAULT
	Tipo_Ambiente	SHORT_STRING			BASIC	BASIC	CONSTANT_LIST

Ilustración 97 Listado de campos para el flujo SolicitudUsuarios

El “Default Value” permite definir si se desea que el campo tenga algún valor definido por defecto, “Permission” permite definir si se desea que el campo requiera permisos para poder acceder al mismo, “Value Changed” ejecuta código cada vez detecte que se modifique el campo en cuestión, finalmente “Validation” valida por medio de código que el campo tenga un valor correcto dentro de los parámetros de este.

El “State Transition Matrix” se encuentra dentro de la carpeta “States and Actions” del flujo.



Ilustración 98 FlujoSolicitudUsuarios - State Transition Matrix

Este permite crear y validar la secuencia de los estados del flujo, indicando en la primera fila el estado en el que se encuentra, debajo de cada estado se pueden observar las

acciones que se pueden ejecutar en dicho estado y a la izquierda de dichas acciones en la primera columna, se evidencia el estado al cual se desplazaran en caso de acceder a esta acción.

From	10_Radicada	15_Solicitud_Informacion	20_En_Aprobación	25_Solicitud_Informacion	30_En_Administracion	50_Finalizado
10_Radicada			Devolver_Solicitud			Renovar_Usuario
15_Solicitud_Informacion			Solicitar_Info_Ente			
20_En_Aprobación	Solicitar_Aprobacion	Solicitar_Apro_Ente				
25_Solicitud_Informacion					Solicitar_Informacion	
30_En_Administracion	Solicitar_Inactivacion		Solicitar_Creacion	Solicitar_Creacion		
50_Finalizado					Administrar_Usuarios	
90_Cancelada	Cancelar	Cancelar	Cancelar	Cancelar	Cancelar	Cancelar
99_Rechazada			Rechazar		Rechazar	
40_En_Revision					Revisión	
26_Solicitud_Informacion					Solicitar_Informacion_In	

Ilustración 99 Transición de estados con sus respectivas acciones

Por ejemplo, en la imagen previa se observa que una de las acciones del estado “10_Radicada” es la de “Solicitar_Aprobacion” y en caso de accionar dicha acción esta dirigirá al flujo al estado “20_En_Aprobación”. Esto se analiza de igual forma para los demás estados y acciones del flujo.

Posteriormente en “Actions” se pueden observar detalladamente las acciones que se pueden ejecutar en el flujo.



Ilustración 100 FlujoSolicitudUsuarios - Actions

Action Name	Type	Access Control	Initialization	Validation	Commit	Notification
► Crear	SUBMIT	All Users	BASIC	BASIC	BASIC	
Import	IMPORT	All Users				
Solicitar_Aprobacion	CHANGE_STATE	BASIC	BASIC	BASIC	BASIC	
Modificar_Aprobacion	MODIFY	BASIC	BASIC	BASIC		
Solicitar_Informacion	CHANGE_STATE	BASIC	BASIC	BASIC		
at_Base	BASE	All Users		PERL	PERL	
EN_Base	BASE	All Users				PERL
Solicitar_Creacion	CHANGE_STATE	BASIC	BASIC		BASIC	
Cancelar	CHANGE_STATE	BASIC	BASIC			
Rechazar	CHANGE_STATE	BASIC	BASIC	BASIC		
Modificar_Admin	MODIFY	User Groups	BASIC			
Eliminar	DELETE	User Groups				
Solicitar_Inactivacion	CHANGE_STATE	BASIC	BASIC			
Devolver_Solicitud	CHANGE_STATE	BASIC	BASIC	BASIC	BASIC	
Solicitar_Info_Ente	CHANGE_STATE	BASIC	BASIC	BASIC		
Solicitar_Apro_Ente	CHANGE_STATE	BASIC	BASIC	BASIC		
Administrar_Usuarios	CHANGE_STATE	BASIC	BASIC			
Renovar_Usuario	CHANGE_STATE	BASIC	BASIC		BASIC	
Finalizar	CHANGE_STATE	User Groups				
Revisión	CHANGE_STATE	User Groups				
Solicitar_Informacion_In	CHANGE_STATE	BASIC	BASIC	BASIC		

Ilustración 101 Listado de acciones para el flujo SolicitudUsuarios

En esta sección se define el nombre de la acción, el tipo de acción la cual define si cambia el estado del flujo o si modifica parámetros únicamente entre otros, “Access Control” define directamente o por medio de código cual usuario tiene acceso para ejecutar esta acción, “Initialization” define con código los cambios que se realizan en el flujo al inicializar la acción, esto puede incluir por ejemplo el acceso a un campo previamente bloqueado, “Validation” valida al momento de guardar la acción que cumpla con lo solicitado para efectivamente ejecutar dicha acción, “Commit” ejecuta por código las acciones posteriores a la ejecución de este, generalmente se realizan tareas de almacenar información en la base de datos, finalmente “Notification” define las notificaciones que se realizan para dicha acción.

La sección de “Behaviors” define a partir del estado del flujo la condición por defecto en la que se va a encontrar un campo.

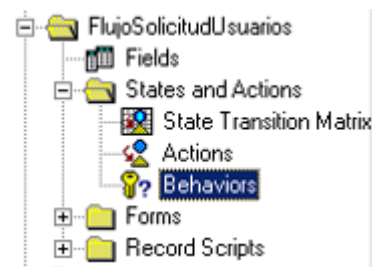


Ilustración 102 FlujoSolicitudUsuarios - Behaviors

	10_Radicada	20_En_Aprobación	30_En_Administración	50_Finalizado	90_Cancelada	99_Rechazada	25_Solicitud_Información
dbid	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
is_active	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
id	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
State	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
version	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
lock_version	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
locked_by	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
ratl_mastership	OPTIONAL	OPTIONAL	OPTIONAL	OPTIONAL	OPTIONAL	OPTIONAL	OPTIONAL
history	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
is_duplicate	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
unduplicate_state	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
record_type	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
tipo_solicitud	MANDATORY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
tipo_usuario	MANDATORY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
ultimo_estado	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
at_field_history	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
usuario_radicalcion	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY
Tipo_Ambiente	MANDATORY	READONLY	READONLY	READONLY	READONLY	READONLY	READONLY

Ilustración 103 Listado comportamientos de campos para cada estado

Estas condiciones definen si el campo es de tipo “READONLY” o solo lectura, “MANDATORY” o de carácter obligatorio y “OPTIONAL” o de carácter opcional. Es necesario mencionar que si bien esta es la condición inicial en la que se encuentran los campos, durante la ejecución del flujo se puede modificar por código según se requiera.



Ilustración 104 FlujoSolicitudUsuarios - Forms

Finalmente, dentro de la carpeta “Forms” se encuentra el GUI o interfaz gráfica del flujo, la cual es la que visualizara el usuario.

Ilustración 105 GUI interfaz gráfica para el flujo SolicitudUsuarios

En esta forma se agregan en la interfaz los diferentes campos que requiera el flujo, estos pueden ser cuadros de texto, tablas, listas desplegables, pestañas y en general todos los tipos de campo comunes que se suelen observar en una interfaz de usuario. Estas se agregan por medio de un cuadro de herramientas que se muestra al acceder a la forma.



Ilustración 106 Cuadro de herramientas para interfaz grafica

Al acceder a un campo se observan las propiedades de este y a su vez asignarle un campo al cuadro en específico.

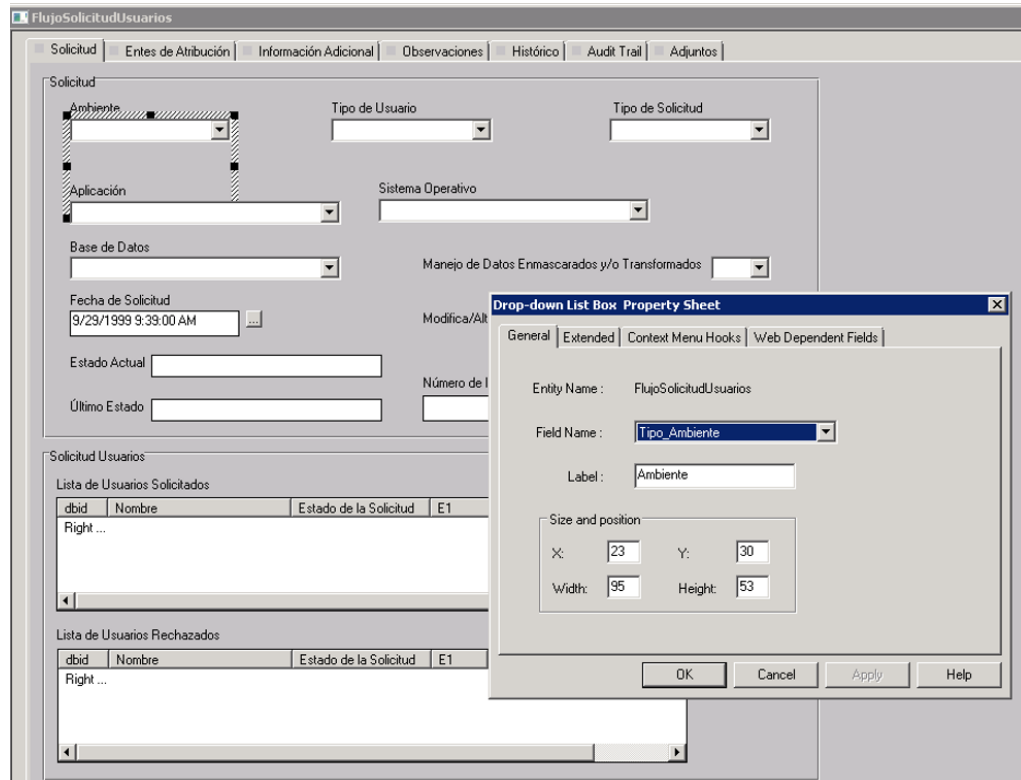


Ilustración 107 Propiedades de los campos o cuadros de texto

Como se observa en la imagen anterior, “Ambiente” es un cuadro de “Drop-down List” y se encuentra asignado al campo “Tipo_Ambiente”.

5.3.4 Desarrollo con Rational ClearQuest

Al igual que con la herramienta de Urbancode Deploy, como parte del soporte es necesario realizar modificaciones a la herramienta según se presenten requerimientos de esta, durante el tiempo de practica se presentaron unas modificaciones que el usuario solicito realizar para un flujo, lo que dio origen a un requerimiento después de su respectivo análisis, posteriormente se realizó el respectivo desarrollo de este siguiendo los siguientes procedimientos.

5.3.4.1 Definición de los cambios a realizar

El cliente definió unas modificaciones que se debían realizar al flujo, por lo que se realizó la toma de todos los cambios que esto requería y se formalizo en un primer documento de análisis, estos cambios se definieron en el documento de la siguiente forma:

Para el flujo base de datos de la herramienta ClearQuest se evidencia los siguientes inconvenientes reportados por el usuario impidiendo que los mismos puedan continuar con la gestión de los flujos y que harán parte de la estimación:

Para llevar a cabo el desarrollo del requerimiento es necesario realizar ajustes sobre los siguientes componentes:

- ✓ Flujo Base de Datos:
 - Pestaña Desarrollo.
 - Pestaña Pruebas.
 - Pestaña Devolución y respaldo.

1.1 Flujo Base de Datos – Registro Desarrollo

 Para la pestaña de “Desarrollo” en el registro relacionado “Registro_Desarrollo” se presentan los siguientes ajustes:

- ✓ Descripción (Campo dinámico a selección del usuario de forma obligatoria).
 - Modificación.
 - Devolución.
 - Verificación
- ✓ URL Instructivo de Instalación (Nuevo campo).

Incluyendo los campos adicionales y ajustándolos, se vería de la siguiente forma la nueva pestaña de “Desarrollo”.

Ilustración 108 Registro Desarrollo actualizado

1.2 Flujo Base de Datos – Pestaña Pruebas

Para la pestaña de “Pruebas” se presentan los siguientes ajustes:

- ✓ Justificación de no presentar pruebas.

Incluyendo los campos adicionales y ajustándolos, se vería de la siguiente forma la nueva pestaña de “Pruebas”.

FlujoBaseDatos:CQPRO00131363

Guardar Guardar y

*General *Desarrollo **Pruebas** *Devolución y Respaldo Aprobaciones Observaciones Histórico Administrativo

Pruebas

☒ Aplica Archivo de Pruebas

Archivo Pruebas

Nombre de archivo	Tamaño	Descripción

Descargar
Añadir...
Eliminar

* Justificación de no presentar pruebas

Ilustración 109 Pestaña Pruebas actualizada

1.3 Flujo Base de Datos – Pestaña Implementación

- Para la pestaña de “Devolución y Respaldo” se presentan los siguientes ajustes:
- ✓ Cambio del nombre de la pestaña por “Implementación”.
 - ✓ Retirar campos de archivos adjuntos.
 - ✓ Justificación si no existe copia de seguridad (solo se habilitará si la respuesta del campo “¿Existe copia de seguridad?” es NO).
 - ✓ Razones para realizar la tarea.
 - ✓ Causas del Problema
 - ✓ Impacto.
 - ✓ Riesgo (Lista dinámica).
 - ✓ Causa de Falla (Lista Dinámica).
 - Fuente de Datos.
 - Aplicación.

➤ Usuario u Operativo.

Incluyendo los campos adicionales y ajustándolos, se vería de la siguiente forma la nueva pestaña de “Implementación”.

FlujoBaseDatos:CQPRU00008002 ▼

*General *Desarrollo ***Implementación** Aprobaciones Observaciones Histórico Administrativo Audit Trail

Implementación

*¿Existe copia de seguridad?

* Justificación si no existe copia de seguridad

* Razones para realizar la tarea

* Causas del problema

* Impacto

* Riesgo

* Causa de falla

Ilustración 110 Pestaña Implementación actualizada

Partiendo de los cambios mencionados anteriormente se definió en este documento una estimación del tiempo que este requerimiento necesita. El esfuerzo para implementar está distribuido de la siguiente manera:

Tarea	Roles Participantes	Entregables	Horas
Análisis	Analista	Documento Análisis	5
Diseño Técnico	Analista	Diseño Técnico	5
Desarrollo	Analista	Componentes	15
Pruebas y Documentación	Analista	Set de Pruebas	10
TOTAL DE HORAS			35

Tabla 3 Estimación tiempos desarrollo

Calendario Propuesto

Fecha de Inicio	Por definir
Fecha de entrega:	Por definir
% Confiabilidad de la estimación	% 90

Tabla 4 Formato calendario propuesto

Una vez la estimación es aprobada por el cliente, se da inicio al desarrollo en el servidor de pruebas.

5.3.4.2 Desarrollo del requerimiento

Para iniciar se modifica la interfaz del usuario para replicar la nueva solicitada, se accede a ClearQuest Designer y en la carpeta del flujo se accede a la carpeta de “Forms” donde se encuentra el GUI del flujo. Utilizando el cuadro de herramientas se agregan los cuadros de texto y campos que se necesitan.



Ilustración 111 Creación cuadro de texto

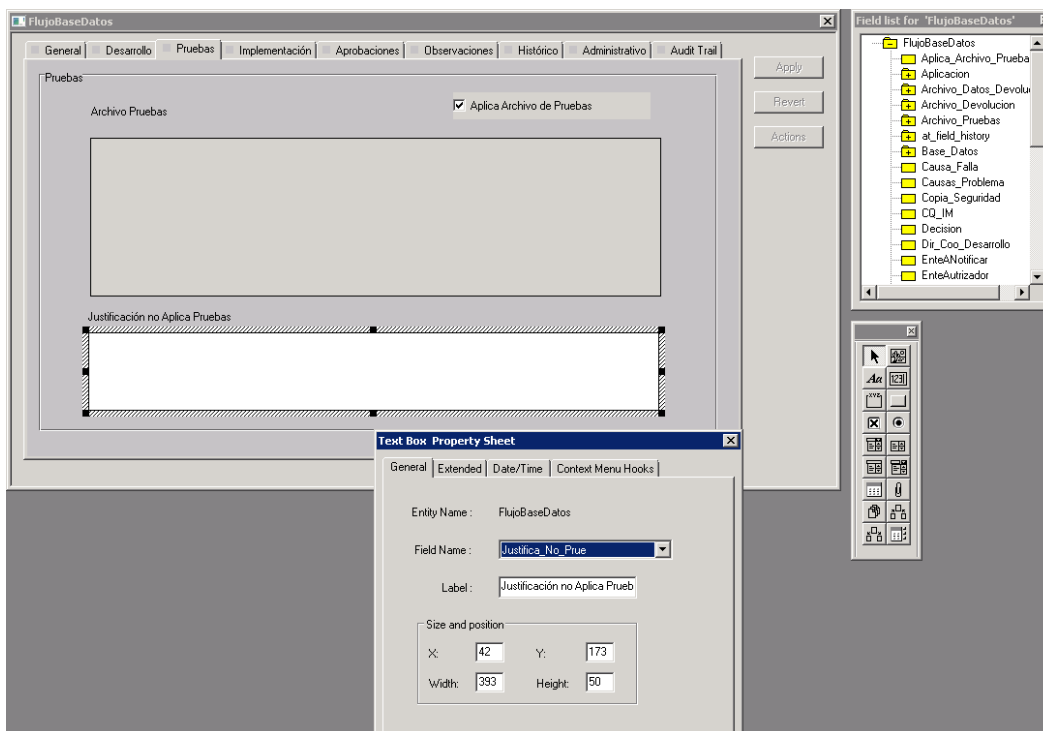


Ilustración 112 Propiedades cuadro de texto

Se crea el “Text Box” y se le asigna el campo “Justifica_No_Prue”, estos campos son creados en la sección de “Fields” del flujo. De igual manera se continua con todos los nuevos campos solicitados.

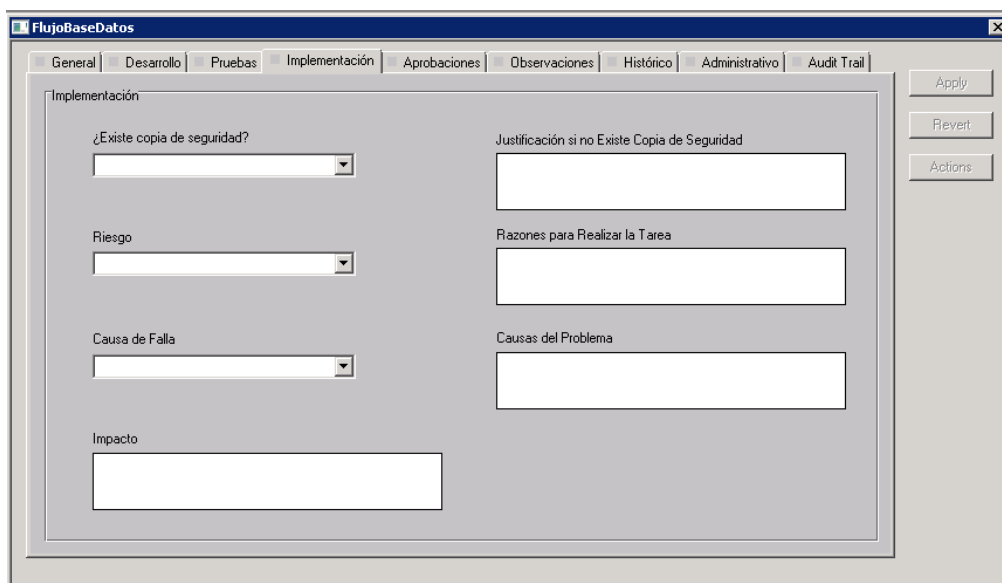


Ilustración 113 Pestaña Implementación final

Registro

Registro Base de Datos | Observaciones | Histórico

Información General

ID Del Requerimiento:

Tipo de Archivo:

Código de Revisión:

Descripción:

URL:

URL Instructivo de Instalación:

Apply
Revert
Actions

Ilustración 114 Registro Desarrollo final

Luego para los campos nuevos de listas se requiere crear una lista dinámica nueva para asignarles, como por ejemplo para la lista de “Causa de Falla”.

Drop-down List Box Property Sheet

General | Extended | Context Menu Hooks | Web Dependent Fields

Entity Name: FlujoBaseDatos

Field Name:

Label:

Size and position

X: Y:

Width: Height:

OK Cancel Apply Help

Causa de Falla:

Impacto:

Ilustración 115 Asignación de campo para lista

Para realizar la creación de una nueva lista dinámica, se busca directamente la carpeta “Dynamic List Names” en la cual se almacenan todas las listas.

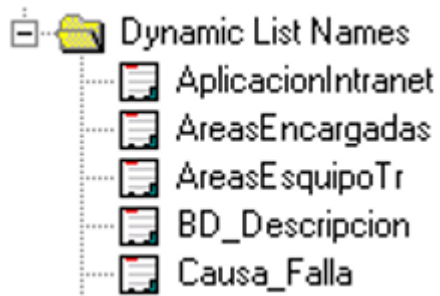


Ilustración 116 Carpeta listas dinámicas

En esta carpeta se realiza la creación de la nueva lista de Causa_Falla

Posteriormente se le asigna a el campo previamente creado que va a utilizar la lista, se selecciona “DYNAMIC LIST” para este campo.



Ilustración 117 Asignación lista dinámica para el campo Causa_Falla

Se selecciona la lista previamente creada de “Causa_Falla” y se marcan los dos campos de “Recalculate Choice List” y “Limit to list”.

Recalculate Choice list: Actualiza constantemente los campos de la lista.

Limit to list: Define que el tamaño de la lista se ajuste fijamente a los campos que esta contenga.

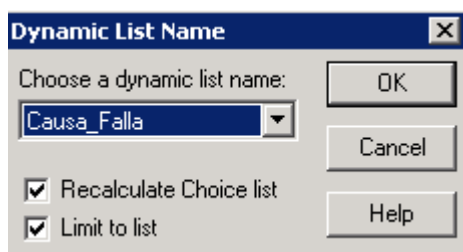


Ilustración 118 Selección de lista dinámica

Luego se accede a Test work al lado de la opción de guardar, esto compila y ejecuta los cambios en el ambiente de pruebas cuando no se presentan errores en el código.

Ilustración 119 Opción guardar y Test work

Cuando finaliza la compilación, automáticamente abre una nueva ventana, en esta ventana se accede a la pestaña “Edit” y en NamedLists se pueden observar las listas creadas.

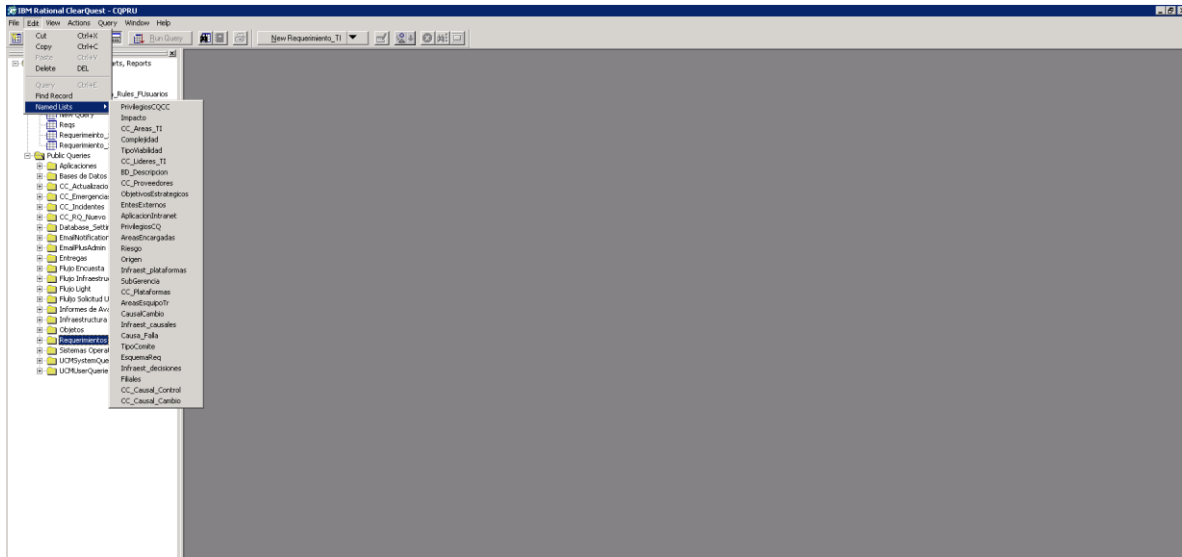


Ilustración 120 Selección de listas dinámicas creadas

Al acceder a la que necesitamos “Causa_Falla” inicialmente se encuentra vacía, por lo que se procede a llenar con los parámetros solicitados.

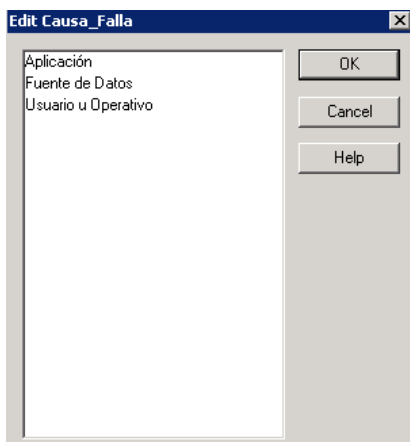


Ilustración 121 Cague de lista dinámica con opciones

Dado que este proceso se realiza inicialmente en la base de datos de pruebas, al momento de realizar el despliegue a producción es necesario acceder a nuevamente a la base de datos de este y realizar la carga de las listas.

Con los campos y las listas creadas se proceden a configurar las validaciones que se solicitaron, para este caso se requiere que cuando la respuesta del campo “¿Existe copia de seguridad?” sea negativo entonces el campo de “Justificación si no Existe Copia de Seguridad” se convierta en un campo obligatorio, de igual forma para el campo “Aplica archivo de pruebas”. Para esto se agrega una validación en el “ValueChanged” del respectivo campo.

```
Sub copia_seguridad ValueChanged(fieldname)
' fieldname As String
' record type name is FlujoBaseDatos
' field name is Copia_Seguridad
REM This hook will be run each time the field's value
REM changes. This hook is run before
REM the new value is validated
REM so be aware the new value may not be valid.
REM Example:
REM Dim op_sys
REM op_sys = GetFieldValue(fieldname).GetValue()
REM If op_sys = "windows" Then
REM SetFieldValue "os_version", "NT 4.x"
REM <etc.>
REM End If
set sessionObj = GetSession
Existecopia = GetFieldValue("Copia_Seguridad").GetValue()

If Existecopia = "NO" then
SetFieldRequirednessForCurrentAction "Justifica_No_Copia", AD_MANDATORY
else
SetFieldRequirednessForCurrentAction "Justifica_No_Copia", AD_READONLY
SetFieldValue "Justifica_No_Copia", ""
End If
End Sub
```

Ilustración 122 Validación de campo para modificar comportamiento

Dentro de este se define una variable que captura el valor del campo, dado que este se ejecuta cada vez que se realice una modificación la variable contendrá el valor actual del campo, luego a partir de esta variable se define un IF en el cual se valida el contenido del campo, en caso de ser “No” entonces modifica el campo de justificación como mandatorio, de lo contrario deje el campo de justificación de solo lectura y elimine el texto que este contenga.

Esta validación se realiza de igual forma para la otra validación del otro campo.

Finalmente se agrega una notificación para cuando el usuario ejecuta la acción de solicitar aprobación, cuyo propósito es informar al usuario que debe realizar la aprobación para dar continuidad al flujo.

Para esto se agrega un script en “Validation” para la acción de “Solicitar_Aprobación” la cual era la que requiere enviar una notificación al director asignado.

Action Name	Type	Access Control	Initialization	Validation	Commit	Notification	
Crear	SUBMIT	User Groups	BASIC	BASIC			
Import	IMPORT	All Users					
Cancelar	CHANGE_STATE	BASIC	BASIC				
Rechazar	CHANGE_STATE	BASIC	BASIC	BASIC			
Solicitar_Aprobación	CHANGE_STATE	BASIC	BASIC	BASIC			
Aprobar_Solicitud	CHANGE_STATE	BASIC	BASIC				
Modificar_RegDesarrollo	MODIFY	BASIC	BASIC				
Modificar_Admin	MODIFY	User Groups	BASIC				
Reanudar	CHANGE_STATE	BASIC	BASIC				
Reasignar_DirectorCoo	MODIFY	BASIC	BASIC				
at_Base	BASE	All Users		PERL	PERL		
EN_Base	BASE	All Users				PERL	
EmailPlusNotify	BASE	All Users				PERL	

Ilustración 123 Creación de código Validation en la acción Solicitar Aprobación

```
Function flujobasedatos Validation(actionname, actiontype)
' actionname As String
' actiontype As Long
' flujobasedatos_Validation As String
' action is Solicitar_Aprobación
' record type name is FlujoBaseDatos
REM Return a non-empty string explaining why the action
REM cannot commit with the current values.
REM Or, if it is valid, return an empty string value.
REM Example:
REM   Dim value_info
REM   Set value_info = GetFieldValue("some field")
REM   If Len(value_info.GetValue()) < 10 Then
REM       flujobasedatos_Validation = "Must be at least 10 chars long"
REM   End If
set sessionObj = getsession

username = GetFieldValue("Dir_Coo_Desarrollo").GetValue()
coma = "-"
pos = InStr(1,username,coma)
usr = Left(username,pos - 2)
SetFieldValue "EnteAtrizador", usr
set usObj = sessionObj.GetEntity("users", usr)
SetFieldValue "EnteANotificar", usObj.GetFieldValue("email").GetValue()
End Function
```

Ilustración 124 Captura de datos de usuario a notificar

En este paso se captura la información del director en cuestión junto con su correo y se almacena en unas variables para su posterior uso.

Finalmente se crea el registro de “Email_Notification_Rule”

Email_Notification_Rule:Nuevo registro ▾

***Main** Header Subject and Body Security

***Name:** ***Record Type:**

Order: ☒ Active

Actions: Field Change:

Condition:

Priority:

Ilustración 125 Creación registro Email Notification Rule

En este se define las reglas para enviar la notificación, junto con el contenido del mensaje.

Email_Notification_Rule:Prueba-FlujoBaseDatos-SolAprob ▾

Main Header Subject and Body Security

Name: Record Type:

Order: ☒ Active

Actions: Field Change:

Condition:

Priority:

Ilustración 126 Asignación de parámetros en Main

Se asigna el nombre del registro, el flujo o registro de la notificación “FlujoBaseDatos” para este caso. Se agrega la acción o acciones que lo van a ejecutar “Solicitar_Aprobación” y se agrega la variable previamente capturada de “EnteAutrizador”.

Email_Notification_Rule:Prueba-FlujoBaseDatos-SolAprob ▾

Main Header Subject and Body Security

From:

To: \$EnteANotificar

CC: \$usuario_radicacion.email

BCC:

Header Add-in:

Ilustración 127 Asignación de destinatarios en Header

Se asigna el destinatario a partir de la variable “EnteANotificar”, para el CC se asigna la variable “usuario_radicacion.email” propia del usuario que radico el flujo.

Email_Notification_Rule:Prueba-FlujoBaseDatos-SolAprob ▾

Main Header Subject and Body Security

Subject: TI - Aprobación - Flujo Solicitud de Usuarios \$id

Body: Se solicita su aprobación para continuar con el flujo del \$id
Favor ingresar al siguiente vínculo [REDACTED]
Agradecemos la atención.
*** Esta es una notificación automática *** No responda a este correo ***

Ilustración 128 Asignación de mensaje en Subject and Body

Finalmente se asignan el asunto y cuerpo del mensaje. Estos utilizan la variable "id" para indicar el número de identificación propio del flujo.

5.3.4.3 Pruebas y documentación

Se diseñan unos casos de prueba para verificar el correcto funcionamiento de flujo, estos son a su vez validados por el cliente el cual a su vez realiza sus propias pruebas según considere necesario.

ID	CASO DE PRUEBA	RESULTADO ESPERADO	APROBADO
1	Visualización del Registro Desarrollo de la pestaña "Desarrollo" – Flujo Base de Datos	La pestaña de desarrollo debe abrir el registro con los parámetros nuevos.	SI
2	Verificar que los campos nuevos del Registro Desarrollo de la pestaña "Desarrollo" sean obligatorios – Flujo de Base de Datos	EL registro debe validar que los parámetros se llenen para poder guardarlo satisfactoriamente.	SI
3	Visualización de los campos de la pestaña "Pruebas" – Flujo Base de Datos	El sistema debe mostrar los campos nuevos en la pestaña.	SI
4	Verificar funcionalidad del campo "Justificación no Aplica Pruebas" de la pestaña "Pruebas" – Flujo Base de Datos	El campo debe ser obligatorio en caso de no marcar "Aplica Archivo de Pruebas"	SI
5	Visualización del nombre "Implementación" y los campos de la pestaña previamente nombrada " Devolución y Respaldo " – Flujo Base de Datos	El sistema debe mostrar el nuevo nombre de la pestaña con sus respectivos campos.	SI
6	Visualización de los campos de la pestaña "Implementación" – Flujo Base de Datos	El sistema debe volver mandatorio el campo "Justificación si no existe copia de seguridad" si la respuesta del campo "¿Existe copia de seguridad?" es NO	SI

Tabla 5 Formato casos de prueba ClearQuest

Una vez se comprueben todos los casos de prueba, se da el visto bueno para realizar el despliegue a producción. Para finalizar se deben llenar los formatos de documentación del banco, una vez enviados se procede a realizar un control de cambios en el cual se registra la fecha en la que se debe realizar el despliegue.

Luego de realizar el despliegue de la versión, se confirma el correcto funcionamiento de la nueva versión en producción y se cierra el requerimiento.

6 CONCLUSIONES

- Se evidenció la diferencia que existen entre el ambiente universitario y el ambiente laboral, los deberes y responsabilidades constantes a los que se enfrenta un ingeniero de sistemas e informática y los riesgos que estos presentan para la organización, dado esto también se evidencio como la comunicación constante y el trabajo en equipo es vital para llevar los proyectos al éxito.
- Durante el periodo de practica se logró comprender el funcionamiento y control de las herramientas Urbancode Deploy y Rational ClearQuest, solucionando incidentes que se presentan todos los días en el banco de occidente, permitiendo de esta manera dar continuidad a los procesos del banco y reduciendo la carga de los demás encargados del soporte de las herramientas.
- Se evidencio la importancia de la línea de gestión de TI, gerencia de proyectos y teoría de la negociación, al poder aplicar los conocimientos vistos en las asignaturas para los desarrollos, las etapas y documentación que estos necesitan, a su vez se observó una mejoraría en las habilidades de atención al cliente, ya que la labor de soporte requiere de la constante interacción y negociación de propuestas con los usuarios, para con los cuales es de vital importancia ofrecer un servicio agradable y eficiente.
- Durante el periodo de practica se experimentó el estar a la cabeza de un cargo y todas las responsabilidades que esto implica, ya que se dio soporte completamente solo durante un periodo de dos semanas, esto debido a vacaciones tomadas por el encargado de las herramientas.
- Se experimento el proceso de desarrollo de un requerimiento para ambas herramientas, se participó de primera mano en todas sus fases de identificación del problema, toma de requerimientos, análisis y estimación, desarrollo, pruebas y documentación. De los cuales se obtuvo para ambos casos un resultado favorable con el cliente.

7 RECOMENDACIONES

- Se recomienda mantener documentación completa y actualizada de las acciones o funciones realizadas del cargo, esto con el fin de que en caso de que se retire una persona no se pierda la información que esta poseía, como se evidencio al retirarse el encargado anterior de las herramientas y por lo tanto se necesitó volver a investigar cómo se realizaban algunos procesos previamente solicitados por el cliente.
- Se evidencio una falta de eficiencia en el proceso de cargue masivo del banco, ya que las solicitudes de estos constantemente se demoran o se cancelan debido a que no se realizan adecuadamente, ya sea que no presentan las aprobaciones necesarias, o porque no cumplen con los parámetros solicitados para su ejecución. Se recomienda un formato específico que todos sigan, en el cual se exijan las autorizaciones necesarias y se especifiquen correctamente los campos requeridos para su ejecución, que garantice que al llenar dicho formato se puede ejecutar el proceso inmediatamente.

8 BIBLIOGRAFIA

[1 IBM, «Presentacion Agile Workshop,» Bogotá, 2019.
]

[2 Angular, «Angular,» [En línea]. Available: <https://angular.io/>. [Último acceso: 27 02
] 2019].

[3 IBM, «YOUR LEARNING LA STUDENT LEARNING PACK,» IBM, [En línea].
] Available: <https://lpb.w3bmix.ibm.com/?id=85258122006F9760>. [Último acceso: 25 02
2019].

[4 IBM, «Diapositiivas WebSphere Message Broker,» Bogotá, 2019.
]

[5 «UrbanCode Deploy plugins,» IBM, [En línea]. Available:
] <https://developer.ibm.com/urbandcode/plugins/ibm-urbandcode-deploy/>. [Último acceso:
28 04 2019].

[6 «IBM Rational ClearQuest,» IBM. [En línea]. [Último acceso: 28 04 2019].
]

[7 «UrbanCode Deploy,» IBM, [En línea]. Available:
] https://www.ibm.com/support/knowledgecenter/es/SS4GSP_6.1.1/com.ibm.udeploy.doc/topics/intro_ch.html. [Último acceso: 28 04 2019].
