

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

DESARROLLO DE MÓDULOS DE APLICATIVOS MÓVILES DE SISTEMA ERP UTILIZANDO PRÁCTICAS DEVOPS SEGURAS PARA MANTICORE LABS – MÓDULO DE RECAUDOS

TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN CIENCIAS DE LA COMPUTACIÓN

WILMAN ALBERTO ARMIJOS MACAS

wilman.armijos@epn.edu.ec

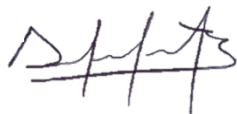
DIRECTOR: VICENTE ADRIÁN EGÜEZ SARZOSA

adrian.eguez@epn.edu.ec

DMQ, agosto 2022

CERTIFICACIONES

Yo, WILMAN ALBERTO ARMIJOS MACAS declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.



WILMAN ALBERTO ARMIJOS MACAS

Certifico que el presente trabajo de integración curricular fue desarrollado por WILMAN ALBERTO ARMIJOS MACAS, bajo mi supervisión.



VICENTE ADRIÁN EGÜEZ SARZOSA

DIRECTOR

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

WILMAN ALBERTO ARMIJOS MACAS
VICENTE ADRIÁN EGÜEZ SARZOSA

DEDICATORIA

A mi papá, por todo lo que supo darme y enseñarme, no por cosas materiales, sino muchos más importantes, por ese incondicional apoyo y todas esas guías que me han servido para ser mejor.

A mi mamá, que pase lo que pase siempre estuvo ahí cuando lo necesitaba, cuando las cosas parecían oscurecer y era ella con quien podía contar, aun estando a cientos de kilómetros de distancia.

A mi hermano, siendo esa persona que ha estado junto a mí a lo largo de este proceso, que estuvo aquí cuando más necesitaba y supo apoyarme, quien hacía todo lo posible por que yo siguiera firme y con ánimo.

Y a todos mis amigos, mis mascotas y de nuevo mi familia, porque este proceso no fue tan sencillo, incluso la incidencia del virus COVID-19 transformó esta etapa en algo más inusual y complicado, pero todos ellos estuvieron siempre ahí preocupándose, no solo porque me vaya bien en mis materias, sino porque querían que yo pudiera ser feliz, y ahora que estoy cerca de cumplir con una nueva meta, ellos son un gran motor y quizás la parte más importante que me permitió conseguir llegar a donde estoy, a todos ellos, no puedo más que estar eternamente agradecido y hacer todo porque estén orgullosos de mí.

Wilman Alberto Armijos Macas

AGRADECIMIENTO

A mi tutor, por la guía y ayuda durante el proceso de consecución del logro del proyecto dentro de esta última etapa de la carrera.

A mis docentes, que a lo largo de todo este proceso de estudio que me prepararon, guiaron y enseñaron sus conocimientos que serán parte de mi vida a partir de ahora, no solo sobre computación o una materia, sino también en la parte humana, a través de sus valores, principios, y su visión, de las cuales yo supe y sabré tomar lo mejor para la vida que queda por delante.

A las personas que se encontraban detrás de la Escuela Politécnica Nacional y fuera de ella, que fueron parte de mi crecimiento como ingeniero, pero más como ser humano, con las cuales atravesé por varios momentos los cuales forman parte de lo que soy ahora.

A mis compañeros y amigos, que fueron parte del viaje hasta este punto, aquellos con los que se ha reído, se ha disfrutado, se ha llorado, pero lo más importante, estuvieron ahí y han sido un gran apoyo. Que a pesar de las dificultades que podría significar el dinero, la distancia, las dificultades personales, o cualquier obstáculo que estaba delante, estuvieron ahí.

A mi padre, a mi madre y a mi hermano, que no solo me dieron esta oportunidad de aprender y formarme como profesional, algo que ya es demasiado, sino por todo lo que han hecho por mí a lo largo de este proceso, todas esas ayudas, esos consejos que he recibido, todas esas veces que me han dado su hombro y su apoyo cuando quizás no lo veía claro, que estuvieron ahí cuando más lo necesitaba para reconfortarme y alentarme a dar lo mejor para conseguir lo que quiero.

Wilman Alberto Armijos Macas

ÍNDICE DE CONTENIDO

CERTIFICACIONES	I
DECLARACIÓN DE AUTORÍA	II
DEDICATORIA	III
AGRADECIMIENTO	IV
ÍNDICE DE CONTENIDO	V
ÍNDICE DE TABLAS	VIII
ÍNDICE DE FIGURAS.....	X
RESUMEN.....	XIII
ABSTRACT	XIV
1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO.....	1
1.1 Objetivo general	1
1.2 Objetivos específicos.....	1
1.3 Alcance.....	1
1.4 Marco teórico.....	2
1.4.1 Enterprise Resource Planning (ERP)	2
1.4.2 SCRUM	3
1.4.3 DevOps	5
1.5 Herramientas de Desarrollo	7
1.5.1 GitLab.....	7
1.5.2 Git	8
1.5.3 Figma	10
1.5.4 Firebase	10
1.5.5 Docker.....	11
1.5.6 Android Studio.....	12
1.6 Lenguajes de Programación	12
1.6.1 Kotlin	12
1.6.2 XML.....	13

1.6.3	YAML	13
2	METODOLOGÍA	14
2.1	Implementación de Scrum	14
2.2	Implementación de DevOps.....	16
2.2.1	Configuraciones del Flujo DevOps	18
2.2.2	Ejecución de Flujo DevOps	26
2.3	Flujo de Implementación de la Aplicación	30
2.3.1	Análisis.....	30
2.3.2	Requerimientos/Historias de Usuario	30
2.3.2.1	Historia de Usuario 001	31
2.3.2.2	Historia de Usuario 002	31
2.3.2.3	Historia de Usuario 003	31
2.3.2.4	Historia de Usuario 004	32
2.3.2.5	Historia de Usuario 005	32
2.3.3	Diseño	33
2.3.3.1	Modelo de Interfaces	33
2.3.3.2	Modelo de Base de Datos	40
2.3.4	Implementación/Pruebas de Funcionalidad	45
2.3.4.1	Sprint 1.....	46
2.3.4.2	Sprint 2.....	48
2.3.4.3	Sprint 3.....	50
2.3.4.4	Sprint 4.....	53
2.3.4.5	Sprint 5.....	60
2.3.4.6	Sprint 6.....	65
2.3.4.7	Sprint 7.....	70
2.3.5	Pruebas de Usabilidad	72
2.3.6	Despliegue	80
3	RESULTADOS	82
4	CONCLUSIONES Y RECOMENDACIONES	89

4.1	Conclusiones	89
4.2	Recomendaciones	90
5	REFERENCIAS BIBLIOGRÁFICAS	92
6	ANEXOS.....	94
	ANEXO I.....	94
	ANEXO II.....	94
	ANEXO III.....	95
	ANEXO IV	98
	ANEXO V	101
	ANEXO VI	103
	ANEXO VII	108
	ANEXO VIII	111
	ANEXO IX	111
	ANEXO X	111
	ANEXO XI	111
	ANEXO XII	112
	ANEXO XII	112
	ANEXO XIV.....	112
	ANEXO XV.....	112
	ANEXO XVI.....	112
	ANEXO XVII.....	113
	ANEXO XVIII.....	113
	ANEXO XIX.....	113
	ANEXO XX.....	113
	ANEXO XXI.....	113
	ANEXO XXII.....	114

ÍNDICE DE TABLAS

Tabla 2.1.1. Roles de Scrum implementados.....	14
Tabla 2.1.2. Artefactos de Scrum implementados.....	15
Tabla 2.1.3. Eventos de Scrum implementados.....	16
Tabla 2.3.2.1.1. Resumen de Historia de Usuario 001.....	31
Tabla 2.3.2.2.1. Resumen de Historia de Usuario 002.....	31
Tabla 2.3.2.3.1. Resumen de Historia de Usuario 003.....	32
Tabla 2.3.2.4.1. Resumen de Historia de Usuario 004.....	32
Tabla 2.3.2.5.1. Resumen de Historia de Usuario 005.....	32
Tabla 2.3.4.1.1 Resultados del Sprint Review.....	46
Tabla 2.3.4.1.2 Resultados del Sprint.....	47
Tabla 2.3.4.2.1 Resultados del Sprint Review.....	49
Tabla 2.3.4.2.2 Resultados del Sprint.....	49
Tabla 2.3.4.3.1 Resultados del Sprint Review.....	52
Tabla 2.3.4.3.2 Resultados del Sprint.....	53
Tabla 2.3.4.4.1 Resultados del Sprint Review.....	59
Tabla 2.3.4.4.2 Resultados del Sprint.....	60
Tabla 2.3.4.5.1 Resultados del Sprint Review.....	64
Tabla 2.3.4.5.2 Resultados del Sprint.....	64
Tabla 2.3.4.6.1 Resultados del Sprint Review.....	68
Tabla 2.3.4.6.2 Resultados del Sprint.....	69
Tabla 2.3.4.7.1 Resultados del Sprint Review.....	71
Tabla 2.3.4.7.2 Resultados del Sprint.....	71
Tabla 2.3.5.1 Preguntas del método SUS [25].....	72
Tabla 2.3.5.2 Clasificación de resultados SUS.....	73
Tabla 2.3.5.3 Equivalencia de encuesta y escala SUS.....	74
Tabla 2.3.5.4 Resultados del primero usuario.....	75
Tabla 2.3.5.5 Resultados del segundo usuario.....	75
Tabla 2.3.5.6 Resultados del tercer usuario.....	76
Tabla 2.3.5.7 Resultados del cuarto usuario.....	77
Tabla 2.3.5.8 Resultados del quinto usuario.....	77
Tabla 2.3.5.9 Resultados del sexto usuario.....	78
Tabla 2.3.5.10 Resultados del séptimo usuario.....	78
Tabla 2.3.5.11 Resultados del octavo usuario.....	79
Tabla 2.3.5.12 Resultados del noveno usuario.....	79

Tabla 3.1 Ocurrencias de flujos DevOps según su resultado	82
Tabla 3.2 Tiempos de Ejecución de Flujos DevOps	83
Tabla 3.3 Número de operaciones para el flujo DevOps y un flujo común	84
Tabla 3.4 Porcentaje de actividades completadas en el Sprint	85
Tabla 3.5 Resultados totales de SUS	86

ÍNDICE DE FIGURAS

Figura 2.2.1. Diagrama de ejecución de DevOps	17
Figura 2.2.1.1 Repositorio remoto en GitLab.....	19
Figura 2.2.1.2 Carpeta objetivo como repositorio local.....	19
Figura 2.2.1.3 Clonación de proyecto a repositorio local	20
Figura 2.2.1.4 Desactivación de opción “Enable shared runners for this project”	21
Figura 2.2.1.5 Registro de un nuevo Runner de forma local.....	21
Figura 2.2.1.6 Definición de la URL del Runner local	21
Figura 2.2.1.7 Definición del Token dentro de GitLab.....	22
Figura 2.2.1.8 Definición del Token del Runner local	22
Figura 2.2.1.9 Definición de la Descripción del Runner local.....	22
Figura 2.2.1.10 Definición del Ejecutor del Runner local	23
Figura 2.2.1.11 Definición de la Imagen del Runner local	23
Figura 2.2.1.12 Inicio del Runner local	24
Figura 2.2.1.13 Runner activo en GitLab.....	24
Figura 2.2.1.14 Descripción de la imagen en la configuración de CI	24
Figura 2.2.1.15 Descripción de ejecuciones previas en la configuración de CI.....	24
Figura 2.2.1.15 Descripción de las etapas en la configuración de CI	25
Figura 2.2.1.15 Etapa “build-job”	25
Figura 2.2.1.16 Etapa “test-job”	26
Figura 2.2.2.1 Especificación de los cambios a subir al repositorio remoto	26
Figura 2.2.2.2 Commit de los cambios al repositorio local.....	27
Figura 2.2.2.3 Push de los cambios al repositorio remoto	27
Figura 2.2.2.4 Cambios en el repositorio remoto.....	27
Figura 2.2.2.5 Ejecución de instrucciones con Integración Continua.....	28
Figura 2.2.2.6 Especificación de los cambios a subir al repositorio remoto	28
Figura 2.2.2.7 Ejecución del flujo DevOps finalizada.....	29
Figura 2.2.2.8 Resultados de la ejecución del trabajo	29

Figura 2.3.3.1.1 Diagrama de Interfaces (Vista General)	34
Figura 2.3.3.1.2 Interfaz – Inicio de Sesión	35
Figura 2.3.3.1.3 Interfaz – Menú Principal.....	35
Figura 2.3.3.1.4 Interfaz – Cuentas	36
Figura 2.3.3.1.5 Interfaz – Cuenta de la Organización	36
Figura 2.3.3.1.6 Interfaz – Cuenta del Cliente	37
Figura 2.3.3.1.7 Interfaz – Informe Contable	37
Figura 2.3.3.1.8 Interfaz – Detalles de Transacción	38
Figura 2.3.3.1.9 Interfaz – Detalles de Documento.....	38
Figura 2.3.3.1.10 Interfaz – Periodo Contable	39
Figura 2.3.3.1.11 Interfaz – Asiento Contable de Transacción	39
Figura 2.3.3.1.12 Interfaz – Agregar Documento a Transacción	40
Figura 2.3.3.2.1 Estructura de la Base de Datos – Colecciones.....	41
Figura 2.3.3.2.2 Estructura – Usuarios.....	41
Figura 2.3.3.2.3 Estructura – Cuentas Contables.....	42
Figura 2.3.3.2.4 Estructura – Detalles de Documentos	42
Figura 2.3.3.2.5 Estructura – Cuentas de la Organización	43
Figura 2.3.3.2.6 Estructura – Cuentas de Terceros.....	43
Figura 2.3.3.2.7 Estructura – Transacciones Contables	44
Figura 2.3.3.2.8 Estructura – Periodos Contables	44
Figura 2.3.3.2.9 Estructura – Documentos	45
Figura 2.3.4.2.1 Proyecto dentro del repositorio remoto.....	48
Figura 2.3.4.3.1 Archivo de Configuración Inicial	50
Figura 2.3.4.3.2 Archivo de Configuración Inicial	51
Figura 2.3.4.3.3 Implementación de la interfaz de Inicio de Sesión	51
Figura 2.3.4.3.4 Ejecución de flujo DevOps (build).....	52
Figura 2.3.4.4.1 Implementación de la interfaz de la Página Principal.....	54
Figura 2.3.4.4.2 Implementación de la interfaz de Cuentas.....	54
Figura 2.3.4.4.3 Implementación de la interfaz de Cuentas de la Organización	55

Figura 2.3.4.4.4 Implementación de la interfaz de Cuentas de Terceros	55
Figura 2.3.4.4.5 Implementación de la interfaz del Periodo Contable	56
Figura 2.3.4.4.6 Implementación de la interfaz de Transacciones	56
Figura 2.3.4.4.7 Implementación de la interfaz de Documentos	57
Figura 2.3.4.4.8 Implementación de la interfaz de Informe Contable	57
Figura 2.3.4.4.9 Implementación de la interfaz de Detalles de Transacción	58
Figura 2.3.4.4.10 Implementación de la interfaz de Detalles de Documento	58
Figura 2.3.4.5.1 Configuración de Firebase App-Distribution en el proyecto	61
Figura 2.3.4.5.2 Resultados de flujos DevOps con Firebase App-Distribution	62
Figura 2.3.4.5.3 Ejecución de flujo DevOps (build)	63
Figura 2.3.4.5.4 Ejecución de flujo DevOps (build)	63
Figura 2.3.4.5.5 Ejecución de flujo DevOps (build)	64
Figura 2.3.4.6.1 Ejecución de flujo DevOps (build, test)	66
Figura 2.3.4.6.2 Ejecución de flujo DevOps (build)	68
Figura 2.3.4.7.1 Ejecución de flujo DevOps (build, test)	70
Figura 2.3.5.1 Vista previa del cuestionario SUS	74
Figura 2.3.5.1 Lanzamiento de la aplicación en Firebase App-Distribution	80
Figura 2.3.5.2 Ejecución en un dispositivo Android real	81
Figura 3.1. Resultados de los Flujos DevOps ejecutados	83
Figura 3.2. Promedios por pregunta/ítem de SUS	87

RESUMEN

El desarrollo de aplicaciones puede abordarse a través de distintas metodologías de desarrollo y gestión, algunas otorgando mejores resultados y mayor valor a los interesados que otras. Dos de estas metodologías, DevOps y Scrum, son algunas de las más utilizadas actualmente, cada una dentro de su ámbito de aplicación, y si bien se observan como unas de las mejores alternativas, cabe la pregunta, ¿qué tan seguros podemos estar de ello? Con ello en mente, el presente trabajo muestra la inclusión de ambas metodologías dentro del desarrollo de una aplicación para dispositivos móviles a través del uso de flujos DevOps seguros. Se presenta una vista acerca del uso de la metodología Scrum y su adaptación dentro del proyecto, destacando los resultados obtenidos a lo largo de su ejecución; así mismo, dentro de cada uno de los puntos en los que se ha dividido el proyecto se muestran los resultados conseguidos a partir del uso de los flujos DevOps que se han diseñado e implementado. Para finalizar y responder a la pregunta, se realiza una comparación de estos con lo que significa ejecutar un proyecto de las mismas características sin el uso de dichos flujos DevOps, así como un análisis introspectivo de su uso dentro de este proyecto y sus posibles resultados dentro de proyectos similares, así como con proyectos de diferentes características al ejecutado.

PALABRAS CLAVE: Flujo DevOps, Etapa, GitLab, Integración Continua, Despliegue Continuo.

ABSTRACT

Applications development can be executed through using many different development and management methodologies, some giving better results and greater value to stakeholders than others. Two of them, DevOps, and Scrum, are two of the most used methodologies currently, each one in its area of application, and even this two seem to be the best alternatives between all, a question arises, ¿how sure can we say that these are the best alternatives? With that in mind, this work shows the usage of these both methodologies in the process of developing an application for mobile devices with the use of secure DevOps flows. It presents a general view of the use of Scrum and its adaption in this project, and the results it gave; likewise, along every segment in which this process was divided are shown the results gotten from the use of secure DevOps flows that were designed and implemented previously. Before ending and to answer the question before, a comparison between these results and the execution of a similar project without the use of these DevOps flows is made, and so an introspective analysis of the use of these flows and how it could have been done in other projects with same characteristics, and so with others different from this.

KEYWORDS: DevOps flow, Stage, GitLab, Continuous Integration, Continuous Deployment.

1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO

El componente desarrollado es el módulo de Recaudos de la empresa Manticore Labs. Este componente abarca varias de las funciones y operaciones que un sistema de Recaudos contempla, tales como:

- Administración de información de cuentas externas.
- Administración de información de cuentas de la empresa.
- Administración de periodos contables.
- Administración de transacciones contables.

Cada una de ellas, se encuentra estructurada y ordenada por medio de una o más interfaces dentro de la aplicación.

1.1 Objetivo general

Desarrollo del módulo de Recaudos de la aplicación móvil para el sistema ERP de la empresa Manticore Labs mediante la implementación de Scrum y metodologías seguras de desarrollo a través de DevOps.

1.2 Objetivos específicos

1. Definir los flujos de DevOps a ser utilizados.
2. Implementar y probar los flujos de DevOps.
3. Implementar los flujos de DevOps a lo largo del ciclo de vida del desarrollo del proyecto de software.
4. Comparar los flujos DevOps utilizados con una implementación que no haga uso de dichos flujos.

1.3 Alcance

El alcance del componente Recaudos es la implementación de funciones de recaudos contables, además de flujos de soporte como gestión de cuentas y de informes, a través de metodología Scrum, como parte de un aplicativo móvil orientado a un sistema ERP, que permita la práctica de flujos de desarrollo DevOps seguros.

1.4 Marco teórico

Desde el inicio y posterior desarrollo de la tecnología, las empresas la han integrado como una parte vital de su crecimiento y de sus operaciones. Este desarrollo se ha ejecutado en distintos ámbitos, los más notables para las personas, aquellos que se refieren a las capacidades de Hardware cuyo impacto es notable en la velocidad de procesamiento o en las capacidades de almacenamiento, como puntos más impactantes para el común. Sin embargo, otro de los cambios importantes que se han ejecutado desde sus inicios, conforme la práctica de desarrollo de Software dejó de ser un trabajo artesanal para pasar a ser un proceso industrial, definido de forma diferente por cada nueva metodología o método de desarrollo de Software, pero todos destinados a la creación de un producto de calidad [1].

1.4.1 Enterprise Resource Planning (ERP)

Las empresas, desde la inclusión de la tecnología en el desarrollo de las empresas, la han utilizado como base para el desarrollo de algunas actividades: administración de productos, contacto con el cliente, recurso de información, administración de servicios, entre algunos otros. Si bien las empresas solían utilizar estas herramientas de a poco, una o dos como mucho, con el pasar del tiempo se empezó a definir un concepto que reúna a estos servicios de forma íntegra, para describir varios aspectos que se consideraban necesarios administrar dentro de una organización empresarial: aquí es donde nace el concepto de la Planeación de Recursos Empresariales o ERP, por sus siglas en inglés, al menos desde el punto de vista de las tecnologías computacionales [2].

Así, un ERP se define como un sistema de software que permite a las organizaciones administrar varios procesos y servicios. Para administrar estos servicios este sistema trabaja de forma modular, tal que cada uno de ellos se centra en el área de negocio de interés [1].

Este tipo de sistemas no se encuentra definido de una forma exacta, tanto los módulos que se encuentran dentro del sistema, así como la propia estructura de lo que contiene internamente cada módulo, puede y variará de acuerdo con las necesidades y capacidades de la organización que la implemente. En ese sentido, la

empresa SAP (Systemanalyse Programmentwicklung) [1] define algunos módulos comunes dentro de dichos sistemas, así:

- Finanzas.
- Abastecimiento y Compras.
- Gestión de Recursos Humanos.
- Ventas.
- Fabricación.
- Cadena de Suministro.
- Servicios.
- Ingeniería de Información más Desarrollo.
- Gestión de Activos Empresariales [1].

1.4.2 SCRUM

Si bien se han desarrollado varias formas de desarrollar software de una forma bien estructurada y metódica, una de las que más impacto han tenido es Scrum. Scrum es, a grandes rasgos, un marco de trabajo para ejecución de proyectos de Software, basado en un desarrollo guiado por una gestión “administrativa” del proyecto, tanto desde sus fases iniciales, hasta incluso después que el producto final ha sido entregado [3].

Scrum, se centra en 3 aspectos que lo consideran fundamentales para la consecución de los objetivos:

- Los Roles de Scrum.
- Los Artefactos de Scrum.
- Los Eventos de Scrum [3].

Roles de Scrum

Dentro de Scrum se define una unidad básica de trabajo para una correcta ejecución del proyecto sobre el cuál se encuentra; esta se conoce como el Equipo de Scrum, y dentro de este equipo se definen tres roles básicos:

- Product Owner.
- Scrum Master.
- Desarrollador [3].

El Product Owner es aquella persona o grupo de personas interesadas y encargadas de garantizar que el producto final contará con todas las especificaciones de usabilidad, calidad y operabilidad requeridas por los interesados [3].

El Scrum Master es aquella persona que está encargada de administrar el proyecto, es aquella persona encargada de guiar el camino del ciclo de vida de desarrollo del producto final, así como al resto del Equipo de Scrum [3].

En tanto, el Desarrollador o grupo de desarrolladores son todas aquellas personas centradas en la codificación de cada aspecto del producto, tanto de sus funcionalidades como de cualquier otra parte del sistema, en otras palabras, es la común imagen que se tiene del desarrollador [3].

Eventos de Scrum

El segundo aspecto considerado se refiere a los eventos, que no son más que un instante o un intervalo de tiempo definido para la ejecución de actividades necesarias para el éxito del proyecto. En ese sentido, Scrum define cinco eventos:

- Sprint.
- Sprint Planning.
- Daily Scrum.
- Sprint Review.
- Sprint Retrospective [3].

El Sprint se define como un intervalo de tiempo, generalmente de un periodo de máximo 4 semanas, en el cual se ejecutan todos los demás eventos de Scrum. En ese sentido, el Sprint enmarca cada intervalo de tiempo en el que se ejecuta el desarrollo y la administración del proyecto [3].

El Sprint Planning, como su nombre indica, representa la planificación de cada uno de los Sprints, de modo que se pueda describir, dentro de ese intervalo de tiempo reducido, las actividades a ser completadas; aquí, se trabaja lo que se conoce el Sprint Backlog [3].

El Daily Scrum, es una pequeña reunión que se ejecuta entre los integrantes del Equipo de Scrum, donde se evalúan los resultados obtenidos cada día, de ese modo, existe un monitoreo diario de los avances y problemas que podría tener el equipo para cumplir con su trabajo [3].

El Sprint Review permite al Equipo de Scrum evaluar el progreso obtenido a lo largo de un Sprint, se enfoca principalmente en la evaluación del producto obtenido dentro de dicho Sprint, respecto a lo planificado en el Sprint Planning [3].

El Sprint Retrospective también ejecuta una evaluación dentro del Sprint, sin embargo, aquí no se enfoca en el producto, sino más bien en el equipo y en su desempeño; aborda una aproximación más administrativa y funcional [3].

Artefactos de Scrum

El segundo aspecto considerado por Scrum son los artefactos, que no son más que representaciones de productos, documentos o estados del desarrollo a los cuales se apunta en un instante determinado. De ese modo, se definen tres artefactos utilizados por Scrum:

- Product Backlog.
- Sprint Backlog.
- Incremento [3].

El Product Backlog representa el conjunto de requerimientos totales necesarios para poder completar todos los detalles del producto final. Es decir, representa el conjunto de requisitos para cumplir con las expectativas del producto del cliente [3].

El Sprint Backlog hace referencia al subconjunto de requerimientos planteados y propuestos a ser cumplido dentro de un Sprint, de modo que puedan plantearse entregas o incrementos hacia el usuario o los interesados, es decir, funciones parciales del producto que representarían valor al cliente [3].

El incremento representa la porción del producto final que ha sido desarrollada hasta un punto en específico, estos incrementos, no necesariamente se entregan exclusivamente al final de un Sprint, sino, un incremento puede ser entregado cada vez que se obtenga algo de valor a lo largo del desarrollo [3].

1.4.3 DevOps

Siguiendo el tema del primer párrafo dentro de esta sección, sabemos que la tecnología se ha vuelto una parte vital dentro de muchos ámbitos, y eso incluye a las empresas y organizaciones, independientemente de su relación directa con la

tecnología. Temas como el marketing, la publicidad o la propia administración del negocio se vuelve una parte importante de su vida empresarial [4].

Sin embargo, muchos de estos avances se centran más bien en las tecnologías de desarrollo de los productos, antes que en las formas en que se administran o se aplican dichas tecnologías. Más allá de la metodología de Scrum, que se centra más bien en la administración del proyecto como tal, también existen metodologías que se centran exclusivamente en cómo se ejecuta el desarrollo del producto final; una de ellas es DevOps [5].

DevOps nace de la combinación de las palabras Development (Dev) y Operations (Ops), es decir, Desarrollo y Operaciones. Este se centra en un conjunto de prácticas que pretenden integrar a los desarrolladores, así como las operaciones, que representan a las actividades que debe cumplir el sistema de software que quiere ejecutar una empresa [5].

Describiéndolo en pocas palabras, DevOps intenta que el desarrollo enfoque los esfuerzos a que las empresas tengan un producto o un porcentaje de él que represente un valor a la empresa, principalmente, como una forma de ser usado para generar mayores ingresos o disminuir los esfuerzos o egresos por costos que se pueden reemplazar por las máquinas [4].

Para que estas prácticas funcionen, se han definido 3 puntos importantes que sirven dentro del desarrollo:

- Integración Continua.
- Entrega Continua [5].

Integración continua

La Integración Continua se basa en la integración del trabajo de cada equipo frecuentemente. Dentro de un equipo de trabajo, es muy común y casi necesario que el trabajo sea repartido entre los integrantes del equipo de desarrollo, cada uno haciendo su trabajo en un dispositivo distinto. De ese modo, que el estado actual del producto que está siendo desarrollado se encuentra actualizado y perfectamente funcional, es muy importante para no encontrar posibles errores o fallas en el futuro. De tal manera, la Integración Continua garantiza que este proceso frecuente y automático permite verificar el estado de la aplicación en tiempo real, para poder ejecutar posibles cambios o similares [5].

Entrega Continua

La Entrega Continua se basa en los mismos principios que la Integración Continua, pero orientado un poco más a la parte de las pruebas de la aplicación. Usualmente, mientras la Integración Continua se ejecuta durante una fase de Construcción de la aplicación, la Entrega Continua se ejecuta durante la fase de Pruebas de la aplicación. Así, la Entrega Continua se basa en que, apenas se haya terminado una porción de la aplicación que se considere funcional, pase a una fase donde se puedan ejecutar las pruebas necesarias para comprobar que su funcionamiento y su cumplimiento con los requerimientos es absoluto [5].

1.5 Herramientas de Desarrollo

1.5.1 GitLab

GitLab es una plataforma orientada al desarrollado por medio de la metodología DevOps. Dicha plataforma ofrece a través de la nube las necesidades que DevOps necesita para funcionar, principalmente, aquellas relacionadas con la Integración Continua y le Entrega Continua [6].

En relación con el trabajo que será mostrado en la sección de Metodología, es necesario describir algunas de las herramientas a ser usadas en el presente trabajo para comprender mejor su ejecución:

- GitLab CI/CD.
- GitLab Runners.
- GitLab Repositories [6].

GitLab CI/CD

Basado en las definiciones de Integración Continua y Entrega Continua (CI/CD) vistas anteriormente, GitLab CI/CD ofrece las capacidades para poder ejecutar dichos procesos de forma automática, así, permite automatizar los procesos de Construcción, Pruebas, Lanzamiento y Monitorización de cada producto o porción de producto que haya sido entregada a este portal [7].

GitLab Runners

La Ejecución de los procesos automatizados dentro de CI/CD debe cumplirse en algún sistema, es decir, debe haber por lo menos un sistema operativo que permita su ejecución de por lo menos bajo nivel, según sea el producto que se desarrolle. Así, estos Runners permiten la ejecución de dichos procesos, sea dentro de máquinas emuladas ofrecidas por el propio GitLab, así como en máquinas que puedan configurarse por el propio cliente, a través de contenedores, máquinas virtuales o semejantes [8].

GitLab Repositories

CI/CD se debe ejecutar sobre una aplicación, y dicha aplicación debe describirse en uno o más archivos que lo representen. Por tal motivo, estos archivos necesitan un lugar sobre el cuál reposar dentro del servicio, que les permita su acceso. GitLab Repositories representa una especie de contenedor que es capaz de almacenar distintos tipos de archivos, principalmente, de programación o configuración, que puedan ser accedidos por otros desarrolladores o por otras funciones de la plataforma [9].

1.5.2 Git

Para entender qué es Git, es necesario conocer primero acerca del Control de Versiones. Este término hace referencia a cualquier sistema que es capaz de almacenar y registrar el historial de cambios, creación, eliminación o ajustes realizados sobre uno o más archivos dentro de un repositorio [10].

Con eso en mente, Git es justamente un Sistema de Control de Versiones. Esta herramienta permite trabajar con el Control de Versiones no solamente dentro de un propio computador, sino, dada la creciente necesidad de las empresas para trabajar de forma remota, también permite ejecutarlo bajo ese mismo concepto [10].

Con eso en mente, es necesario conocer algunos conceptos importantes necesarios para entender cómo funciona Git, que serán presentados en tres temas:

- Repositorio
- Ramas.
- Operaciones [10].

Repositorio

Un repositorio se puede entender como una especie de carpeta, la cual permite almacenar distintos archivos que serán posteriormente manipulados de alguna forma, en el contexto de Git, sirve como la carpeta base donde se ubican todos los archivos de los cuales se quiere mantener y controlar el historial de versiones [11].

Ramas

Las ramas hacen referencia a la opción de la herramienta de otorgar las capacidades de crear distintas líneas de cambios y configuración sobre los archivos. De ese modo, a partir de un mismo conjunto inicial de archivos, es posible que distintas personas ejecuten una porción de ellos sin necesidad de afectar el trabajo de los demás [12].

Operaciones

Dada la capacidad de Git para trabajar de forma local como de forma remota. Se definen algunas operaciones para poder describir el flujo que una persona podría utilizar para trabajar, así, principalmente, se tienen tres conceptos.

- Pull.
- Push.
- Commit [13].

La operación Pull permite extraer el contenido de un repositorio ubicado de forma remota hacia una carpeta personal, es utilizada principalmente para mantenerse actualizado con las versiones actuales de los archivos del repositorio, así como para extraer una versión que no modifique la dirección del proyecto principal [13].

La operación Push permite enviar los cambios que se desean actualizar sobre el repositorio ubicado de forma remota desde el computador, de ese modo, se puede enviar la información que actualice dicho repositorio [13].

La operación Commit, es un poco más compleja que las anteriores, pues este, ejecuta un proceso parecido a la operación Push, sin embargo, lo que hace es subir los cambios al repositorio local creado, esto sirve, para describir únicamente los archivos que en efecto se desean actualizar, pues, es posible que de todos los documentos que se modificaron de forma local, solo uno de ellos requiera una actualización posterior hacia el repositorio remoto [13].

1.5.3 Figma

Es una herramienta que permite el trabajo de forma colaborativa dentro del diseño de cualquier tipo de figuras o documentos, cuyo principal fuerte es la capacidad de ser operada dentro de un navegador web. Sus capacidades permiten ejecutar el diseño de una gran variedad de objetos, siendo uno de ellos, el desarrollo de interfaces de un programa o aplicación de Software, o al menos uno de los más conocidos [14].

1.5.4 Firebase

Dentro de la variedad de herramientas que ofrece Google para los desarrolladores, existe que se ha especializado principalmente para ofrecer distintos servicios y ayudas a quienes se dedican al desarrollo de aplicaciones móviles, principalmente para el sistema Operativo Android, una de estas herramientas es Firebase [15].

Firebase es una plataforma que otorga múltiples utilidades y funcionalidades para el desarrollo de aplicaciones. Desde bases de datos hasta la capacidad de ejecutar pruebas sobre la aplicación, Firebase ofrece varias opciones a los desarrolladores para ayudar durante el ciclo de vida de desarrollo de su aplicación. En ese sentido, y en relación con el presente trabajo, se utilizarán dos de ellas [15].

Cloud Firestore

Cloud Firestore es, en simples palabras, una base de datos de tipo NoSQL, es decir, una base de datos que no almacena la información de una forma estrictamente estructurada, sino que otorga al desarrollador la capacidad de almacenar su información de acuerdo con las necesidades del negocio [16].

Dentro de la estructura de Cloud Firestore, la información es guardada a través de una estructura de tipo jerárquica, donde:

- Lo más bajo son los datos, que representan en la base de datos la información que se desea almacenar.
- Cada uno de los datos es a su vez almacenado dentro de un documento, que representa la entidad a la cuál cada uno de esos conjuntos de datos pertenecen.
- Y en lo más alto, se encuentran las colecciones, que no son más que el conjunto de varios documentos, que permite almacenar a aquellos que, desde el punto de vista de desarrollo, poseen características similares [17].

Firebas App-Distribution

Permite a los desarrolladores distribuir o lanzar una aplicación de una forma más fácil y sencilla. En el caso de este proyecto, se ha utilizado esta capacidad para la distribución de una aplicación móvil creada para el sistema operativo Android, de modo que se ejecuta un lanzamiento automático del producto ejecutable de dicha aplicación hacia un repositorio [18].

1.5.5 Docker

Docker es una plataforma de libre uso cuyo principal fin es el de permitir a usuario desarrollar y ejecutar aplicaciones dentro de un entorno virtual emulado y aislado de los demás activos existentes dentro del computador [19].

Docker, permite crear, administrar y ejecutar servicios en entornos dentro del computador, pero aislados de los recursos de este. Esto lo hace por medio de dos recursos, así:

- Imágenes.
- Contenedores [19].

Imagen

Una imagen se puede contemplar como almacenar la información de un sistema operativo o una aplicación en un instante determinado dentro de un recurso, en este caso, un archivo. Esta imagen, a grandes rasgos, guarda todos los archivos que hacen referencia a dicho sistema o aplicación, de modo que pueda ser instalado y usado en otros lugares con las herramientas necesarias. [19]

En el caso de Docker, estas imágenes se ejecutan en lo que se conocen como contenedores.

Contenedor

Un contenedor hace referencia al espacio creado para la creación y ejecución de una imagen. Estos contenedores pueden ser vistos como un pequeño computador que está ejecutándose dentro del computador, pero aislado de los demás archivos de Software del computador local [19].

La principal característica de los contenedores es la capacidad de permitir ejecutar otras aplicaciones dentro de un entorno controlado que no interrumpa ni interfiera con el entorno que se quiere proteger del computador local, usualmente, durante la ejecución de pruebas sobre dichas aplicaciones, que podrían causar daños sobre la máquina, y que con el uso de estos contenedores son aislados [19].

1.5.6 Android Studio

Tal y como existen muchos ambientes de desarrollo dentro del mercado para la escritura de código de forma convencional, sin embargo, también existen varios otros que han sido desarrollados para especializarse dentro del desarrollo de aplicaciones dentro de un ambiente o con un propósito en específico, dentro de ellas se encuentra Android Studio, que, como su nombre nos da una pista, se centra en ayudar al desarrollador a la creación de proyectos de aplicaciones para dicho dispositivos móviles basados en sistemas operativos Android [20].

De ese modo, Android Studio no es más que un ambiente de desarrollo, basado en el ambiente IntelliJ IDEA, que está orientado al desarrollo de aplicaciones para dispositivos móviles bajo sistemas operativos Android, ofreciendo una gran variedad de características cuyo fin es facilitar el trabajo a los desarrolladores en configurar e integrar su proyecto hacia dichos dispositivos [20].

1.6 Lenguajes de Programación

1.6.1 Kotlin

Kotlin es un lenguaje de programación de código abierto orientado a la escritura de código de desarrollo de aplicaciones móviles, principalmente, destinado para el desarrollo dentro del ambiente Android Studio. A pesar de ser un nuevo lenguaje, se ve potenciado gracias a tomar varias de las mejores características de otros lenguajes tales como C# o JAVA, principalmente respecto a su sintaxis y funciones internas, así como su capacidad de trabajar con Programación Funcional o Programación Orientada a Objetos, entre otros [21].

1.6.2 XML

XML, o Extensible Markup Language por sus siglas en inglés, es una especie de lenguaje que sirve para describir formatos de intercambio de información entre distintas aplicaciones. A través de XML, se puede definir un formato constante para transmitir información, incluso ejecutable, independientemente de la plataforma sobre la cual repose el documento [22].

1.6.3 YAML

YAML, o Yet Another Markup Language por sus siglas en inglés, se trata de un lenguaje de serialización de datos, cuyo principal fin es el de permitir la configuración de distintos entornos capaces de entender su sintaxis. Con ello, esta configuración es capaz de describir al entorno donde podría ejecutarse los requerimientos, las opciones y los comandos que debe ejecutar para conformarse de acuerdo con lo que el usuario a descrito en él [23].

2 METODOLOGÍA

Por un lado, la implementación del módulo como parte de un aplicativo móvil se ejecutó a través de la metodología DevOps, contando como eje principal la integración continua. En tanto, la gestión del proyecto de desarrollo se ejecutó a través del uso de artefactos, roles y eventos dispuestos por la metodología Scrum.

Con esto en mente, es preciso definir cómo se implementaron dichas metodologías dentro del desarrollo del proyecto, pues tanto el alcance del proyecto como el equipo de desarrollo definido para su implementación limitan en un cierto porcentaje el uso de todas las prácticas y guías que estas metodologías ofrecen.

2.1 Implementación de Scrum

La implementación de Scrum, respecto a su definición completa usualmente, fue disminuida en ciertos aspectos, de modo que sea adaptable a un equipo de desarrollo conformado por una sola persona, la cual dificultaría el desarrollo de sus actividades si quisiera encargarse de cada uno de los aspectos que Scrum define para ser ejecutados entre más de una persona.

Tomando en cuenta los roles descritos en el capítulo 1.4.2, en la siguiente tabla se describen los roles que fueron parte del proyecto, la persona que cumplió dicho rol y su función dentro de la ejecución.

Tabla 2.1.1. Roles de Scrum implementados

Rol	Persona	Función
Scrum Master	Adrián Egüez	Como director del proyecto, definiendo y evaluando los avances necesarios para el cumplimiento de este.
Product Owner	Adrián Egüez	Como revisor, asegurando que el producto final otorgue un verdadero valor al cliente, en este caso, la empresa Manticore Labs.
Desarrollador	Wilman Armijos	Encargado de la ejecución de los distintos trabajos necesarios para el cumplimiento del proyecto: análisis, diseño, implementación,

		pruebas y despliegue de los distintos productos obtenidos.
--	--	--

De la tabla anterior, es importante destacar dos cosas:

- Por un lado, el hecho que los roles de Scrum Master y Product Owner estuvieron a cargo de la misma persona que, además, podría ser vista como un representante de la empresa cliente, al ser también parte de esta. Este hecho provoca que los eventos ejecutados se desarrollen de una forma más ágil pues implican que tanto los resultados de la ejecución a nivel administrativo del proyecto, como de los avances en términos del producto, son analizados y revisados por el mismo individuo; esto, permitiendo a los dos enfoques administrativos y de desarrollo apuntar más fácilmente en la misma dirección.
- El otro punto por destacar es la definición del equipo de desarrollo, dado que en este caso particular dicho equipo está conformado por una sola persona. Esto es un factor clave dentro de ambos flujos, esto principalmente por la naturaleza de ambas metodologías a ser destinadas a equipos de trabajos múltiples, donde factores como la colaboración y la interacción entre integrantes del equipo es fundamental.

Retomando la descripción de los ítems de Scrum, tomando como referencia los artefactos del capítulo 1.4.2, en la siguiente tabla se describen aquellos utilizados y su función dentro del presente trabajo.

Tabla 2.1.2. Artefactos de Scrum implementados

Artefacto	Descripción
Product Backlog	Descrito a través de 5 historias de usuario y sus escenarios, que permite la implementación de interfaces y de sus funcionalidades, según los requerimientos.
Sprint Backlog	A través de las mismas historias de usuario, a cada ejecución de un Sprint se seleccionan las historias de usuario a ejecutarse dentro del Sprint.
Incremento	Representado en porciones funcionales del producto final dentro del repositorio.

Por último, se tienen los eventos de Scrum, y es este punto el que muestra mayores diferencias respecto a los mostrados dentro del capítulo 1.4.2, Scrum define 5 eventos a ser ejecutados, de los cuales, se han decidido tomar 3 de ellos debido a

las características del proyecto, los cuales fueron vistos como los más necesarios y acordes a dichas características. En la siguiente tabla se muestran dichos eventos y la forma en que fueron adaptados.

Tabla 2.1.3. Eventos de Scrum implementados

Artefacto	Descripción
Sprint	Fueron definidos con una duración de 1 semana, a lo largo de las 6 semanas de implementación del módulo, dentro de los cuales se ejecutó principalmente la fase de diseño, implementación, pruebas y despliegue de la aplicación.
Sprint Planning	Ejecutado de forma informal al inicio de cada Sprint, donde se definieron objetivos como resultado de la revisión del anterior Sprint.
Sprint Review	Ejecutado de forma semiformal al final de cada Sprint, ejecutado a modo de reporte enviado al Product Owner/Scrum Master respondiendo 3 preguntas: <ul style="list-style-type: none"> - ¿Qué se hizo en este Sprint? - ¿Qué inconvenientes encontré en este Sprint? - ¿Qué se hará en el siguiente Sprint?

2.2 Implementación de DevOps

La implementación de la metodología DevOps se centró en una de sus características básicas como lo es la Integración Continua. Para conseguirlo, se utilizó algunas herramientas que, en conjunto, permitieron ejecutar el flujo DevOps de una forma fácil y rápida.

Para ello, la siguiente imagen que describe a grandes rasgos el flujo de actividades y trabajo diseñado dentro del trabajo realizado.

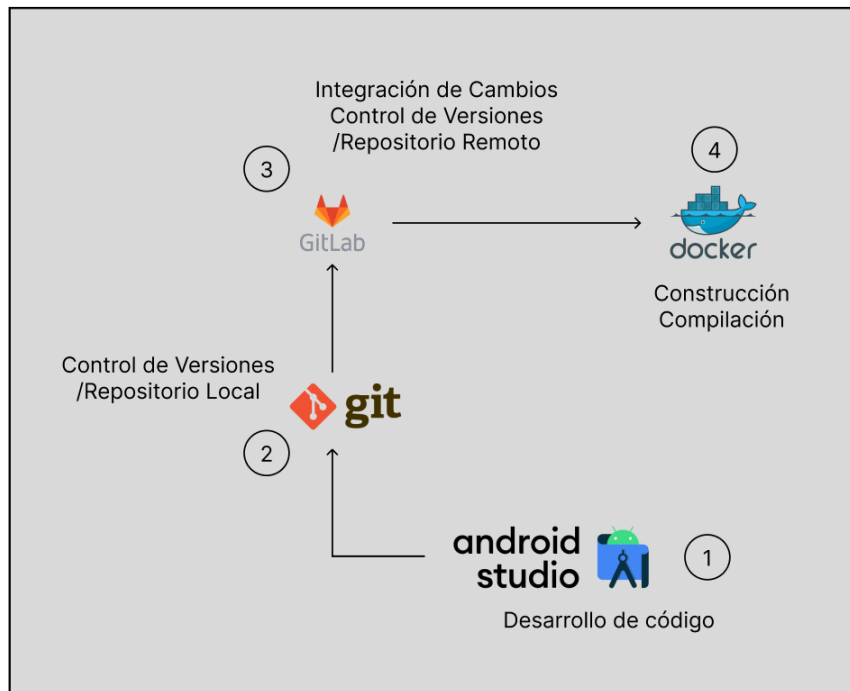


Figura 2.2.1. Diagrama de ejecución de DevOps

La figura 2.2.1 representa el flujo general del uso de DevOps dentro del trabajo. Para ello, se definen 4 puntos importantes, cuyo propósito será descrito de forma general a continuación.

Diagrama: Punto 1

El primer punto, y el más fácil de entender, representa el entorno de desarrollo de la aplicación. En este caso, se usó el entorno Android Studio, con ayuda del lenguaje de programación Kotlin, para la implementación del aplicativo móvil soportado por dispositivos sobre sistemas operativos Android.

Además, se realizó la integración de algunas herramientas otorgadas por Google Firebase como soporte al desarrollo y despliegue de la aplicación, así:

- Firebase Cloud Firestore: utilizada como la base de datos que almacenará la información que sea procesada dentro de la aplicación.
- Firebase App-Distribution: utilizada como una herramienta que facilite el despliegue y la distribución de la aplicación.

Diagrama: Punto 2

Como segundo punto, se tiene el uso de Git, para poder trabajar con los repositorios que almacenarán los archivos parte del proyecto de forma local y de forma remota.

Diagrama: Punto 3

Como tercer punto, se tiene la herramienta GitLab, que servirá como repositorio del producto final (ver Anexo I), así como la herramienta que permite la ejecución de la Integración Continua necesaria para la ejecución del Flujo DevOps.

El repositorio cuenta con 3 ramas, de las cuales las ramas “Development” y “Testing” son sobre las que se centraron los flujos.

En la rama “Development”, se ejecutaron tareas de compilación del proyecto de Android Studio cuyo resultado se veía reflejado en un “Android Application Package” o APK, que es el ejecutable de la aplicación.

En tanto, dentro de la rama “Testing”, además de ejecutar estas tareas de compilación descritas previamente para la rama “Development”, se ejecuta una tarea extra que consiste en distribuir dicho APK hacia otros usuarios, que puede ser utilizado para ejecutar pruebas de uso o para distribuir el producto final.

Diagrama: Punto 4

Como cuarto y último punto, se tiene la herramienta Docker, esta herramienta será utilizada como soporte para la ejecución de tareas descritas dentro de GitLab como parte del flujo de trabajo.

En este caso particular, Docker cuenta con un sistema operativo basado en Linux dentro de sí, el cual cuenta con las herramientas necesarias para la compilación de un proyecto de Android Studio, así como con la descripción de instrucciones necesarias para su compilación y la distribución automática del producto.

Una vez conociendo el propósito general de cada uno de los componentes del flujo DevOps, es necesario conocer cómo se hicieron las configuraciones o tareas necesarias para incluir dichos componentes de forma práctica dentro del proyecto, lo cual será descrito dentro del capítulo 2.2.1.

2.2.1 Configuraciones del Flujo DevOps

Las configuraciones necesarias para trabajar con el Flujo DevOps nacen a partir del Punto 2 que se encuentra en la figura 2.2.1, es decir, dentro de Git.

Para que Git funcione, hay que contar con un repositorio remoto ya existente, con el cuál podamos decirle a la herramienta donde queremos que se guarden los cambios, el cual se encuentra dentro del Anexo I, que se observa en la figura 2.2.1.1.

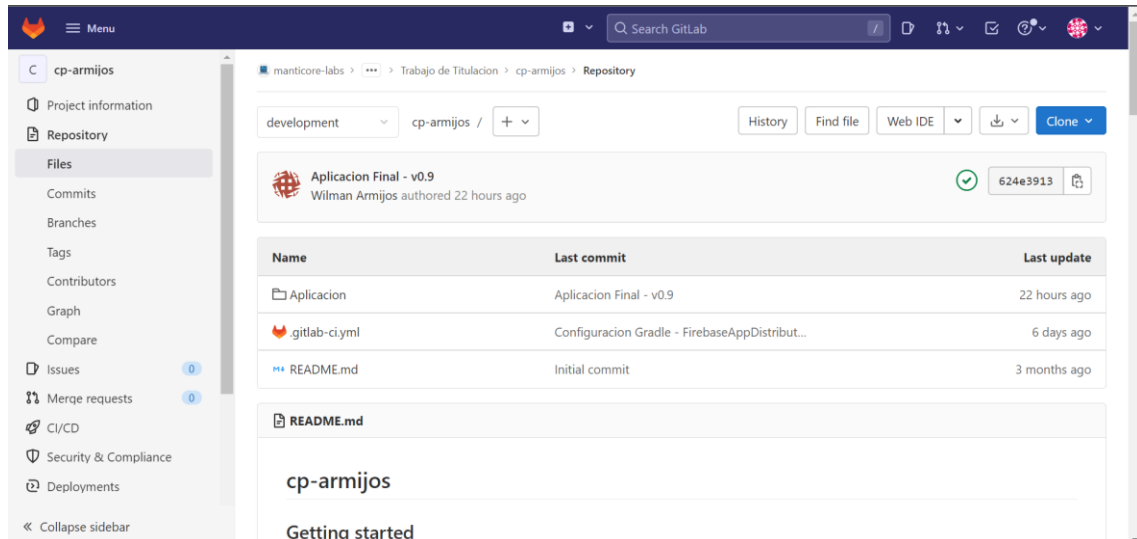


Figura 2.2.1.1 Repositorio remoto en GitLab

Con el repositorio remoto ya existente, necesitamos conectarlo a algún “Repositorio Local”, es decir, en alguna ubicación dentro de nuestro computador, una carpeta. En el caso particular, se utilizó una carpeta con el nombre “Proyecto”, tal como se observa en la figura 2.2.1.2.

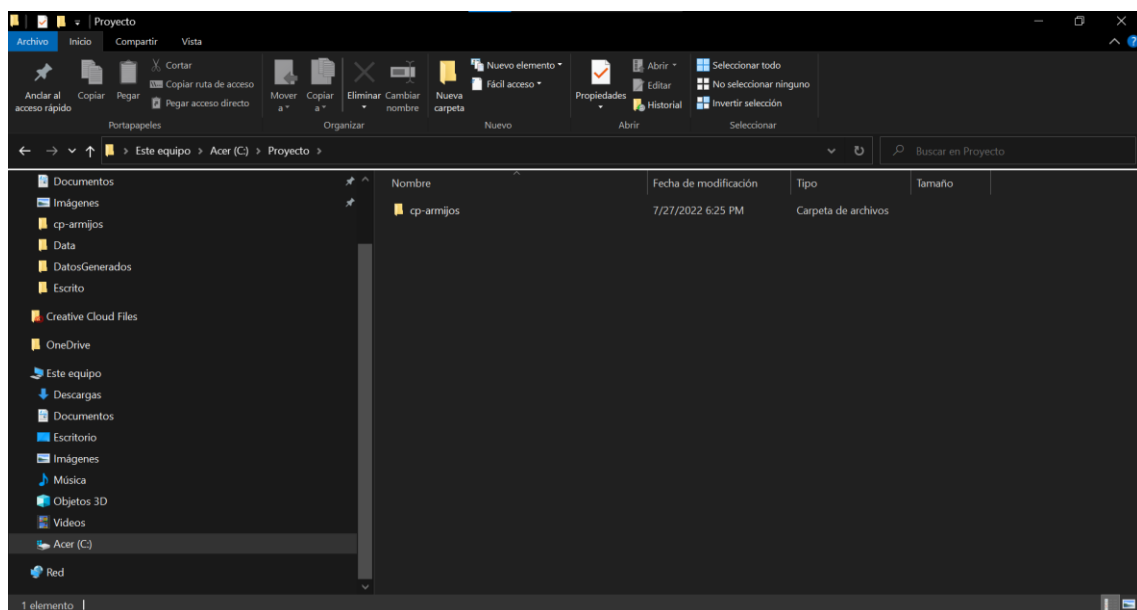
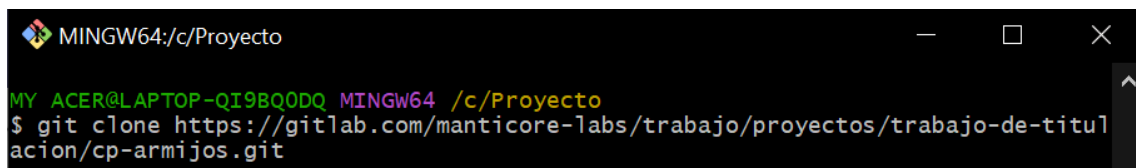


Figura 2.2.1.2 Carpeta objetivo como repositorio local

Con estos dos repositorios definidos, solo queda usar la línea de comandos para ejecutar las instrucciones necesarias para enlazar ambos repositorios.

Para ello, se hizo uso de la herramienta Git Bash, que es una interfaz de línea de comandos ofrecida por la propia herramienta Git.

Lo primero que se debe hacer, es traer el repositorio de GitLab hacia nuestro computador, para lo cual, se ejecutó el comando que se muestra en la figura 2.2.1.3 dentro de la carpeta destino, es decir, "Proyecto".

A screenshot of a terminal window titled "MINGW64:/c/Proyecto". The prompt is "MY ACER@LAPTOP-QI9BQ0DQ MINGW64 /c/Proyecto". The command entered is "\$ git clone https://gitlab.com/manticore-labs/trabajo/proyectos/trabajo-de-titulacion/cp-armijos.git". The output is not visible, only the command and prompt are shown.

```
MINGW64:/c/Proyecto
MY ACER@LAPTOP-QI9BQ0DQ MINGW64 /c/Proyecto
$ git clone https://gitlab.com/manticore-labs/trabajo/proyectos/trabajo-de-titulacion/cp-armijos.git
```

Figura 2.2.1.3 Clonación de proyecto a repositorio local

En este punto, nos pedirá que ingresemos la cuenta de GitLab con la cuál queremos enlazar al repositorio, como parte del sistema de autenticación y autorización de GitLab. Aquí, simplemente se ingresan las credenciales y se puede seguir con el proceso.

Con los pasos descritos, fue configurada la conexión entre los repositorios local y remoto, y con ello será posible ejecutar las operaciones del flujo, que serán descritas en el punto 2.2.1.

Lo siguiente necesario por configurar es el punto 3 dentro de la figura 2.2.1, es decir, el entorno de GitLab. Si bien, ya se contaba con el repositorio y sus distintas ramas, ahora es necesario configurar algunos archivos y opciones necesarias para la ejecución de las tareas del flujo DevOps.

Para ello, lo primero que se configuró fue los Runners de GitLab (ver punto 1.5.1). Por defecto, GitLab ofrece contenedores que se ejecutan de forma compartida, seguramente dentro de servidores o computadores en sus instalaciones. Sin embargo, estos pueden presentar problemas en la velocidad de ejecución de las tareas, para lo cual, otra alternativa es hacer uso de contenedores propios, que, aquí fueron ejecutados sobre la herramienta Docker (Punto 4 de la figura 2.2.1).

Entonces, la configuración de estos Runners fue cambiada dentro de la página de GitLab, para ello, dentro las configuraciones de CI/CD, en el apartado de Runners, se deshabilitó la opción “Enable shared runners for this project”, tal y como se observa en la figura 2.2.1.4.

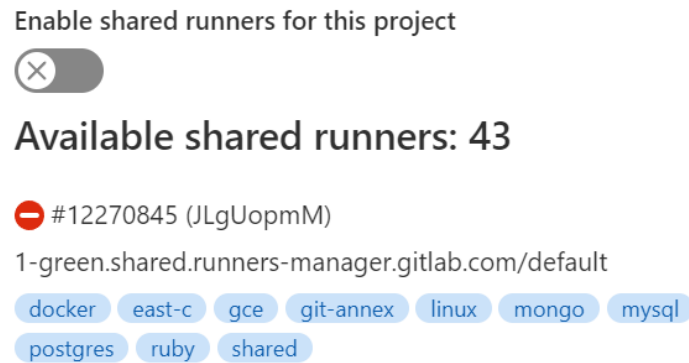


Figura 2.2.1.4 Desactivación de opción “Enable shared runners for this project”

Una vez deshabilitada dicha opción, ahora fue necesario decirle a GitLab cuál será el Runner que yo le daré para que ejecute las acciones. Para ello, y con la herramienta GitLab Runners previamente instalada, vamos a la carpeta donde se encuentre el archivo descargado y lo ejecutamos en modo de registro de un nuevo Runner, de la forma que se observa en la figura 2.2.1.5.



Figura 2.2.1.5 Registro de un nuevo Runner de forma local

Una vez hecho eso, nos pedirá algunos datos necesarios para el registro del nuevo contenedor. Primero, nos pedirá la URL de la instancia de GitLab, para lo cual, enviamos la URL recomendada por la propia aplicación, tal como en la figura 2.2.1.6.



Figura 2.2.1.6 Definición de la URL del Runner local

A continuación, se nos pedirá un Token para el registro del nuevo Runner, para ello, dentro del menú descrito anteriormente, existe una sección llamada Specific Runners, la cual describe, además de los pasos para crear un Runner propietario, el Token necesario para su registro, esto se observa en la figura 2.2.1.7.

Specific runners

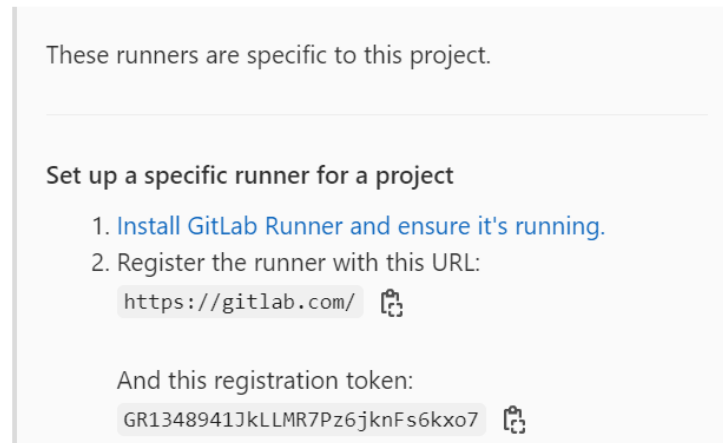


Figura 2.2.1.7 Definición del Token dentro de GitLab

Con el Token de GitLab, ya podemos registrarlo dentro del nuevo Runner, tal y como en la figura 2.2.1.8.



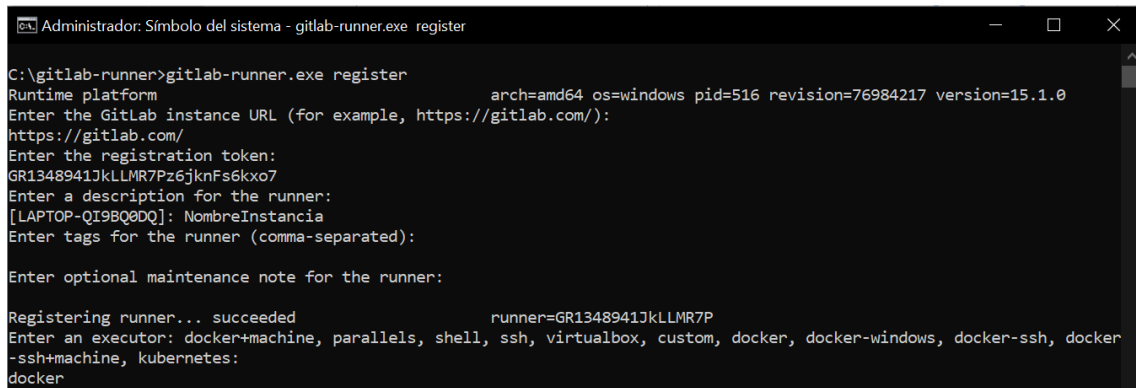
Figura 2.2.1.8 Definición del Token del Runner local

A continuación, se nos pedirá una descripción para el Runner, que en otras palabras es el nombre del Runner o la identificación con la que podremos encontrarlo posteriormente dentro de GitLab, en este caso, se utilizó el nombre del computador local por defecto, de forma similar a como se observar en la figura 2.2.1.9.



Figura 2.2.1.9 Definición de la Descripción del Runner local

A continuación, pedirá algunos campos como etiquetas y notas opcionales, las cuales serán omitidas y mantenidas en sus valores por defecto. Después de estos, se debe definir el ejecutor del Runner, y aquí es donde entra en juego Docker, esto se observa en la figura 2.2.1.10.



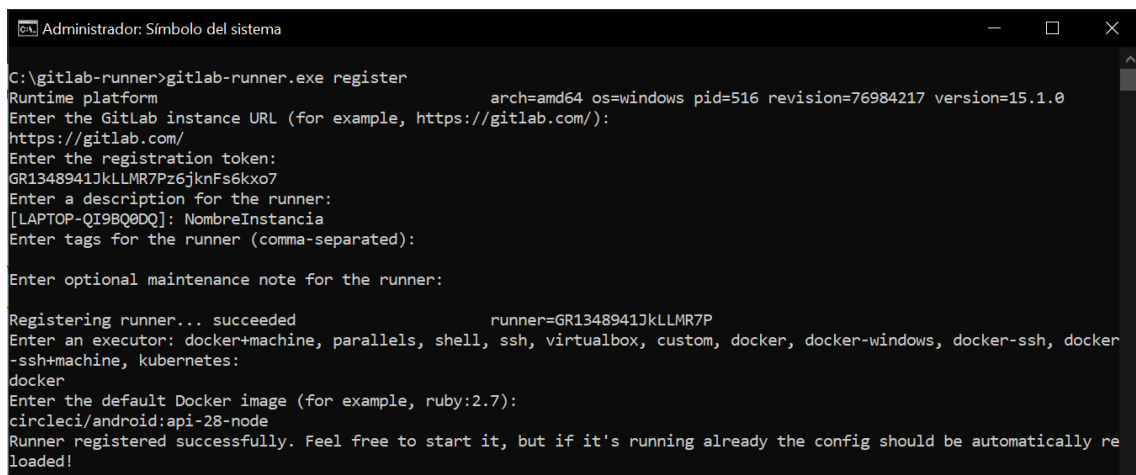
```
Administrador: Símbolo del sistema - gitlab-runner.exe register
C:\gitlab-runner>gitlab-runner.exe register
Runtime platform arch=amd64 os=windows pid=516 revision=76984217 version=15.1.0
Enter the GitLab instance URL (for example, https://gitlab.com/):
https://gitlab.com/
Enter the registration token:
GR1348941JkLLMR7Pz6jknFs6kxo7
Enter a description for the runner:
[LAPTOP-QI9BQ0DQ]: NombreInstancia
Enter tags for the runner (comma-separated):

Enter optional maintenance note for the runner:

Registering runner... succeeded runner=GR1348941JkLLMR7P
Enter an executor: docker+machine, parallels, shell, ssh, virtualbox, custom, docker, docker-windows, docker-ssh, docker-ssh+machine, kubernetes:
docker
```

Figura 2.2.1.10 Definición del Ejecutor del Runner local

Habiéndose establecido Docker como el ejecutor, ahora se debe especificar la imagen que trabajará como sistema operativo sobre el cuál se desarrollarán las actividades de DevOps, para esto, se utilizará la imagen CircleCI/Android [24], que cuenta con varias configuraciones predefinidas para proyectos de Android, tal y como se muestra en la figura 2.2.1.11.



```
Administrador: Símbolo del sistema
C:\gitlab-runner>gitlab-runner.exe register
Runtime platform arch=amd64 os=windows pid=516 revision=76984217 version=15.1.0
Enter the GitLab instance URL (for example, https://gitlab.com/):
https://gitlab.com/
Enter the registration token:
GR1348941JkLLMR7Pz6jknFs6kxo7
Enter a description for the runner:
[LAPTOP-QI9BQ0DQ]: NombreInstancia
Enter tags for the runner (comma-separated):

Enter optional maintenance note for the runner:

Registering runner... succeeded runner=GR1348941JkLLMR7P
Enter an executor: docker+machine, parallels, shell, ssh, virtualbox, custom, docker, docker-windows, docker-ssh, docker-ssh+machine, kubernetes:
docker
Enter the default Docker image (for example, ruby:2.7):
circleci/android:api-28-node
Runner registered successfully. Feel free to start it, but if it's running already the config should be automatically reloaded!
```

Figura 2.2.1.11 Definición de la Imagen del Runner local

Con estos datos, fue completada la descripción del Runner local, ahora, solo queda inicializarlo para comprobar su registro dentro de GitLab. Para ello, lo primero será ejecutar el comando de ejecución, tal y como se muestra en la figura 2.2.1.12.



Figura 2.2.1.12 Inicio del Runner local

Y con ello, verificamos dentro de GitLab que dicho contenedor se encuentra activo y listo para ser utilizado, tal y como muestra la figura 2.2.1.13.

Available specific runners



Figura 2.2.1.13 Runner activo en GitLab

Con la configuración del Runner completada, lo restante para poder ejecutar las distintas actividades de los flujos DevOps, es escribir un archivo de configuración para la ejecución continua dentro de GitLab.

Para ello, se crea un archivo de configuración de extensión YML con el nombre “.gitlab-ci.yml”, donde se especificarán algunas opciones para su correcto funcionamiento, las cuales serán descritas a continuación.

El ítem “image” permite describir sobre que imagen se ejecutarán las instrucciones que describamos más adelante, la cual concuerda con la imagen descrita previamente en la creación del Runner local, tal como muestra la figura 2.2.1.14.

```
1 image: circleci/android:api-28-node
```

Figura 2.2.1.14 Descripción de la imagen en la configuración de CI

El ítem “before_script”, permite describir instrucciones a ser ejecutadas antes de las instrucciones que deseamos, permite, de alguna forma, preparar el ambiente para las instrucciones del flujo DevOps, tal como muestra la figura 2.2.1.15.

```
3 before_script:  
4   - echo "Before Script ..."  
5   - echo "Before Script Complete"
```

Figura 2.2.1.15 Descripción de ejecuciones previas en la configuración de CI

El ítem “stages”, permite describir cuáles serán las distintas etapas bajo las cuales serán ejecutado cada conjunto de instrucciones; en este caso, dividido para cada una de las ramas “Development” y “Testing”.

```
7 stages:
8   - build
9   - test
```

Figura 2.2.1.15 Descripción de las etapas en la configuración de CI

A partir de aquí, se describen los ítems con los cuales se diferencia cada flujo de instrucciones según el caso. El ítem “build-job”, se describió para las instrucciones a ejecutarse en la rama “Development”.

Así, dentro de este ítem se definen 3 cosas:

- Ítem “stage”, para describir qué etapa se ejecutará.
- Ítem “only”, para describir en qué rama se ejecutará.
- Ítem “script”, que representa el conjunto de instrucciones a ser ejecutadas dentro de este flujo.

Con estos ítems definidos, el conjunto de instrucciones dentro de la etapa “build-job” se encarga de compilar el proyecto, así como lo muestra la figura 2.2.1.16.

```
11 build-job:
12   stage: build
13   only:
14     - development
15   script:
16     - echo "Building on Development..."
17     - cd /builds/manticore-labs/trabajo/proyectos/trabajo-de-titulacion/cp-armijos/Aplicacion
18     - sudo chmod +x gradlew
19     - ./gradlew assembleDebug
20     - echo "Building Compelte"
```

Figura 2.2.1.15 Etapa “build-job”

Bajo las mismas directrices, la etapa “test-job” consta de los mismos ítems, sin embargo, en este caso se ejecutará dentro de la rama “Testing”. Sin embargo, en este caso, las instrucciones se encargan no solamente de compilar el proyecto, sino también de generar el archivo APK y que este sea enviado a Firebase para que junto con Firebase App-Distribution permite distribuir este APK; tal y como se muestra en la figura 2.2.1.16.


```

22 test-job:
23   stage: test
24   only:
25     - testing
26   script:
27     - echo "Building on Development..."
28     - cd /builds/manticore-labs/trabajo/proyectos/trabajo-de-titulacion/cp-armijos/Aplicacion
29     - sudo chmod +x gradlew
30     - export FIREBASE_TOKEN=1//05EuIuyEbP8XuCgYIARAAGAUSNwF-L9Iru1H_8FjOmGsTPy3F-
      kr9nWgRX7korwr5Rp2HUVVvk9axV1P021NVZVWgDnPMY-_hzjFU
31     - ./gradlew assembleDebug appDistributionUploadDebug
32     - echo "Testing Complete"

```

Figura 2.2.1.16 Etapa “test-job”

Con estas configuraciones descritas, este flujo fue configurado totalmente y listo para su ejecución, la cual será explicada a continuación.

2.2.2 Ejecución de Flujo DevOps

Conocidas las configuraciones ejecutadas, ahora es necesario conocer cómo se utilizan estas herramientas para la ejecución del flujo DevOps. Para ello, se tomará como referencia el esquema general descrito dentro de la figura 2.2.1.

Para ello, lo primero que se debe tener en cuenta es el estado de desarrollo en el que se encuentra la aplicación. Sea porque existen largos cambios que requieren ser respaldados o porque ya existe un producto de suficiente valor para que sea publicado, entonces se ejecutan las operaciones necesarias con Git para enviar dichos cambios hacia el repositorio remoto.

Para ello, podemos utilizar la interfaz de línea de comandos de Git, con ello, dentro de la carpeta donde se encuentra el proyecto, lo primero que se requiere es especificar los cambios que se quieren enviar desde el repositorio local hacia el repositorio remoto, tal como se muestra en la figura 2.2.2.1.



```

MINGW64:/c/Proyecto/cp-armijos
MY ACER@LAPTOP-QI9BQ0DQ MINGW64 /c/Proyecto/cp-armijos (development)
$ git add --all
warning: in the working copy of 'Aplicacion/app/src/main/java/com/ticml/contabilidad/InicioSesion.kt', LF will be replaced by CRLF the next time Git touches it
MY ACER@LAPTOP-QI9BQ0DQ MINGW64 /c/Proyecto/cp-armijos (development)
$

```

Figura 2.2.2.1 Especificación de los cambios a subir al repositorio remoto

A continuación, debemos ejecutar la operación de Commit, para que aquellos cambios especificados sean almacenados de forma permanente en el repositorio local, así como se observa en la figura 2.2.2.2.

```
MY ACER@LAPTOP-QI9BQ0DQ MINGW64 /c/Proyecto/cp-armijos (development)
$ git commit -a -m "Aplicacion Final v0.9.1"
[development 1c82241] Aplicacion Final v0.9.1
 2 files changed, 4 insertions(+), 116 deletions(-)
```

Figura 2.2.2.2 Commit de los cambios al repositorio local

Con ello, ya es posible enviar dichos cambios hacia el repositorio remoto, para lo cual solamente ejecutamos la operación Push hacia el repositorio remoto, así como especificando la rama sobre la cual se quieren enviar los cambios, algo muy importante, pues como se vio en las configuraciones, según la rama sobre la que se trabaje se ejecutará un flujo de instrucciones, así como muestra la figura 2.2.2.3.

```
MY ACER@LAPTOP-QI9BQ0DQ MINGW64 /c/Proyecto/cp-armijos (development)
$ git push origin development
Enumerating objects: 23, done.
Counting objects: 100% (23/23), done.
Delta compression using up to 8 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (12/12), 942 bytes | 235.00 KiB/s, done.
Total 12 (delta 5), reused 0 (delta 0), pack-reused 0
remote:
remote: To create a merge request for development, visit:
remote:   https://gitlab.com/manticore-labs/trabajo/proyectos/trabajo-de-titulacion/cp-armijos/-/merge_requests/new?merge_request%5Bsource_branch%5D=development
remote:
To https://gitlab.com/manticore-labs/trabajo/proyectos/trabajo-de-titulacion/cp-armijos.git
 624e391..1c82241  development -> development
```

Figura 2.2.2.3 Push de los cambios al repositorio remoto

Con ello, podemos comprobar que dicha operación se ve reflejada dentro del repositorio remoto, como se puede ver en la figura 2.2.2.4.

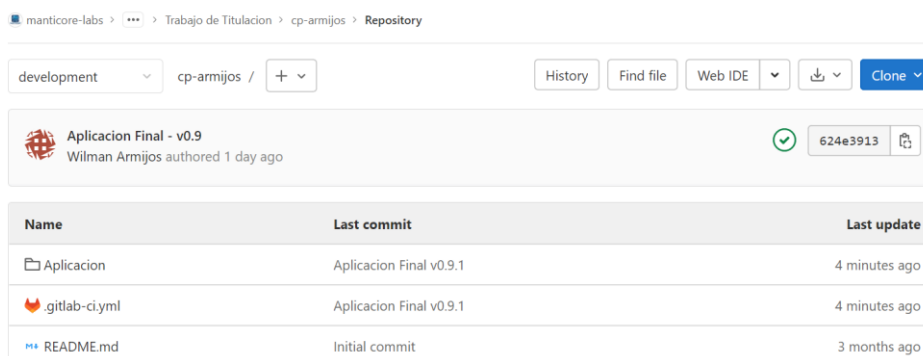


Figura 2.2.2.4 Cambios en el repositorio remoto

Aquí, es donde entra en juego el flujo DevOps, pues, en el momento que dichos cambios se ven reflejados en el repositorio remoto, las instrucciones descritas dentro del archivo de configuración “.gitlab-ci.yml” se ejecutan, de acuerdo a lo que se describió en la configuración, dicha ejecución representa la Integración Continua como parte del proyecto, y tal como se observa en la figura 2.2.2.5.

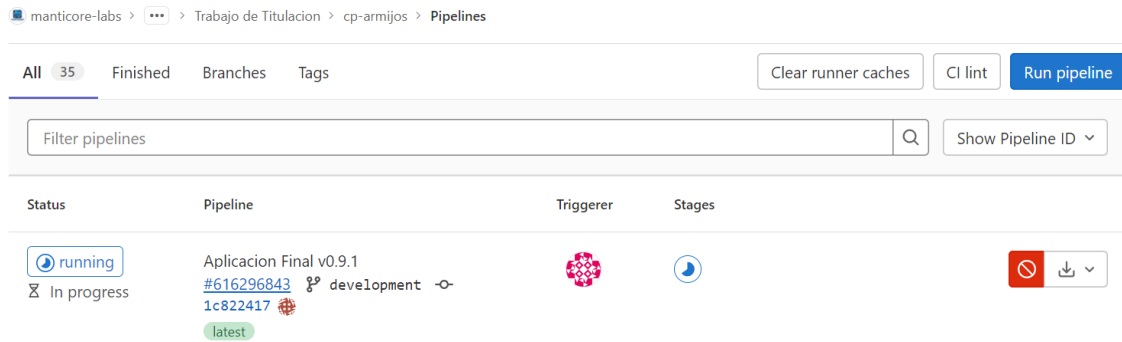


Figura 2.2.2.5 Ejecución de instrucciones con Integración Continua

Aquí, es donde la imagen de Docker entra en acción, pues, aunque dicha ejecución empieza en el servicio web de GitLab, esta es enviada hacia la imagen de Docker que describimos previamente como el Runner local. Como se aprecia en la figura 2.2.2.6, los dos primeros contenedores de Docker que ejecutaron procesos de GitLab, muy seguramente para poder enlazarse con la aplicación Docker de la máquina local y crear el tercer contenedor que, como se puede apreciar, su imagen es “circleci/android”, que es justamente la que se definió anteriormente, tanto en la creación del Runner local como en el archivo de configuración de GitLab.

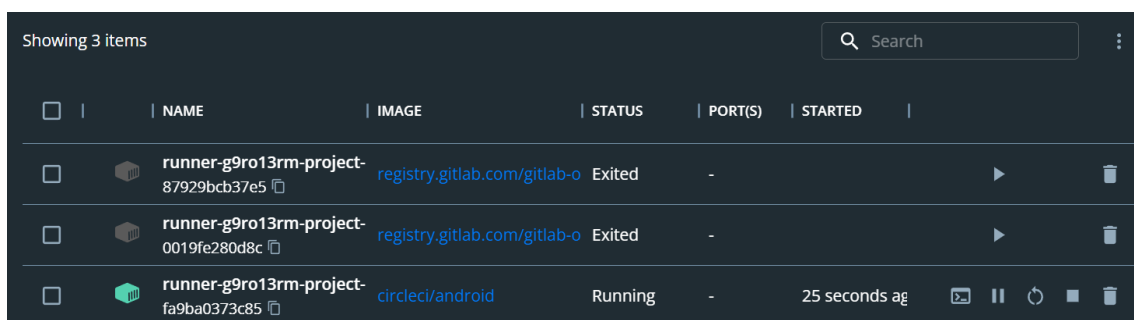


Figura 2.2.2.6 Especificación de los cambios a subir al repositorio remoto

Una vez termine dicho proceso, entonces podremos verificar el resultado de la ejecución de dicho flujo. En el caso de la figura

manticore-labs > Trabajo de Titulacion > cp-armijos > Jobs

All 47 Finished

Filter jobs

Status	Job	Pipeline	Stage	Name	Duration	Coverage
passed	#2898839220 develop... 1c822417	#616296... created by	build	build-job	00:05:46 in 1 minute	

Figura 2.2.2.7 Ejecución del flujo DevOps finalizada

Con lo descrito en el presente punto, entonces se ha definido completamente el flujo de DevOps que será utilizado en la implementación de la aplicación. Sin embargo, como unos puntos adicionales, también podemos verificar los resultados de dichas ejecuciones, por un lado, para verificar qué es lo que sucede en el caso de ejecuciones fallidas, de modo que se pueda comprobar su resultado. Para ello, podemos simplemente hacer clic sobre el número de “Job” de la figura 2.2.2.7, y se abrirá la ventana de ejecución de la figura 2.2.2.8, para observar los resultados.

Search job log

```

141 w: /builds/manticore-labs/trabajo/proyectos/trabajo-de-titulacion/cp-armijos/Aplicacion/app/src/main/java/com/ticml/contabilidad/Transaccion.kt: (105, 22): Name shadowed: cuentaCliente
142 w: /builds/manticore-labs/trabajo/proyectos/trabajo-de-titulacion/cp-armijos/Aplicacion/app/src/main/java/com/ticml/contabilidad/Transaccion.kt: (120, 61): Parameter 'view' is never used, could be renamed to _
143 w: /builds/manticore-labs/trabajo/proyectos/trabajo-de-titulacion/cp-armijos/Aplicacion/app/src/main/java/com/ticml/contabilidad/Transaccion.kt: (120, 70): Parameter 'l' is never used, could be renamed to _
144 > Task :app:compileDebugJavaWithJavac
145 > Task :app:dexBuilderDebug
146 > Task :app:mergeProjectDexDebug
147 > Task :app:mergeDebugJavaResource
148 > Task :app:packageDebug
149 > Task :app:createDebugApkListingFileRedirect
150 > Task :app:assembleDebug
151 BUILD SUCCESSFUL in 5m 30s
152 36 actionable tasks: 36 executed
153 $ echo "Building Compelte"
154 Building Compelte
156 Cleaning up project directory and file based variables
158 Job succeeded

```

build-job

Duration: 5 minutes 46 seconds
Finished: 33 minutes ago
Queued: 0 seconds
Timeout: 1h (from project)
Runner: #16662043 (g9ro13rm) LAPTOP-QI9BQ0DQ

Commit 1c822417
Aplicacion Final v0.9.1

Pipeline #616296843 for development

build

→ build-job

build-job

Figura 2.2.2.8 Resultados de la ejecución del trabajo

Con ello, se pueden tomar acciones y medidas respecto a los resultados de la ejecución, quizás respecto a las instrucciones definidas para el flujo o respecto al proyecto en sí mismo.

2.3 Flujo de Implementación de la Aplicación

Con la metodología bien definida, ahora podemos entrar a la parte de la implementación. Con ello, este flujo de implementación de la aplicación ha sido segmentado en 5 fases:

- Análisis.
- Requerimientos/Historias de Usuario.
- Diseño.
- Implementación/Pruebas de Funcionalidad.
- Pruebas de Usabilidad.
- Despliegue.

2.3.1 Análisis

Para la fase de análisis, se realizó una inspección informal sobre el problema, de modo que se abordaron las especificaciones necesarias para poder cumplir con las funcionalidades esperadas del producto final.

Dicha inspección fue plasmada en 3 artefactos que fueron la columna vertebral de la posterior implementación, los cuáles son:

- Modelo de Base de Datos (No relacional).
- Modelo de Interfaces.
- Historias de Usuario (Requerimientos)

A continuación, se presentan a breves rasgos los resultados obtenidos tras el modelamiento y descripción de dichos modelos. Así, se han dividido en un capítulo para observar los requerimientos/historias de usuario (ver capítulo 2.3.2); mientras que se ha otorgado otro capítulo para los dos modelos restantes (ver capítulo 2.3.3), que se encuentran segmentados dentro del mismo, el modelo de base de datos (ver capítulo 2.3.3.1) y el modelo de interfaces (ver capítulo 2.3.3.2).

2.3.2 Requerimientos/Historias de Usuario

Como se mencionó anteriormente, a través del análisis realizado en la etapa anterior, el primer aspecto importante por definir es la parte de los requerimientos. En este

caso, estos requerimientos fueron definidos a partir de Historias de Usuario, las cuales fueron escritas a través del formato descrito en el Anexo II.

A continuación, se describirán a breves rasgos los resultados de las historias de usuario obtenidas, describiendo en resumen su propósito a través de los escenarios contemplados para estas (Para ver las historias de usuario completas, revisar los Anexos III, IV, V, VI y VI, respectivamente).

2.3.2.1 Historia de Usuario 001

Tabla 2.3.2.1.1. Resumen de Historia de Usuario 001

Descripción General
Permite la manipulación, visualización y uso de información de las cuentas transaccionales de la empresa.
Lista de Escenarios
Visualizar lista de cuentas de la organización.
Registrar una cuenta de la organización.
Actualizar una cuenta de la organización.
Eliminar una cuenta de la organización.
Registrar una cuenta de la organización desde el registro de una transacción.

2.3.2.2 Historia de Usuario 002

Tabla 2.3.2.2.1. Resumen de Historia de Usuario 002

Descripción General
Permite la manipulación, visualización y uso de información de las cuentas transaccionales de terceros.
Lista de Escenarios
Visualizar lista de cuentas de terceros.
Registrar una cuenta de terceros.
Actualizar una cuenta de terceros.
Eliminar una cuenta de terceros.
Registrar una cuenta de terceros desde el registro de una transacción.

2.3.2.3 Historia de Usuario 003

Tabla 2.3.2.3.1. Resumen de Historia de Usuario 003

Descripción General
Permite la gestión de los periodos contables que, internamente, permita la posterior manipulación y uso de la información de las transacciones.
Lista de Escenarios
Visualizar lista de periodos contables.
Visualizar un periodo contable.
Registrar un periodo contable.

2.3.2.4 Historia de Usuario 004

Tabla 2.3.2.4.1. Resumen de Historia de Usuario 004

Descripción General
Permite la manipulación, visualización y uso de información de los documentos de facturación para su asiento contable.
Lista de Escenarios
Visualizar lista de transacciones contables de un mismo periodo contable.
Visualizar información de una transacción contable.
Visualizar lista de documentos de una transacción contable.
Visualizar documento de una transacción contable.
Agregar un documento en el registro de una transacción contable.
Registrar una transacción contable.

2.3.2.5 Historia de Usuario 005

Tabla 2.3.2.5.1. Resumen de Historia de Usuario 005

Descripción General
Permite la manipulación, visualización y uso de información de las transacciones contables para la visualización de periodos contables como motivo de sus posteriores análisis.
Lista de Escenarios
Generar informe contable sin filtro de ningún tipo.
Generar informe contable filtrado por fecha.
Generar informe contable filtrado por cliente.
Generar informe contable filtrado por cuenta contable.

2.3.3 Diseño

Es importante destacar que tanto la fase de requerimientos recién revisada, como la presente fase de diseño fue ejecutada prácticamente a la par. Esto, dado que las historias de usuario fueron descritas no solamente como requerimientos de operaciones necesarias por el cliente, sino también como flujos dentro de una aplicación descrita de forma muy fiel y cercana al producto final.

En ese sentido, tanto la descripción de la base de datos como de las interfaces, fue ejecutada a través de los requerimientos obtenidos en la fase de análisis, sin embargo, los requerimientos fueron completados una vez las interfaces y los flujos entre ellas fueron descritas.

Con esto en mente, lo primero a abordar en la fase de diseño serán las interfaces, para tener una mejor visión del producto en relación con las necesidades del cliente, y posteriormente se mostrará el diseño de la base de datos, como resultado de las fases y los artefactos anteriores y como introducción a la implementación del producto descrita en el siguiente capítulo.

2.3.3.1 Modelo de Interfaces

La aplicación consta de 11 interfaces, cada una con un propósito bien definido dentro de las funcionalidades de la aplicación, desde un inicio de sesión como motivo de ser parte de un ERP, hasta las capacidades de encontrar resúmenes con reportes de la información de las transacciones de la empresa que lo adopte. La figura 2.3.3.1.1 muestra una visión general de las interfaces que conforman la aplicación. (Para visualizar el diagrama a más detalle, revisar el Anexo VIII)

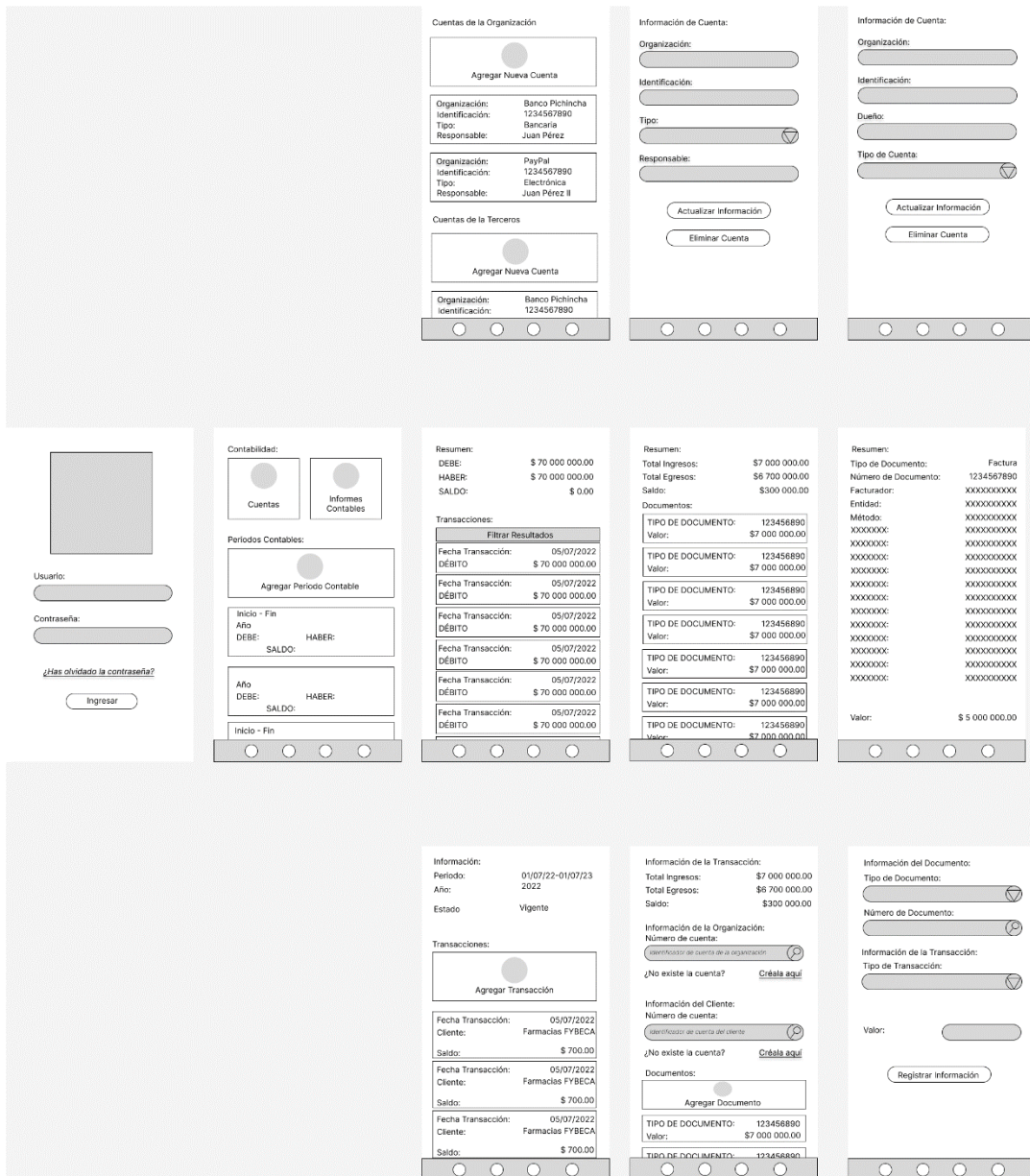


Figura 2.3.3.1.1 Diagrama de Interfaces (Vista General)

A partir de aquí se describen cada una de las pantallas que conforman todo el conjunto de interfaces de la aplicación, así como su utilidad.

La primera pantalla, es el Inicio de Sesión (ver figura 2.3.3.1.2), la cual sirve de interfaz de autenticación y autorización entre el usuario y la aplicación.

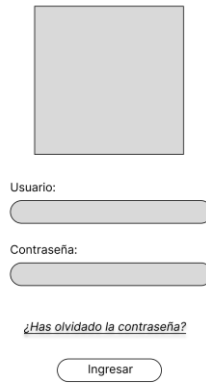


Figura 2.3.3.1.2 Interfaz – Inicio de Sesión

A continuación, después de ingresar en la aplicación tras la pantalla de inicio de sesión, se muestra la pantalla principal, que cuenta ya con algunas opciones disponibles, tal y como se destacan en la figura 2.3.3.1.13, los cuáles se pueden dividir en 3 flujos:

- Gestión de Cuentas.
- Gestión de Informes Contables.
- Gestión de Periodos Contables.

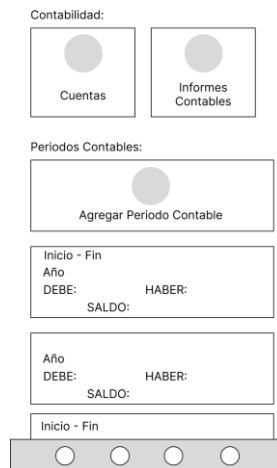



Figura 2.3.3.1.3 Interfaz – Menú Principal

Las figuras 2.3.3.1.4, 2.3.3.1.5 y 2.3.3.1.6, describen el principal flujo de la Gestión de Cuentas. Para ello, la interfaz de la figura 2.3.3.1.4 permite una visualización general de las cuentas de la empresa, así como el acceso a la creación de nuevas cuentas y acceso a cuentas existentes.

Cuentas de la Organización


 Agregar Nueva Cuenta

Organización:	Banco Pichincha
Identificación:	1234567890
Tipo:	Bancaria
Responsable:	Juan Pérez

Organización:	PayPal
Identificación:	1234567890
Tipo:	Electrónica
Responsable:	Juan Pérez II

Cuentas de la Terceros


 Agregar Nueva Cuenta

Organización:	Banco Pichincha
Identificación:	1234567890

○
○
○
○

Figura 2.3.3.1.4 Interfaz – Cuentas

Si seleccionamos una cuenta existente o la opción de agregar una nueva cuenta dentro del apartado de cuentas de la organización de la interfaz anterior, entonces se mostrará la interfaz dentro de la figura 2.3.3.1.5, siendo llena o no dependiendo de la opción, además de permitir la eliminación de la cuenta.

Información de Cuenta:

Organización:

Identificación:

Tipo:

Responsable:

○
○
○
○

Figura 2.3.3.1.5 Interfaz – Cuenta de la Organización

Siguiendo la misma lógica que la recientemente vista, la parte de la administración de las cuentas de terceros o clientes es similar con unos pequeños cambios en la interfaz, tal como muestra la figura 2.3.3.1.6, sin embargo, ofreciendo las mismas características que la previamente descrita para las cuentas de la organización.

Información de Cuenta:

Organización:

Identificación:

Dueño:

Tipo de Cuenta:

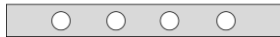


Figura 2.3.3.1.6 Interfaz – Cuenta del Cliente

A continuación, tenemos el flujo de la Gestión de Informes Contables, descritos en las figuras 2.3.3.1.7, 2.3.3.1.8 y 2.3.3.1.9.

La primera, se trata de la interfaz principal para la parte de los informes contables, tal como muestra la figura 2.3.3.1.7, que mostrará un resumen de las transacciones, así como de valores globales para un periodo de tiempo determinado, la cual también cuenta con una opción de filtrado de resultados que ofrece por medio de:

- Intervalo de Tiempo.
- Cuentas Contables.
- Cuentas de los Clientes.

Resumen:

DEBE:	\$ 70 000 000.00
HABER:	\$ 70 000 000.00
SALDO:	\$ 0.00

Transacciones:

Filtrar Resultados	
Fecha Transacción:	05/07/2022
DÉBITO	\$ 70 000 000.00
Fecha Transacción:	05/07/2022
DÉBITO	\$ 70 000 000.00
Fecha Transacción:	05/07/2022
DÉBITO	\$ 70 000 000.00
Fecha Transacción:	05/07/2022
DÉBITO	\$ 70 000 000.00
Fecha Transacción:	05/07/2022
DÉBITO	\$ 70 000 000.00
Fecha Transacción:	05/07/2022
DÉBITO	\$ 70 000 000.00



Figura 2.3.3.1.7 Interfaz – Informe Contable

La figura 2.3.3.1.8, muestra otra interfaz, la cuál será desplegada al escoger una de las transacciones de la interfaz anterior, ofreciendo una vista de los detalles de esta,

así como la opción a revisar los documentos que han sido agregados dentro de la misma.

Resumen:	
Total Ingresos:	\$7 000 000.00
Total Egresos:	\$6 700 000.00
Saldo:	\$300 000.00
Documentos:	
TIPO DE DOCUMENTO:	123456890
Valor:	\$7 000 000.00
TIPO DE DOCUMENTO:	123456890
Valor:	\$7 000 000.00
TIPO DE DOCUMENTO:	123456890
Valor:	\$7 000 000.00
TIPO DE DOCUMENTO:	123456890
Valor:	\$7 000 000.00
TIPO DE DOCUMENTO:	123456890
Valor:	\$7 000 000.00
TIPO DE DOCUMENTO:	123456890
Valor:	\$7 000 000.00
TIPO DE DOCUMENTO:	123456890
Valor:	\$7 000 000.00

Figura 2.3.3.1.8 Interfaz – Detalles de Transacción

Como se mencionó previamente, después de entrar a las transacciones, podemos entrar dentro de los detalles de los documentos, tal y como muestra la figura 2.3.3.1.9, donde se encuentran varios de los detalles que resumen a los documentos (facturas, notas de créditos o notas de débito), que han sido procesados.

Resumen:	
Tipo de Documento:	Factura
Número de Documento:	1234567890
Facturador:	XXXXXXXXXX
Entidad:	XXXXXXXXXX
Método:	XXXXXXXXXX
XXXXXXXX:	XXXXXXXXXX
XXXXXXXX:	XXXXXXXXXX
XXXXXXXX:	XXXXXXXXXX
XXXXXXXX:	XXXXXXXXXX
XXXXXXXX:	XXXXXXXXXX
XXXXXXXX:	XXXXXXXXXX
XXXXXXXX:	XXXXXXXXXX
XXXXXXXX:	XXXXXXXXXX
XXXXXXXX:	XXXXXXXXXX
XXXXXXXX:	XXXXXXXXXX
XXXXXXXX:	XXXXXXXXXX
XXXXXXXX:	XXXXXXXXXX
XXXXXXXX:	XXXXXXXXXX
XXXXXXXX:	XXXXXXXXXX
Valor:	\$ 5 000 000.00

Figura 2.3.3.1.9 Interfaz – Detalles de Documento

Por último, tenemos el flujo de la Gestión de Periodos Contables, los cuáles, se centran en la capacidad de dar al cliente las opciones de anotar transacciones dentro de un periodo contable predefinido. Su flujo se describe en las siguientes tres figuras.

La figura 2.3.3.1.10, muestra un breve resumen con información acerca del periodo actual que está siendo manejado por la empresa, donde se muestra un pequeño resumen del Periodo en sí mismo, así como la capacidad de Agregar (Anotar) una transacción en ella.

Información:

Periodo: 01/07/22-01/07/23
 Año: 2022
 Estado: Vigente

Transacciones:


 Agregar Transacción

Fecha Transacción:	05/07/2022
Cliente:	Farmacias FYBECA
Saldo:	\$ 700.00

Fecha Transacción:	05/07/2022
Cliente:	Farmacias FYBECA
Saldo:	\$ 700.00

Fecha Transacción:	05/07/2022
Cliente:	Farmacias FYBECA
Saldo:	\$ 700.00



Figura 2.3.3.1.10 Interfaz – Periodo Contable


Esta capacidad de anotar la transacción se ejecuta en la siguiente interfaz (ver figura 2.3.3.1.11), donde se otorga al usuario las herramientas necesarias para anotar una transacción.

Información de la Transacción:

Total Ingresos: \$7 000 000.00
 Total Egresos: \$6 700 000.00
 Saldo: \$300 000.00

Información de la Organización:


Número de cuenta:



¿No existe la cuenta? [Créala aquí](#)


Información del Cliente:

Número de cuenta:



¿No existe la cuenta? [Créala aquí](#)

Documentos:


 Agregar Documento

TIPO DE DOCUMENTO:	123456890
Valor:	\$7 000 000.00

TIPO DE DOCUMENTO:	123456890
--------------------	-----------




Figura 2.3.3.1.11 Interfaz – Asiento Contable de Transacción

Para finalizar este flujo, una transacción puede agrupar más de un documento a la vez, para cuál se ha integrado la interfaz que se observa en la figura 2.3.3.1.12, donde se observa un formulario cuyo fin es el de permitir al usuario agregar documentos a una transacción, para que pueda ser anotada posteriormente.

Información del Documento:

Tipo de Documento:

Número de Documento:

Información de la Transacción:

Tipo de Transacción:

Valor:




Figura 2.3.3.1.12 Interfaz – Agregar Documento a Transacción

Con esto, se ha repasado en breves rasgos las interfaces, así como los flujos y las funciones que ocupa cada una de ellas dentro de la aplicación.

2.3.3.2 Modelo de Base de Datos

Con la información previamente descrita, ya se puede entrar en una fase de modelo más orientada a la implementación, algo que recae dentro del diseño del modelo de base de datos que debe implementarse para que sostenga la aplicación.

En este caso, dado que la aplicación no se centra especialmente en la velocidad o la cantidad de escrituras ejecutadas, sino más bien en la importancia de la visualización de los datos, se decidió hacer uso de una Base de Datos No Relacional, para lo cual el proyecto fue apoyado por el servicio Firebase Cloud Firestore, otorgado por Google.

Con esto en mente, se describe de forma rápida el esquema de base de datos utilizado para su construcción (Para visualizar el modelo a mayor detalle, revisar el Anexo IX).

La figura 2.3.3.2.1 muestra el esquema general de la base de datos modelada para el proyecto, destacando principalmente como se agrupan las colecciones dentro de toda la base de datos. Para ello, se ha segmentado la base de datos en dos grupos:

- Colecciones Dinámicas en Ejecución: que representa aquellas colecciones que pueden aceptar cambios en sus documentos durante el uso de la aplicación.

- Colecciones Estáticas en Ejecución: que representa aquellas colecciones que no pueden ser modificadas de ninguna forma durante la ejecución, sino que, por su naturaleza, existen fuera de este, como listas estáticas o como conjuntos de datos que deben ser agregados de forma externa.

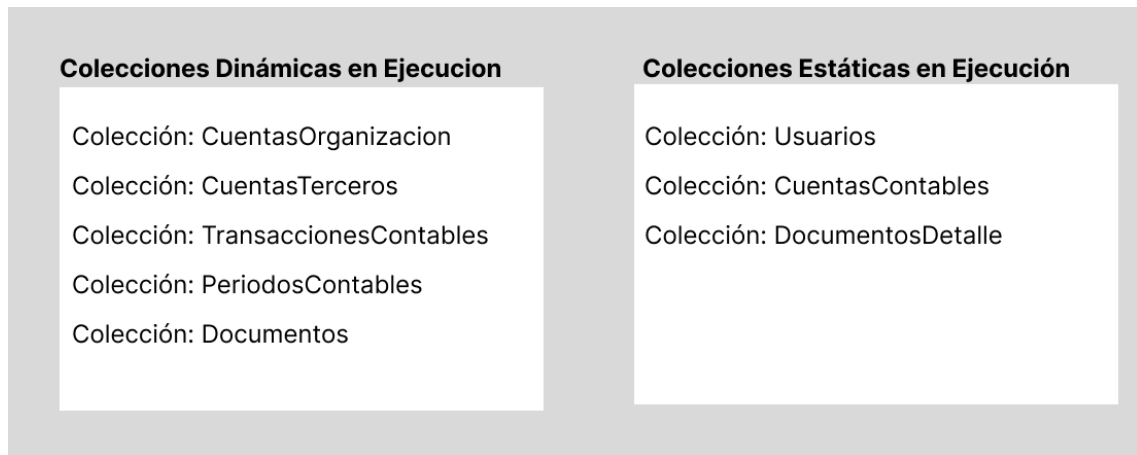


Figura 2.3.3.2.1 Estructura de la Base de Datos – Colecciones

Con la estructura de las colecciones en mente, ahora podemos repasar la información contenida dentro de cada Colección, para ello, lo que se presentará a continuación es la estructura de cada uno de los documentos que forman parte de las colecciones dentro de la base de datos; esto, teniendo en cuenta que todos los documentos dentro de una misma colección cuentan con la misma estructura y conjunto de atributos.

La figura 2.3.3.2.2 muestra la estructura de los documentos de la colección de usuarios, que cuenta únicamente con el atributo de la clave o contraseña, mientras el usuario es representado por el nombre del documento.



Figura 2.3.3.2.2 Estructura – Usuarios

La figura 2.3.3.2.3 muestra la estructura de los documentos de las cuentas contables, donde se establecen los 5 niveles de estas cuentas, además de contar con un campo auxiliar para ejecutar algunas consultas dentro de la aplicación.

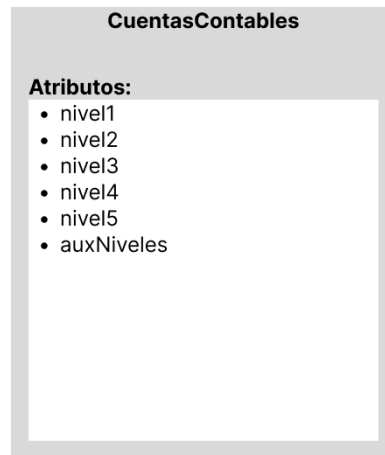


Figura 2.3.3.2.3 Estructura – Cuentas Contables

La figura 2.3.3.2.4 muestra la estructura de los documentos de tipo detalle, donde se muestran todos los atributos correspondientes y definidos como necesarios para entender a los posibles documentos que pueden ser procesados dentro de la aplicación como parte del asiento de una transacción.

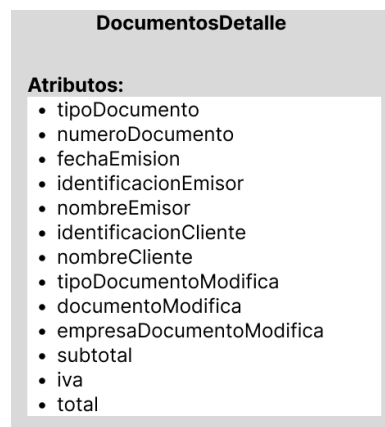


Figura 2.3.3.2.4 Estructura – Detalles de Documentos

La figura 2.3.3.2.5 muestra la estructura de las cuentas de la organización, donde, en relación con la interfaz, muestra los mismos datos necesarios, a la par de un identificador necesario para su relación con otros documentos.

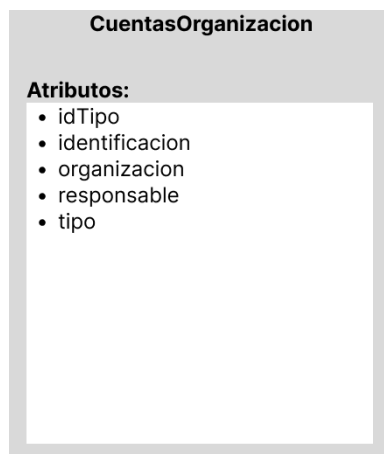


Figura 2.3.3.2.5 Estructura – Cuentas de la Organización

Tal y como con la interfaz anterior, la figura 2.3.3.2.6 muestra una estructura similar pero destinada al manejo de cuentas de terceros, que, al igual como se mostró en la definición de las interfaces, cuenta con los mismos datos.

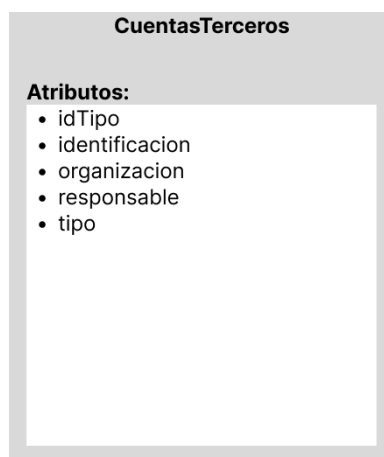


Figura 2.3.3.2.6 Estructura – Cuentas de Terceros

La figura 2.3.3.2.7 muestra la estructura de las transacciones contables, reflejando dentro de sus atributos el asiento de la transacción ejecutado, este cuenta con atributos propios de la transacción, y a su vez cuenta con algunos atributos como idPeriodo o todos los atributos cuyo nombre termina con la palabra filtro, los cuales fueron definidos para facilitar la ejecución de algunas operaciones dentro de la lógica de la implementación de la aplicación.

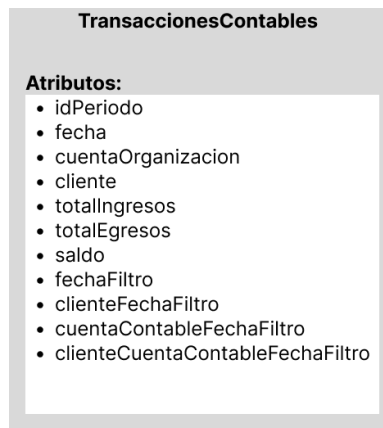


Figura 2.3.3.2.7 Estructura – Transacciones Contables

La figura 2.3.3.2.8 demuestra la estructura de los documentos de los Periodos Contables, que cuentan con una información básica de esta.

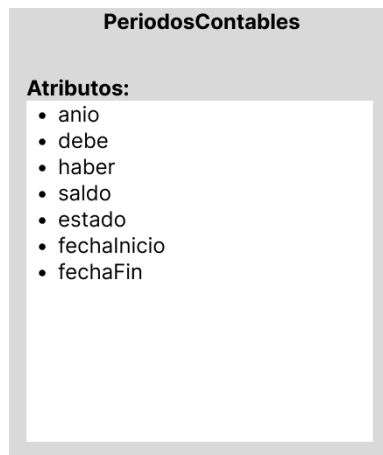


Figura 2.3.3.2.8 Estructura – Periodos Contables

Para finalizar, la figura 2.3.3.2.9 describe la estructura de la colección de documentos que, en relación con la colección descrita anteriormente, no representa los documentos como tal, sino la inclusión de dichos documentos como parte de una transacción que ha sido registrada. Por ello, es que cuenta con atributos como idTransacciónDetalle o idDocumentoDetalle, ambos campos que permiten la relación de los documentos con la transacción a la cual pertenece, así como el documento que pretende relacionarlo.

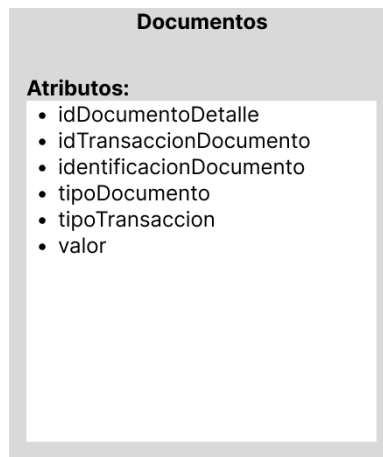


Figura 2.3.3.2.9 Estructura – Documentos

2.3.4 Implementación/Pruebas de Funcionalidad

Antes de entrar de lleno al proceso de implementación, se describirán algunos puntos acerca del proceso, para describir algunas de las herramientas y generalidades tomadas en cuenta dentro del proceso, así:

- La implementación del producto se ejecutó dentro del ambiente de Android Studio, de esa manera, la aplicación está destinada a ejecutarse dentro de entornos sobre sistemas operativos Android, o entornos semejantes capaces de ejecutar archivos APK.
- Se hizo uso de la plataforma de servicios Firebase, tanto para la definición de la base de datos a través de Firebase Cloud Firestore, así como para el lanzamiento y distribución del producto final a través de Firebase App-Distribution.

Una vez se han definido los modelos y los requerimientos del producto final, así como se han hecho las aclaraciones necesarias en el párrafo anterior, podemos revisar cuál fue el desempeño dentro del desarrollo de la aplicación. Para explicar el proceso de implementación de la aplicación, se ha utilizado el esquema de Scrum, y se ha segmentado el proceso en cada uno de los Sprints ejecutados, cada uno de los cuáles será descrito a continuación.

Del mismo modo, y tal como se definieron los eventos y los artefactos, para definir cada uno de los Sprints se hará uso de los 3 eventos que fueron descritos en la tabla 2.1.3 dentro del capítulo 2.1, incluyendo además un pequeño apartado donde se estudiarán los resultados obtenidos dentro de dicho Sprint, en relación con lo planificado al inicio del mismo.

2.3.4.1 Sprint 1

Durante los primeros Sprints, la implementación se centró en la configuración del entorno de Integración Continua, así como la finalización de las etapas de diseño previamente descritas.

Sprint Planning

Para el primer Sprint, se marcan los siguientes objetivos:

- Finalizar la descripción de las historias de usuario.
- Ajustar las interfaces de usuario.
- Revisar configuración de Integración Continua en GitLab.

Ejecución del Sprint

En estas fases iniciales, los esfuerzos fueron centrados en el entendimiento y la práctica de las herramientas nuevas a ser utilizadas:

- Por un lado, tal como se revisaron en los capítulos anteriores, los modelos de Bases de Datos, de Interfaces y las historias de Usuario fueron completamente descritas, de modo que permitan entrar de pleno en la codificación.
- Por otro lado, se revisó información acerca del uso y configuración de la herramienta GitLab, principalmente, para la inclusión de la Integración Continua dentro de proyectos de desarrollo. Para ello, se revisó diversa documentación de algunas fuentes, principalmente del propio GitLab, además desde propias fuentes de CircleCI/Android.

Los resultados de los modelos y de las historias de usuarios, se encuentran dentro de los Anexos III, IV, V, VI, VII, VIII y IX, tal y como se había descrito anteriormente.

Sprint Review

Tabla 2.3.4.1.1 Resultados del Sprint Review

Pregunta	Resultado
¿Qué hice en el Sprint actual?	Descripción de las Historias de Usuario. Ajustes y Mejoras en las Interfaces de Usuario.

	Revisión de la Configuración de CI/CD en GitLab.
¿Qué problemas tuvo?	Falta de práctica y conocimiento del entorno y las capacidades de GitLab.
¿Qué hare el siguiente Sprint?	Configurar CI/CD en GitLab con soporte de Docker. Configurar el entorno de Android Studio con Firebase. Inclusión del proyecto en GitLab.

Resultados del Sprint

En la tabla 2.3.4.1.2, se encuentra un resumen sobre los resultados del presente Sprint, describiendo a mayor detalle las actividades que se planificaron dentro de la fase de Sprint Planning, además de describir el porcentaje de avance.

Tabla 2.3.4.1.2 Resultados del Sprint

Actividad	Ejecución
Descripción de la Historia de Usuario 001	Sí
Descripción de la Historia de Usuario 002	Sí
Descripción de la Historia de Usuario 003	Sí
Descripción de la Historia de Usuario 004	Sí
Descripción de la Historia de Usuario 005	Sí
Diseño de la interfaz de Inicio de Sesión	Sí
Diseño de la Página Principal	Sí
Diseño de la interfaz de Cuentas	Sí
Diseño de la interfaz de Cuenta de la Organización	Sí
Diseño de la interfaz de Cuenta de Terceros	Sí
Diseño de la interfaz de Periodo Contable	Sí
Diseño de la interfaz de Asiento de Transacción	Sí
Diseño de la interfaz de Documento en Transacción	Sí
Diseño de la interfaz de Informe Contable	Sí
Diseño de la interfaz de Detalle de Transacción	Sí
Diseño de la interfaz de Detalle de Documento	Sí
Revisión de GitLab Repositories	Sí
Revisión de GitLab Runners	No
Revisión de GitLab CI/CD	Sí
Revisión de GitLab Jobs	No
PORCENTAJE CONSEGUIDO (Aproximado):	90%

2.3.4.2 Sprint 2

Sprint Planning

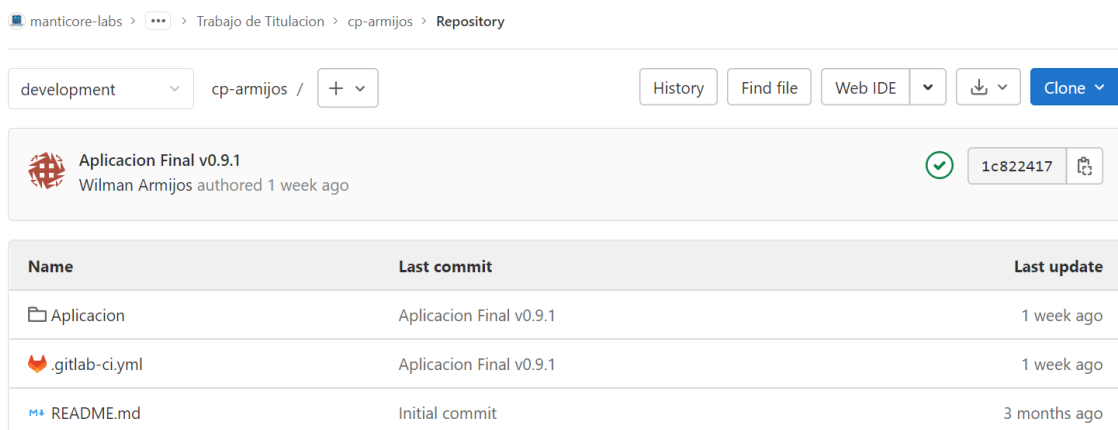
En el Segundo Sprint, se definen las siguientes actividades:

- Subir el proyecto de Android Studio en el repositorio remoto de GitLab.
- Ejecutar pruebas y más configuraciones con el entorno de GitLab.
- Configuración de Git para las operaciones de carga y descarga del proyecto entre los repositorios.

Ejecución del Sprint

Respecto al segundo Sprint, lo primero por ser ejecutado fue el proceso de inclusión del proyecto dentro del repositorio remoto ubicado en GitLab.

Todos los archivos parte del proyecto se encuentran dentro del repositorio que se encuentra en el enlace del Anexo I, la figura 2.3.4.2.1 muestra el proyecto dentro del repositorio.



The screenshot shows the GitLab interface for a repository. The breadcrumb path is 'manticore-labs > Trabajo de Titulacion > cp-armijos > Repository'. The current branch is 'development' and the file path is 'cp-armijos /'. There are buttons for 'History', 'Find file', 'Web IDE', 'Clone', and 'Download'. A commit is highlighted with a green checkmark, showing the commit hash '1c822417' and the author 'Wilman Armijos' who authored it '1 week ago'. Below this, a table lists the commit history:

Name	Last commit	Last update
Aplicacion	Aplicacion Final v0.9.1	1 week ago
.gitlab-ci.yml	Aplicacion Final v0.9.1	1 week ago
README.md	Initial commit	3 months ago

Figura 2.3.4.2.1 Proyecto dentro del repositorio remoto

Por otro lado, se realizaron más revisiones para definir un bosquejo inicial del archivo YML en GitLab, cuyo resultado aún no se lograba en punto de ejecución válido, aún donde se ejecutaban varias pruebas.

Sprint Review

Tabla 2.3.4.2.1 Resultados del Sprint Review

Pregunta	Resultado
¿Qué hice en el Sprint actual?	Inclusión del proyecto de Android Studio en el repositorio de GitLab. Avance y pruebas con el archivo de configuración de Integración Continua en GitLab.
¿Qué problemas tuve?	Falta de práctica y conocimiento del entorno de GitLab. Problemas con el uso del cliente de escritorio de Git (GitKraken).
¿Qué hare el siguiente Sprint?	Configurar correctamente las necesidades de Git (integración con GitLab y el repositorio local). Desarrollo de la aplicación: interfaces de Inicio de Sesión y la Gestión de Cuentas.

Resultados del Sprint

En la tabla 2.3.4.2.2, se encuentra un resumen sobre los resultados del presente Sprint, describiendo a mayor detalle las actividades que se planificaron dentro de la fase de Sprint Planning, además de describir el porcentaje de avance.

Tabla 2.3.4.2.2 Resultados del Sprint

Actividad	Ejecución
Instalación y configuración de Git	Sí
Pruebas de Git	Sí
Clonación del repositorio al computador local	Sí
Push del proyecto de Android Studio a GitLab	No
Instalación de GitLab Runners	Sí
Configuración de GitLab Runners en el computador	No
Configuración de Docker	No
Configuración de GitLab Runners en GitLab	No
Descripción de archivo de configuración en GitLab	No
Pruebas de GitLab Runners.	No
PORCENTAJE CONSEGUIDO (Aproximado):	40%

2.3.4.3 Sprint 3

Sprint Planning

Dentro del tercer Sprint, se planean las siguientes actividades:

- Inclusión del proyecto en GitLab con la configuración total de Git.
- Configuración del archivo YML dentro de GitLab para la Integración Continua.
- Desarrollo de las Interfaces de Inicio de Sesión y Gestión de Cuentas.

Ejecución del Sprint

En este punto, ya fue posible definir el archivo de configuración para la definición de los flujos DevOps, como primer avance. Con ello, el resultado del archivo YML dentro de GitLab se muestra en la figura 2.3.4.3.1.



```
.gitlab-ci.yml 405 bytes
Edit in pipeline editor
Replace Delete
1 image: circleci/android:api-28-node
2
3 before_script:
4   - echo "Before Script ..."
5   - echo "Before Script Complete"
6
7 stages:
8   - build
9   - test
10
11 build-job:
12   stage: build
13   script:
14     - echo "Building on Development..."
15     - cd /builds/manticore-labs/trabajo/proyectos/trabajo-de-titulacion/cp-armijos/Aplicacion
16     - sudo chmod +x gradlew
17     - ./gradlew assembleDebug
18     - echo "Building Compelte"
```

Figura 2.3.4.3.1 Archivo de Configuración Inicial

Con la inclusión del archivo de configuración ya funcionando, ya fue posible dedicarse exclusivamente al desarrollo de código y, aún más importante, a la ejecución de este proceso con la integración y ejecución de los flujos DevOps.

Con eso, lo primero que se ejecutó fue el desarrollo de la interfaz de Inicio de Sesión dentro del ambiente de desarrollo, tal como muestra la figura 2.3.4.3.2.

```
.gitlab-ci.yml 405 bytes
Edit in pipeline editor Replace Delete

1 image: circleci/android:api-28-node
2
3 before_script:
4   - echo "Before Script ..."
5   - echo "Before Script Complete"
6
7 stages:
8   - build
9   - test
10
11 build-job:
12   stage: build
13   script:
14     - echo "Building on Development..."
15     - cd /builds/manticore-labs/trabajo/proyectos/trabajo-de-titulacion/cp-armijos/Aplicacion
16     - sudo chmod +x gradlew
17     - ./gradlew assembleDebug
18     - echo "Building Compelte"
```

Figura 2.3.4.3.2 Archivo de Configuración Inicial

Con esto en mente, ejecutar los flujos DevOps descritos dentro del capítulo 2.2.2. Para ello, y siguiendo con la planificación del Sprint, lo primero en concretarse fue la ejecución de la implementación de la interfaz de Inicio de Sesión de la Aplicación, tal como muestra la figura 2.3.4.3.3.

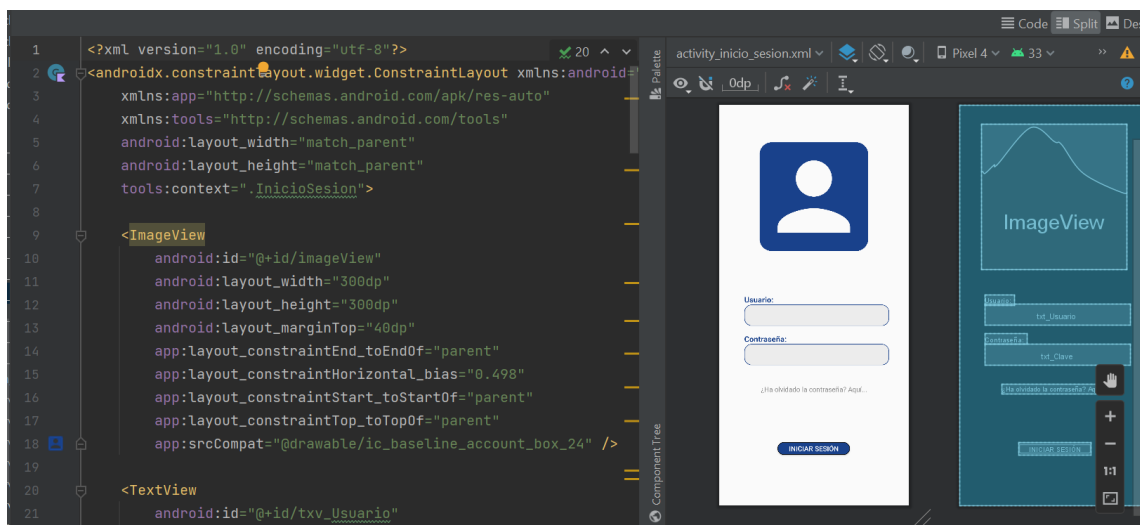


Figura 2.3.4.3.3 Implementación de la interfaz de Inicio de Sesión

Con esto desarrollado, se procedió a hacer una prueba “real” del flujo DevOps definido en el Sprint anterior. Así, podemos ver el resultado de dicho flujo dentro del propio GitLab, tal como muestra la figura 2.3.4.3.4.

Interfaces - Inicio Sesion y Pruebas

The screenshot shows the GitLab CI/CD pipeline interface. At the top, it indicates the parent commit is 4848b24a and the current pipeline is 'testing'. A message states 'No related merge requests found'. Below that, a green checkmark indicates 'Pipeline #609976401 passed with stage in 4 minutes'. The interface has tabs for 'Changes' (59) and 'Pipelines' (2). The pipeline list shows two entries:

Status	Pipeline	Triggerer	Stages
 00:04:00 2 weeks ago	Interfaces - Inicio Sesion y Pruebas #609976401 testing -> 67ce49c2		
 00:00:11 2 weeks ago	Interfaces - Inicio Sesion y Pruebas #607808186 development -> 67ce49c2		

Figura 2.3.4.3.4 Ejecución de flujo DevOps (build)

Así mismo, se puede observar que la ejecución resultó fallida en un primer momento, sin embargo, esto fue debido solamente a que el entorno de Docker no se encontraba activo, una vez esto fue solucionado y volverse a ejecutar, fue completado exitosamente.

Sprint Review

Tabla 2.3.4.3.1 Resultados del Sprint Review

Pregunta	Resultado
¿Qué hice en el Sprint actual?	Inclusión y configuración del proyecto en GitLab y Git. Descripción del archivo de configuración para la Integración Continua en GitLab. Desarrollo de la Interfaz de Inicio de Sesión.
¿Qué problemas tuve?	Dificultades en configuración de GitLab (Solucionados en medio de la ejecución del Sprint). Problemas con el flujo con Docker, debido a limitaciones de red y procesamiento del computador.
¿Qué hare el siguiente Sprint?	Desarrollo de las interfaces y su flujo en Android Studio. Desarrollo del flujo completo de cuentas base.

Resultados del Sprint

En la tabla 2.3.4.3.2, se encuentra un resumen sobre los resultados del presente Sprint, describiendo a mayor detalle las actividades que se planificaron dentro de la fase de Sprint Planning, además de describir el porcentaje de avance.

Tabla 2.3.4.3.2 Resultados del Sprint

Actividad	Ejecución
Inclusión completa del proyecto en GitLab	Sí
Creación del archivo de configuración YML en GitLab	Sí
Configuración del archivo YML en GitLab	Parcialmente
Desarrollo de la interfaz de Inicio de Sesión	Sí
Desarrollo de la Interfaz de Cuentas	No
PORCENTAJE CONSEGUIDO (Aproximado):	70%

2.3.4.4 Sprint 4

Sprint Planning

Dentro del cuarto Sprint, se definen las actividades descritas a continuación:

- Desarrollo de las interfaces de la aplicación.
- Desarrollo de la lógica de la interfaz de Inicio de Sesión y de la Página Principal.

Ejecución del Sprint

Después de comenzar con la implementación, se decidió, en lugar de ejecutar la implementación a través de conjunto de interfaces, primero, diseñar todas las interfaces dentro de la aplicación.

Tras la página de Inicio de Sesión, la interfaz de la Página Principal fue diseñada, tal y como muestra la figura 2.3.4.4.1.

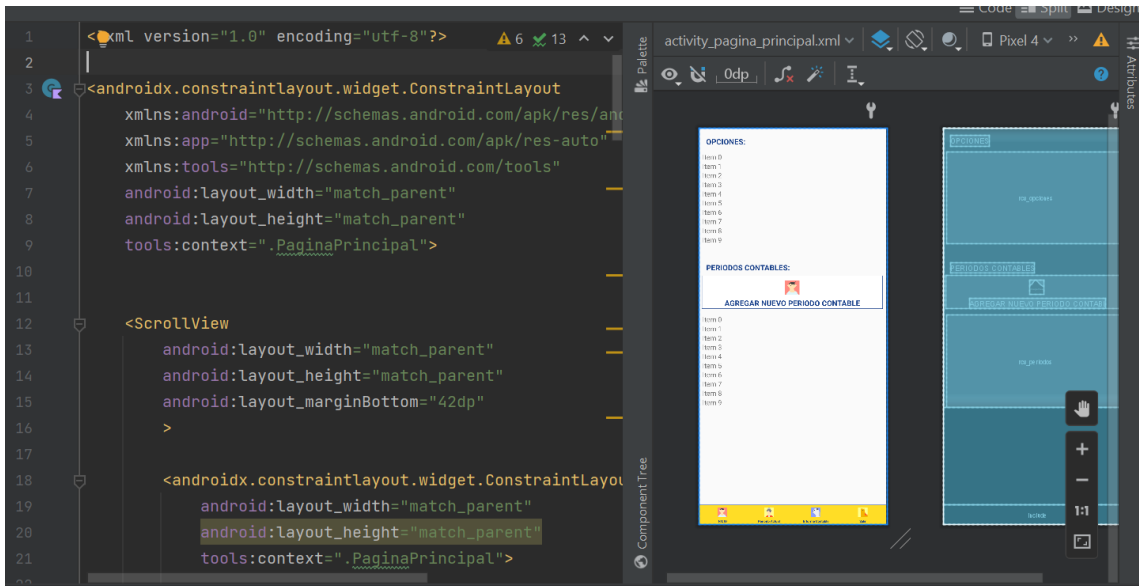


Figura 2.3.4.4.1 Implementación de la interfaz de la Página Principal

Lo siguiente, fue la interfaz de las cuentas como muestra la figura 2.3.4.4.2.

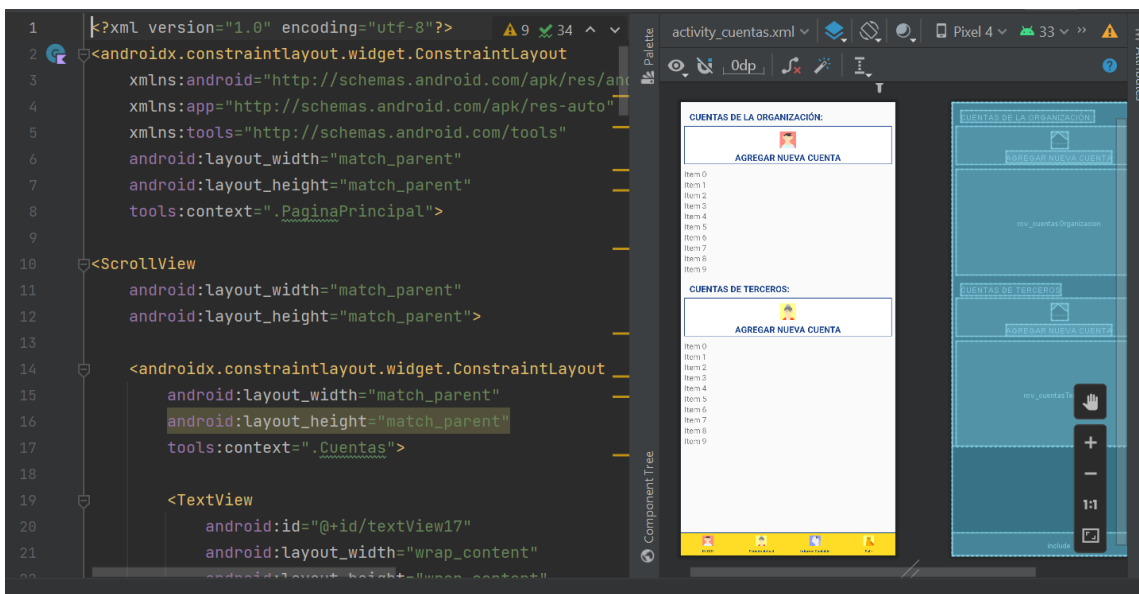


Figura 2.3.4.4.2 Implementación de la interfaz de Cuentas

A continuación, como se ve en la figura 2.3.4.4.3, fue el diseño de la interfaz de las cuentas de la organización.

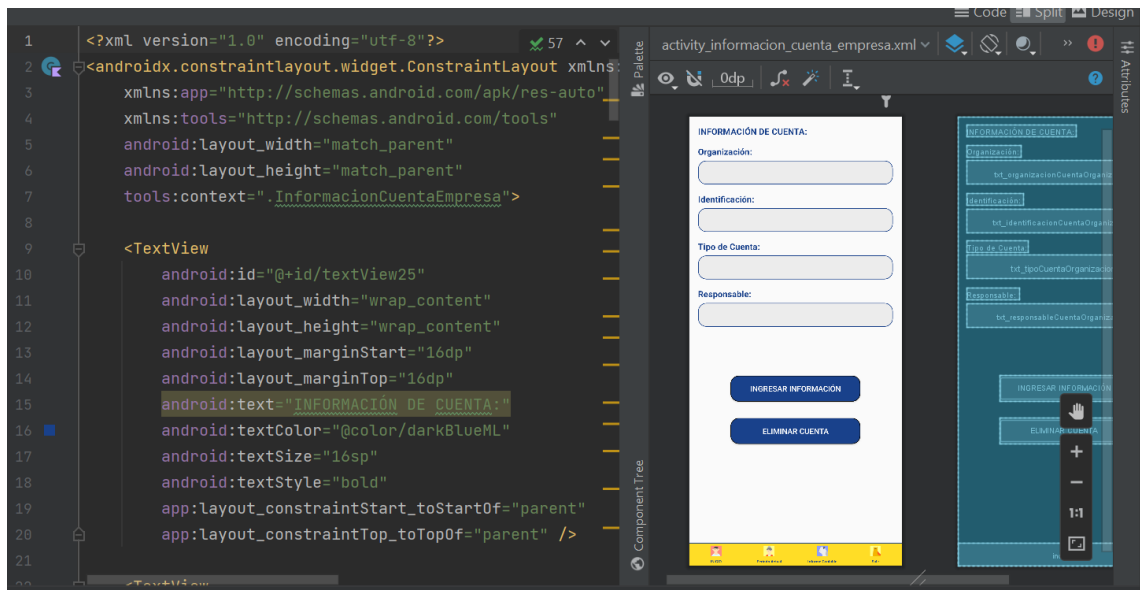


Figura 2.3.4.4.3 Implementación de la interfaz de Cuentas de la Organización

De la misma manera, lo siguiente fue el diseño de las cuentas de terceros, esto se puede observar en la figura 2.3.4.4.4.

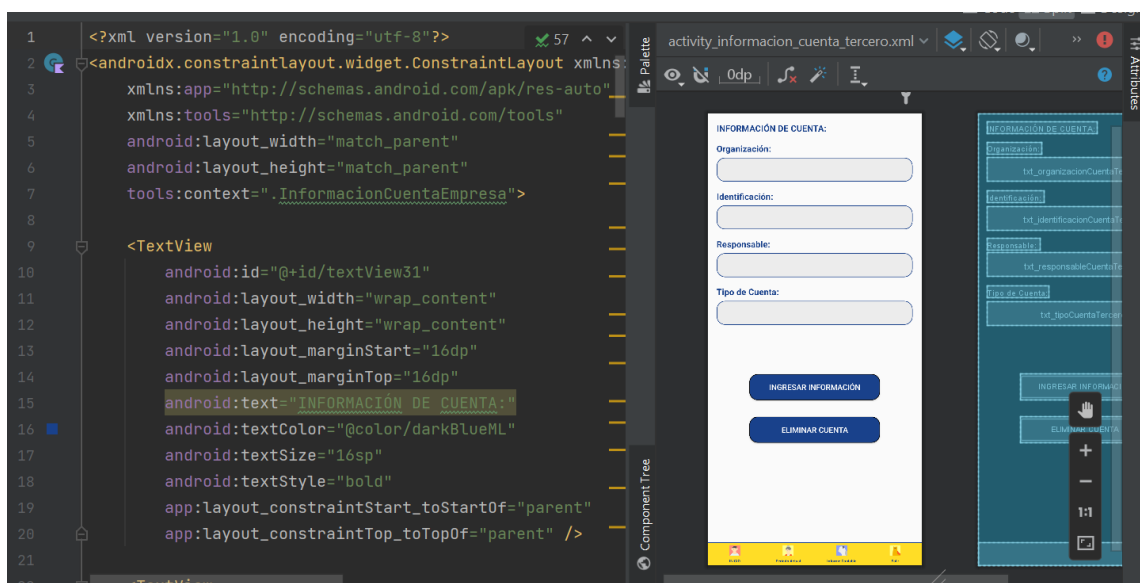


Figura 2.3.4.4.4 Implementación de la interfaz de Cuentas de Terceros

Lo siguiente, fue el diseño de la interfaz de Periodos Contables, la cual dará paso a los asientos contables, esta se muestra en la figura 2.3.4.4.5.

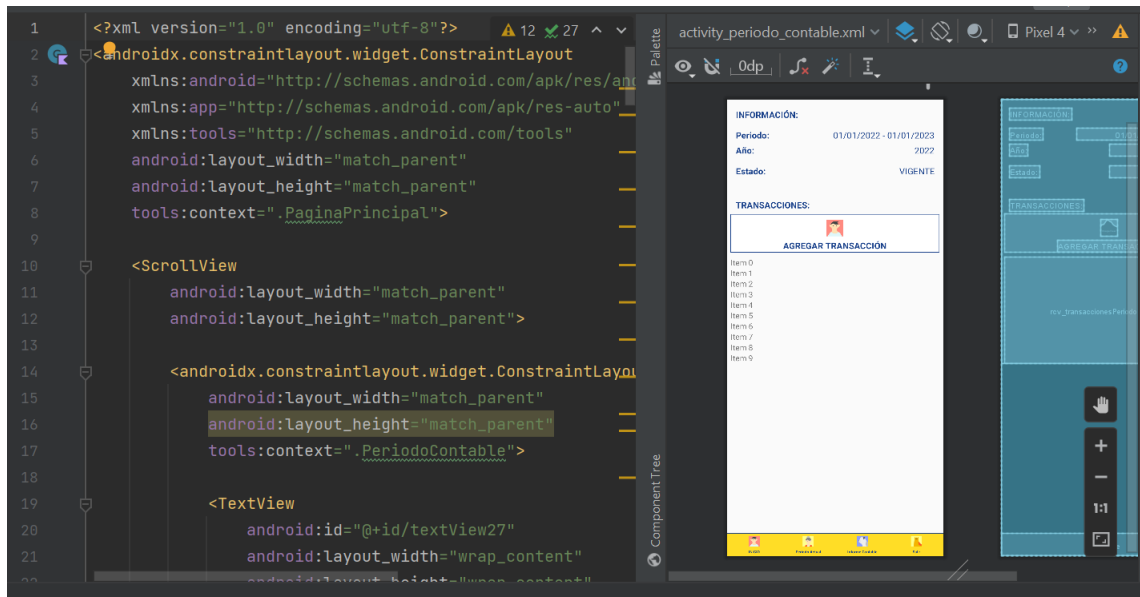


Figura 2.3.4.4.5 Implementación de la interfaz del Periodo Contable

Con la interfaz anterior, ya podemos describir la interfaz de Transacciones (asiento contable), como muestra la figura 2.3.4.4.6.

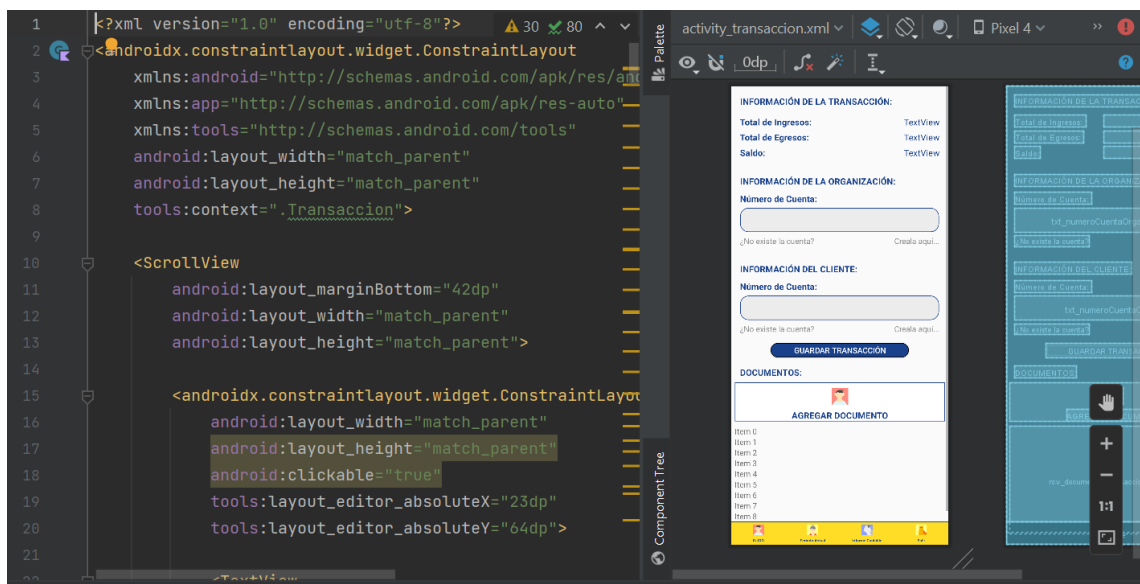


Figura 2.3.4.4.6 Implementación de la interfaz de Transacciones

Como parte final del flujo de las transacciones, se ejecutó el diseño de la interfaz de Documentos, que no es más que una interfaz adicional para los asientos contables, así, esto se muestra en la figura 2.3.4.4.7.

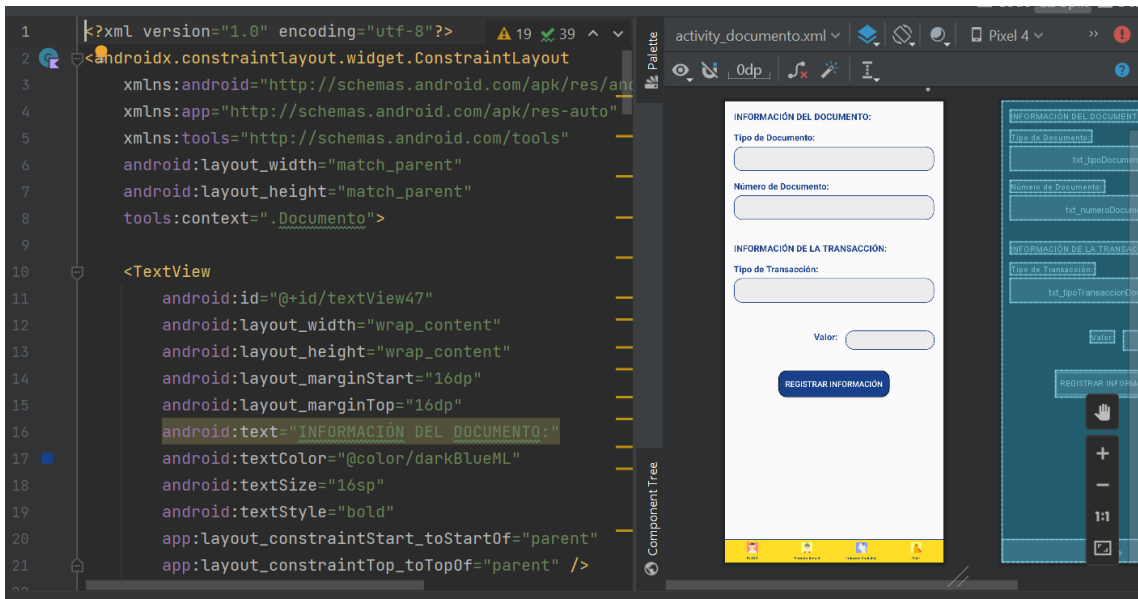


Figura 2.3.4.4.7 Implementación de la interfaz de Documentos

Para finalizar, se tienen las interfaces del flujo de Informes Contables, la interfaz principal de este flujo se describe en la figura 2.3.4.4.8.

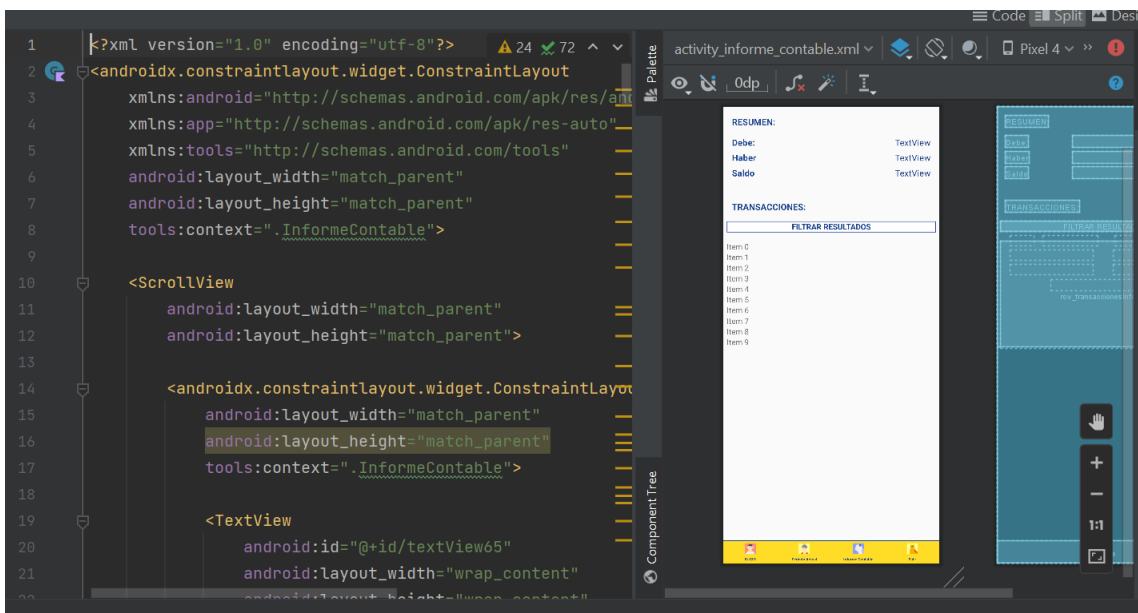


Figura 2.3.4.4.8 Implementación de la interfaz de Informe Contable

Luego, tenemos dos interfaces adicionales, principalmente para mostrar información adicional de algunos conceptos, el primero, de las transacciones, la cual se muestra en la figura 2.3.4.4.9.

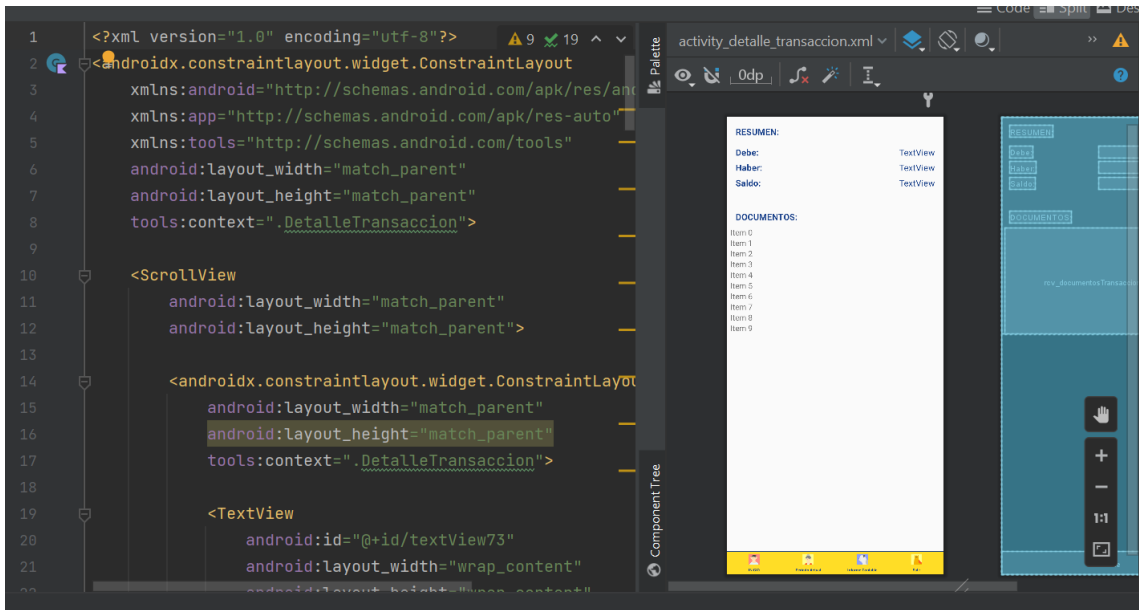


Figura 2.3.4.4.9 Implementación de la interfaz de Detalles de Transacción

Y la segunda, como parte de la interfaz del Detalles de los Documentos, como parte de esa información adicional otorgada al cliente de la aplicación, tal como se observa en la figura 2.3.4.4.10.

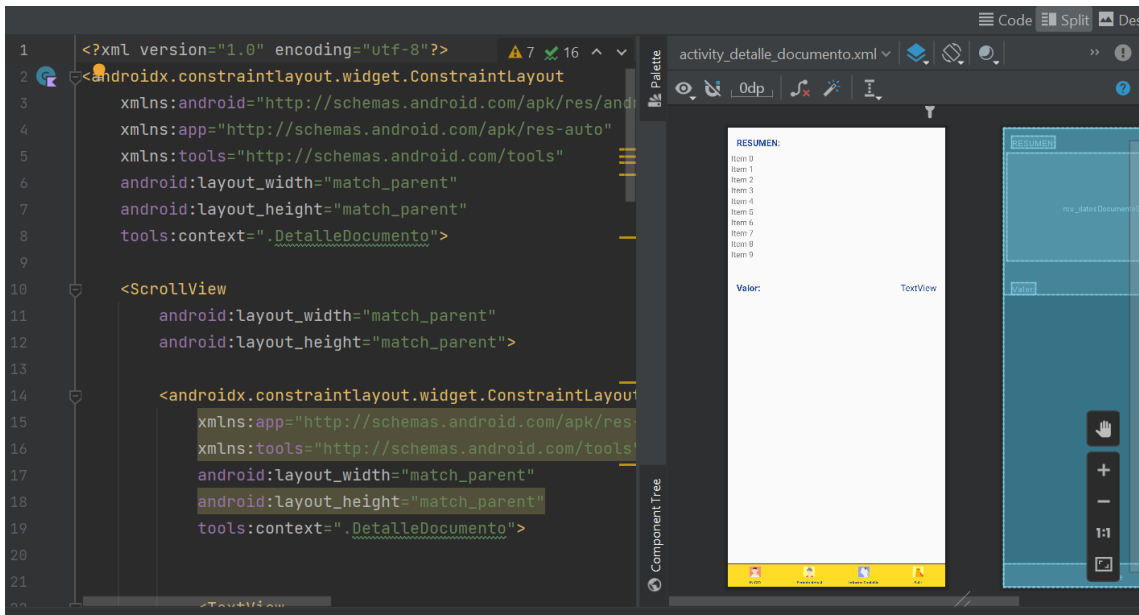


Figura 2.3.4.4.10 Implementación de la interfaz de Detalles de Documento

Con las interfaces definidas, también se ejecutó la implementación de la funcionalidad de algunas pantallas, más específicamente, la interfaz de Inicio de Sesión y la Página Principal de la aplicación.

Con ello, la funcionalidad de la interfaz de Inicio de Sesión descrita en la figura 2.3.4.3.3 ya está lista para cumplir con sus funciones (para observar su uso, revisar el Anexo X), las cuáles son:

- Permitir la identificación del usuario que ingresa a la aplicación.
- Autenticar y autorizar el acceso a la aplicación al usuario.

Por otro lado, también se supo implementar la funcionalidad de la Página Principal (ver figura 2.3.4.4.1), la cual cuenta con las siguientes características (para observar su uso, revisar el Anexo XI):

- Acceso a la página inicial y flujo de Cuentas.
- Acceso a la página inicial y flujo de Informe Contable.
- Agregar un Nuevo Periodo Contable.
- Acceder a un Periodo Contable vigente.

Aquí, si bien se hubiera ejecutado el flujo DevOps, se decidió aplazar su ejecución hasta llegar a completar al menos uno de los flujos, en ese sentido, la inclusión de estas funcionalidades fue integrada en el siguiente Sprint.

Sprint Review

Tabla 2.3.4.4.1 Resultados del Sprint Review

Pregunta	Resultado
¿Qué hice en el Sprint actual?	Desarrollo de las interfaces del programa, sin el flujo básico de las mismas. Desarrollo de la funcionalidad de Inicio de Sesión y la Página Principal.
¿Qué problemas tuve?	Limitaciones del computador en el desarrollo.
¿Qué hare el siguiente Sprint?	Desarrollo del flujo de la gestión de cuentas. Desarrollo del flujo de la gestión de periodos contables (asiento de transacciones). Configuración e integración de Firebase App-Distribution.

Resultados del Sprint

En la tabla 2.3.4.4.2, se encuentra un resumen sobre los resultados del presente Sprint, describiendo a mayor detalle las actividades que se planificaron dentro de la fase de Sprint Planning, además de describir el porcentaje de avance.

Tabla 2.3.4.4.2 Resultados del Sprint

Actividad	Ejecución
Desarrollo de la interfaz de la Página Principal	Sí
Desarrollo de la interfaz de Cuentas	Sí
Desarrollo de la interfaz de Cuentas de la Organización	Sí
Desarrollo de la interfaz de Cuentas de Terceros	Sí
Desarrollo de la interfaz de Periodo Contable	Sí
Desarrollo de la Interfaz de Asiento de Transacción	Sí
Desarrollo de la Interfaz de Documento en Transacción	Sí
Desarrollo de la Interfaz de Informe Contable	Sí
Desarrollo de la Interfaz de Detalle de Transacción	Sí
Desarrollo de la Interfaz de Detalle de Documento	Sí
Funcionalidad: ingreso de credenciales	Sí
Funcionalidad: problemas en inicio de sesión	No
Funcionalidad: autorización de credenciales	Sí
Funcionalidad: acceso a Interfaz de Cuentas	Sí
Funcionalidad: acceso a interfaz de Periodo Contable	Sí
Funcionalidad: visualización de datos de Periodos	Parcialmente
Funcionalidad: acceso a Interfaz de Informe Contable	Sí
Funcionalidad: agregar Periodo Contable	Parcialmente
PORCENTAJE CONSEGUIDO (Aproximado):	90%

2.3.4.5 Sprint 5

Sprint Planning

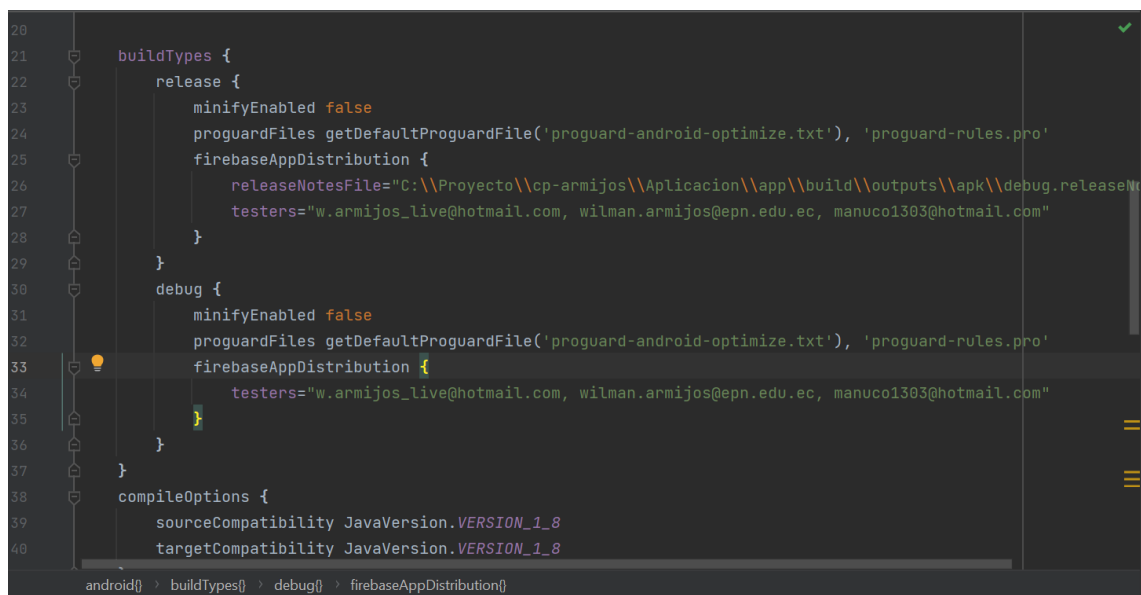
Para el quinto Sprint, se proponen como actividades las siguientes:

- Desarrollo del flujo de Gestión de Cuentas.
- Desarrollo del flujo de Gestión de Periodos.
- Integración de Firebase App-Distribution al proyecto.

Ejecución del Sprint

Si bien, ya se hizo un acercamiento inicial a los flujos DevOps, este fue definido exclusivamente para la etapa “build”, que fue destinada a la integración y compilación del proyecto dentro del repositorio remoto. Sin embargo, también era necesario configurar el flujo para la etapa “test”, destinada al lanzamiento del producto.

Para ello, lo primero que se debió hacer fue modificar el archivo build.gradle a nivel de la aplicación, agregando las líneas descritas dentro de la figura 2.3.4.5.1.



```
20
21 buildTypes {
22     release {
23         minifyEnabled false
24         proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
25         firebaseAppDistribution {
26             releaseNotesFile="C:\\Proyecto\\cp-armijos\\Aplicacion\\app\\build\\outputs\\apk\\debug.releaseNotes.txt"
27             testers="w.armijos_live@hotmail.com, wilman.armijos@epn.edu.ec, manuco1303@hotmail.com"
28         }
29     }
30     debug {
31         minifyEnabled false
32         proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
33         firebaseAppDistribution {
34             testers="w.armijos_live@hotmail.com, wilman.armijos@epn.edu.ec, manuco1303@hotmail.com"
35         }
36     }
37 }
38 compileOptions {
39     sourceCompatibility JavaVersion.VERSION_1_8
40     targetCompatibility JavaVersion.VERSION_1_8
41 }
```

Figura 2.3.4.5.1 Configuración de Firebase App-Distribution en el proyecto

Además, cabe destacar que en este punto se ejecutaron varios flujos DevOps, sin embargo, estos fueron realizados simplemente como pruebas de parte de la integración del uso de Firebase App-Distribution. La figura 2.3.4.5.1 muestra un resumen de todas las ejecuciones relacionadas con esa configuración y sus resultados.

```
20
21 buildTypes {
22     release {
23         minifyEnabled false
24         proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
25         firebaseAppDistribution {
26             releaseNotesFile="C:\\Proyecto\\cp-armijos\\Aplicacion\\app\\build\\outputs\\apk\\debug.release\\
27             testers="w.armijos_live@hotmail.com, wilman.armijos@epn.edu.ec, manuco1303@hotmail.com"
28         }
29     }
30     debug {
31         minifyEnabled false
32         proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
33         firebaseAppDistribution {
34             testers="w.armijos_live@hotmail.com, wilman.armijos@epn.edu.ec, manuco1303@hotmail.com"
35         }
36     }
37 }
38 compileOptions {
39     sourceCompatibility JavaVersion.VERSION_1_8
40     targetCompatibility JavaVersion.VERSION_1_8

```

Figura 2.3.4.5.2 Resultados de flujos DevOps con Firebase App-Distribution

Siguiendo con la implementación y lo planificado dentro del Sprint, lo que se ejecutó fue la implementación de las 3 interfaces que conforman el flujo de Cuentas.

La interfaz de Cuentas descrita en la figura 2.3.4.4.2 fue implementada con las siguientes funcionalidades (para observar su uso, revisar el Anexo XII):

- Agregar una nueva cuenta de la organización.
- Gestionar una cuenta de la organización existente.
- Agregar una nueva cuenta de un tercero.
- Gestionar una cuenta de un tercero existente.

La interfaz de Gestión de Cuentas de la Organización descrita en la figura 2.3.4.4.3 cuenta con las funciones descritas a continuación (para observar su uso, revisar el Anexo XIII):

- Visualizar información de una cuenta de la organización existente.
- Actualizar información de una cuenta de la organización existente.
- Ingresar información en la creación de una nueva cuenta de la organización.
- Eliminar una cuenta de la organización existente.

La interfaz de Gestión de Cuentas de Terceros descrita en la figura 2.3.4.4.4 cuentas con las mismas funciones que la interfaz descrita anteriormente, sin embargo, enfocada las cuentas de terceros (para observar su uso, revisar el Anexo XIV):

- Visualizar información de una cuenta de un tercero existente.
- Actualizar información de una cuenta de un tercero existente.

- Ingresar información en la creación de una nueva cuenta de un tercero.
- Eliminar una cuenta de un tercero existente.

De ese modo, aquí fue ejecutado el flujo DevOps descrito en el capítulo 2.2.2 para la etapa de “build”, cuyo resultado es mostrado en la figura 2.3.4.5.3, para el flujo de cuentas realizado y algunas pocas configuraciones iniciales del flujo de Gestión de las Transacciones.

Flujo de Cuentas y principio de Transacciones

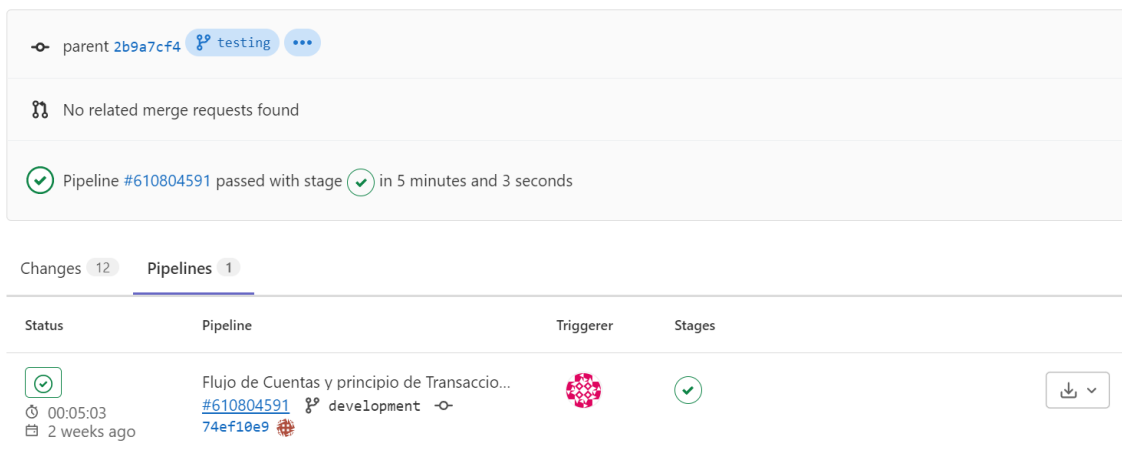


Figura 2.3.4.5.3 Ejecución de flujo DevOps (build)

Posterior a esto, se siguió con el proceso de desarrollo de otras funcionalidades dentro de este flujo, cuyos resultados se muestran en las figuras 2.3.4.5.4 y 2.3.4.5.5, ambas con algunos cambios pequeños dentro de la programación.

Flujo de Cuentas y principio de Transacciones

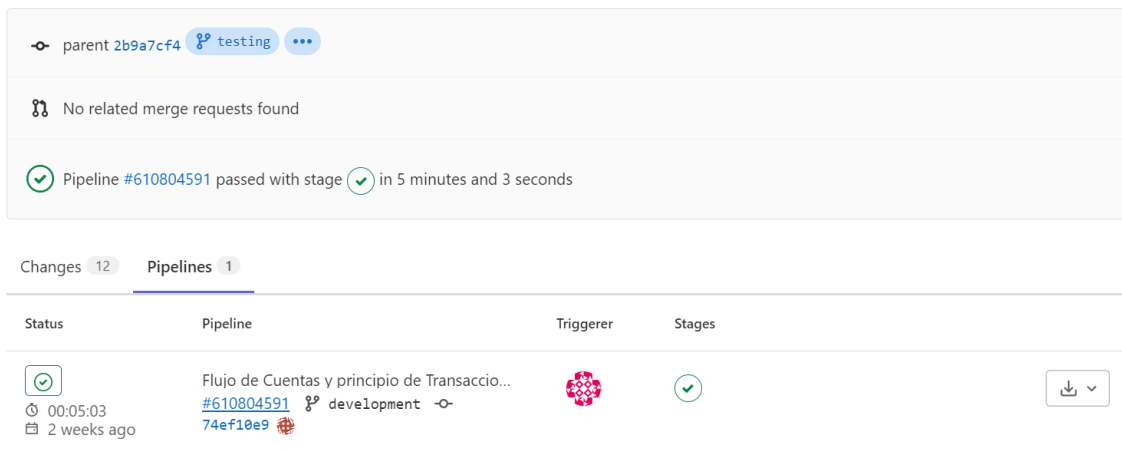


Figura 2.3.4.5.4 Ejecución de flujo DevOps (build)

Flujo de Cuentas y principio de Transacciones

The screenshot shows a GitHub Actions pipeline execution. At the top, it indicates the parent commit is '2b9a7cf4' and the workflow is 'testing'. Below this, it states 'No related merge requests found'. A green checkmark icon indicates that the pipeline '#610804591' passed with stage 'testing' in 5 minutes and 3 seconds. Below the pipeline summary, there are tabs for 'Changes' (12) and 'Pipeline' (1). The pipeline details table shows the following information:

Status	Pipeline	Triggerer	Stages
 00:05:03 2 weeks ago	Flujo de Cuentas y principio de Transaccio... #610804591 74ef10e9	 development	

Figura 2.3.4.5.5 Ejecución de flujo DevOps (build)

Sprint Review

Tabla 2.3.4.5.1 Resultados del Sprint Review

Pregunta	Resultado
¿Qué hice en el Sprint actual?	Desarrollo del flujo de Gestión de Cuentas de la Aplicación. Avances dentro del desarrollo del flujo de Gestión de Periodos Contables (transacciones). Integración parcial de Firebase App-Distribution (configuración del archivo YML y pruebas).
¿Qué problemas tuve?	Ninguno.
¿Qué hare el siguiente Sprint?	Terminar el desarrollo del flujo de Transacciones. Desarrollo del flujo de Gestión de Informes Contables.

Resultados del Sprint

En la tabla 2.3.4.5.2, se encuentra un resumen sobre los resultados del presente Sprint, describiendo a mayor detalle las actividades que se planificaron dentro de la fase de Sprint Planning, además de describir el porcentaje de avance.

Tabla 2.3.4.5.2 Resultados del Sprint

Actividad	Ejecución
Funcionalidad: visualizar Cuentas de la Organización	Sí

Funcionalidad: visualizar Cuentas de Terceros	Sí
Funcionalidad: creación de Cuenta de la Organización	Sí
Funcionalidad: creación de Cuenta de un Tercero	Sí
Funcionalidad: acceder a Cuenta de la Organización	Sí
Funcionalidad: acceder a Cuenta de la Organización	Sí
Funcionalidad: creación de Cuenta de la Organización	Sí
Funcionalidad: eliminar Cuenta de la Organización	Sí
Funcionalidad: eliminar Cuenta de un Tercero	Sí
Funcionalidad: agregar Periodo Contable	Sí
Funcionalidad: acceder a Periodo Contable actual	Sí
Funcionalidad: negar acceso a Periodo anterior	No
Funcionalidad: visualización de datos del Periodo	No
Funcionalidad: acceso a Anotar Transacción	Sí
Funcionalidad: ingreso de datos de la Transacción	No
Funcionalidad: visualización de datos de la Transacción	No
Funcionalidad: acceso a creación de Cuenta de la Organización	No
Funcionalidad: acceso a creación de Cuenta de Tercero	No
Funcionalidad: acceso a Documento de Transacción	Sí
Funcionalidad: anotar Transacción en el Periodo	No
Funcionalidad: ingreso de datos del Documento	No
Funcionalidad: agregar Documento en la Transacción	No
Configuración de Firebase App-Distribution en el entorno de programación	Parcialmente
Actualización del archivo de Configuración	Parcialmente
PORCENTAJE CONSEGUIDO (Aproximado):	60%

2.3.4.6 Sprint 6

Sprint Planning

Para el sexto Sprint, se ha planificado el siguiente conjunto de actividades:

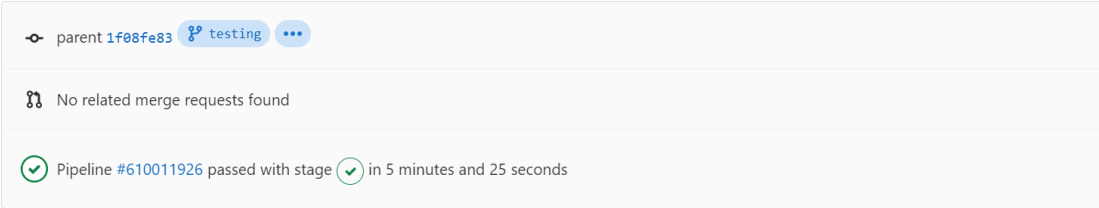
- Terminar configuración de App-Distribution y los flujos DevOps.
- Implementación de todos los flujos de las interfaces.

Ejecución del Sprint

El comienzo de este Sprint fue definido por la completa y correcta configuración e integración de Firebase App-Distribution dentro de los flujos DevOps, de modo que a partir de ahora es posible realizar ejecuciones dentro de la etapa de “test” de la forma en que se había previsto, es decir, para publicar la aplicación dentro de Firebase.

Así, la figura 2.3.4.6.1 muestra el resultado de la ejecución del primer flujo DevOps con la etapa de “test” también siendo reproducida.

Configuración Gradle - FirebaseAppDistribution - 6



The screenshot shows a pipeline execution summary. At the top, it indicates the parent pipeline ID '1f08fe83' and the current stage 'testing'. Below this, it states 'No related merge requests found'. A green checkmark icon indicates that the pipeline '#610011926' passed with the 'testing' stage in 5 minutes and 25 seconds. Below the summary, there are tabs for 'Changes' (1) and 'Pipelines' (2). The 'Pipelines' tab is active, showing a table of pipeline runs.










Status	Pipeline	Triggerer	Stages
 00:05:25 2 weeks ago	Configuración Gradle - FirebaseAppDistrib... #610011926 testing 2b9a7cf4		 
 00:00:20 2 weeks ago	Configuración Gradle - FirebaseAppDistrib... #610011867 development 2b9a7cf4		  

Figura 2.3.4.6.1 Ejecución de flujo DevOps (build, test)

Además de esta inclusión, se logró desarrollar todos los flujos y funcionalidades restantes, es decir, los flujos de Gestión de Periodos Contables y de Gestión de Informe Contable, sin embargo, estos se presentan con algunos errores hasta el final del presente Sprint, que serán corregido a lo largo del siguiente y último.

Con esto en mente, en breve se describen todas las características de cada una de las interfaces desarrolladas.

La primera, fue la interfaz de Periodo Contable, descrita en la figura 2.3.4.4.5, que cuenta con las siguientes funciones (para observar su uso, revisar el Anexo XV):

- Observar información de dicho periodo.
- Agregar un nuevo asiento (transacción) dentro del presente periodo.

Siguiendo este flujo, la interfaz de Transacción descrita en la figura 2.3.4.4.6, cuenta con las funciones (para observar su uso, revisar el Anexo XVI):

- Ver información de la transacción que es anotada.
- Agregar un nuevo documento sobre la transacción.
- Anotar la transacción dentro del periodo que lo contiene.

Y, por último, se tiene la interfaz de Documento descrita en la figura 2.3.4.4.7, que permite al usuario ejecutar las operaciones descritas a continuación (para observar su uso, revisar el Anexo XVII):

- Elegir un documento contable para que pueda ser anotado.
- Asignar el valor procesado por dicho documento a ser anotado.
- Guardar el documento dentro de la transacción que lo contiene.

Por otro lado, se ejecutó también la implementación del flujo de Gestión de Informe Contable, cuyas interfaces serán descritas a continuación.

La interfaz de Informe Contable que fue descrita en la figura 2.3.4.4.8, cuyas funcionalidades son (para observar su uso, revisar el Anexo XVIII):

- Visualización un pequeño resumen de acuerdo con los parámetros del informe.
- Visualización de transacciones según los parámetros del informe.
- Opción de filtro de resultados de las transacciones según fecha.
- Opción de filtro de resultados de las transacciones según cuenta del cliente.
- Opción de filtro de resultados de las transacciones según cuenta contable.

Luego, la interfaz que muestra el detalle de la Transacción, que muestra la figura 2.3.4.4.9, cuenta con las siguientes funcionalidades (para observar su uso, revisar el Anexo XVIII):

- Visualización de la información de una Transacción.

Por último, se cuenta con la interfaz de detalle del Documento, que muestra la figura 2.3.4.4.10, cuyas funcionalidades son (para observar su uso, revisar el Anexo XVIII):

- Visualización de la información de un Documento.

Con todas estas funcionalidades implementadas, ya fue posible realizar la ejecución del flujo DevOps, cuyo resultado se muestra en la figura 2.3.4.6.2.

Pruebas 1 - Informe Contable

The screenshot shows a Jenkins pipeline execution summary. At the top, it indicates the parent build ID '0e1e4fb5' and the current build is in the 'testing' phase. Below this, it states 'No related merge requests found'. A green checkmark icon indicates that the pipeline '#614796577' passed with stage '#614796577' in 4 minutes and 44 seconds. Below the summary, there are tabs for 'Changes' (22) and 'Pipelines' (1). The main table shows the pipeline details:

Status	Pipeline	Triggerer	Stages
 00:04:44 1 week ago	Pruebas 1 - Informe Contable #614796577 development 7e4e1005		

Figura 2.3.4.6.2 Ejecución de flujo DevOps (build)

Con esto, simplemente se dejaron algunos detalles de la programación y la ejecución de flujos finales para el último Sprint.

Sprint Review

Tabla 2.3.4.6.1 Resultados del Sprint Review

Pregunta	Resultado
¿Qué hice en el Sprint actual?	Configuración completa de Firebase App-Distribution. Implementación de todos los distintos flujos de la aplicación (falta de funcionalidad en la presentación de los datos de ciertas pantallas).
¿Qué problemas tuve?	Ninguno.
¿Qué hare el siguiente Sprint?	Culminar con los detalles de funcionalidad de la aplicación. Lanzamiento de la aplicación.

Resultados del Sprint

En la tabla 2.3.4.6.2, se encuentra un resumen sobre los resultados del presente Sprint, describiendo a mayor detalle las actividades que se planificaron dentro de la fase de Sprint Planning, además de describir el porcentaje de avance.

Tabla 2.3.4.6.2 Resultados del Sprint

Actividad	Ejecución
Funcionalidad: negar acceso a Periodo anterior	Sí
Funcionalidad: visualización de datos del Periodo	Sí
Funcionalidad: acceso a Anotar Transacción	Sí
Funcionalidad: ingreso de datos de la Transacción	Sí
Funcionalidad: visualización de datos de la Transacción	Sí
Funcionalidad: acceso a creación de Cuenta de la Organización	No
Funcionalidad: acceso a creación de Cuenta de Tercero	No
Funcionalidad: acceso a Documento de Transacción	Sí
Funcionalidad: anotar Transacción en el Periodo	Sí
Funcionalidad: ingreso de datos del Documento	Sí
Funcionalidad: agregar Documento en la Transacción	Sí
Configuración de Firebase App-Distribution en el entorno de programación	Sí
Actualización del archivo de Configuración	Sí
Funcionalidad: problemas de inicio de sesión	No
Funcionalidad: eliminar una cuenta de la Organización	No
Funcionalidad: eliminar una cuenta de un Tercero	No
Funcionalidad: visualizar Transacciones en Informe Contable	Sí
Funcionalidad: acceso a Detalle de Transacción	Sí
Funcionalidad: visualización de datos de Detalle de Transacción	Sí
Funcionalidad: acceso a Detalle de Documento	Sí
Funcionalidad: visualización de datos de Detalle de Documento	Parcialmente
Funcionalidad: filtro por Fechas	Parcialmente
Funcionalidad: filtro por Cuenta de Cliente	Parcialmente
Funcionalidad: filtro por Cuenta Contable	Parcialmente
PORCENTAJE CONSEGUIDO (Aproximado):	70%

2.3.4.7 Sprint 7

Sprint Planning

Para el último Sprint, definido como el Sprint 7, se planifican las actividades:

- Culminar con los detalles de funcionalidad de la aplicación.
- Lanzamiento completo de la aplicación a través de Firebase App-Distribution.

Ejecución del Sprint

En este Sprint, lo que se ejecutaron fueron varias pruebas, agregados, eliminaciones y modificaciones de código respecto a temas de la funcionalidad y congruencia con las interfaces, así como algunas validaciones de campos. Con eso, simplemente resta la ejecución de un flujo DevOps final, que refleje los cambios, cuyo resultado se muestra en la figura 2.3.4.7.1.

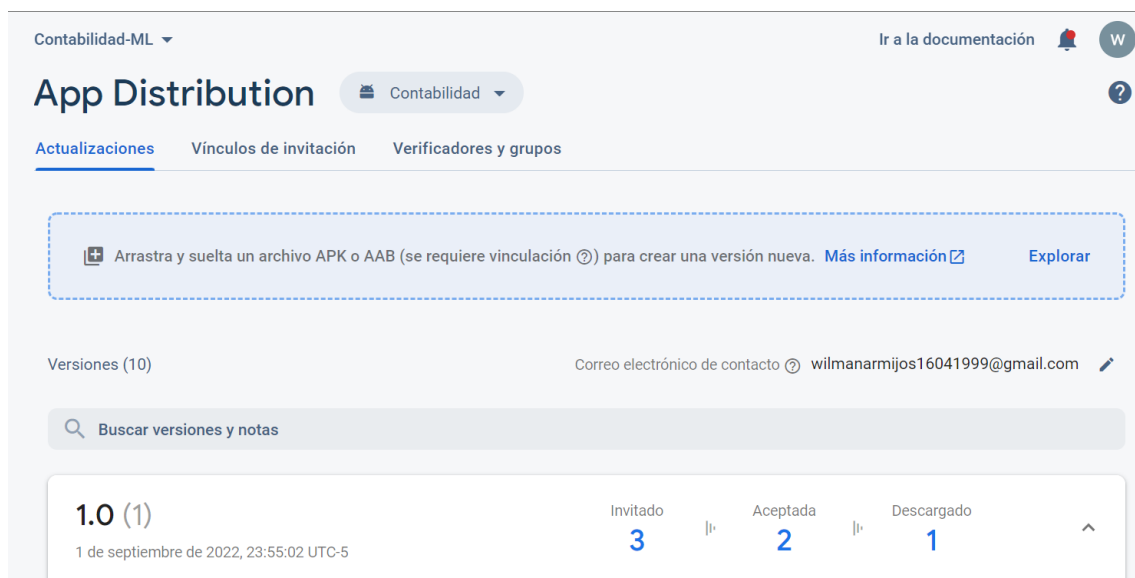


Figura 2.3.4.7.1 Ejecución de flujo DevOps (build, test)

Sprint Review

Tabla 2.3.4.7.1 Resultados del Sprint Review

Pregunta	Resultado
¿Qué hice en el Sprint actual?	Implementación y finalización de todos los flujos, así como ejecución de pruebas y control de errores. Lanzamiento de la aplicación.
¿Qué problemas tuve?	Ninguno.
¿Qué hare el siguiente Sprint?	No existe siguiente Sprint.

Resultados del Sprint

En la tabla 2.3.4.7.2, se encuentra un resumen sobre los resultados del presente Sprint, describiendo a mayor detalle las actividades que se planificaron dentro de la fase de Sprint Planning, además de describir el porcentaje de avance.

Tabla 2.3.4.7.2 Resultados del Sprint

Actividad	Ejecución
Funcionalidad: acceso a creación de Cuenta de la Organización	Sí
Funcionalidad: acceso a creación de Cuenta de Tercero	Sí
Funcionalidad: problemas de inicio de sesión	Sí
Funcionalidad: eliminar una cuenta de la Organización	Sí
Funcionalidad: eliminar una cuenta de un Tercero	Sí
Funcionalidad: visualización de datos de Detalle de Documento	Sí
Funcionalidad: filtro por Fechas	Sí
Funcionalidad: filtro por Cuenta de Cliente	Sí
Funcionalidad: filtro por Cuenta Contable	Sí
Desarrollo de menú de navegación	Sí
Control de dispositivos compatibles	No
Control de ingreso de datos	Parcialmente
Control de flujo de la aplicación	Parcialmente

Controles extra	Parcialmente
PORCENTAJE CONSEGUIDO (Aproximado):	85%

2.3.5 Pruebas de Usabilidad

Método de Evaluación de Usabilidad

Como una parte importante dentro del desarrollo de la aplicación, es necesario ejecutar pruebas respecto a cómo un cliente real siente el uso de dicha aplicación, en relación con su experiencia en cuanto a las funcionalidades y las características generales de la aplicación.

Para ejecutar estas pruebas de usabilidad, se hará uso de un sistema conocido como la Escala de Usabilidad del Sistema (SUS por sus siglas en inglés, System Usability Scale), el cuál será descrito brevemente a continuación.

El uso de SUS puede ser descrito como una especie de cuestionario que incluye 10 ítems, cada uno de los cuáles está centrado en describir un efecto o consecuencia resultados de la experiencia del usuario después de haber usado una aplicación, cada uno de ellos siendo medidas a través de una escala del 1 al 5, de ahí el nombre de dicha técnica [25].

Las 10 ideas o sentencias presentadas al usuario están descritas en la tabla 2.3.5.1, y especificadas en el orden mostrado dentro de la misma, el cuál es importante tener en cuenta pues será utilizado posteriormente.

Tabla 2.3.5.1 Preguntas del método SUS [25]

Número de Pregunta	Pregunta (Sentencia)
1	Pienso que me gustaría utilizar este sistema frecuentemente
2	Encuentro el sistema innecesariamente complicado
3	Pienso que el sistema fue fácil de usar
4	Pienso que necesitaría ayuda de un técnico para poder utilizar el sistema
5	Pienso que las funciones del sistema fueron bien integradas
6	Pienso que existe mucha inconsistencia dentro del sistema

7	Pienso que la mayoría de las personas aprenderían a utilizar este sistema muy rápido
9	Sentí mucha confianza al utilizar el sistema
10	Necesité aprender muchas cosas antes de usar el sistema

Conocidas estas preguntas, lo que sigue es conocer de qué forma se obtienen los resultados finales, dado que este método no se basa solamente en sumar los puntajes y nada más.

Aquí, es donde se retomará la parte de las posiciones de las preguntas, pues con ellas lo que se ejecuta es una especie de algoritmo, que será descrito a continuación, con el cuál se otorgan los valores finales de cada evaluación ejecutada:

- Al puntaje asignado a una pregunta impar, se le resta 1 (Ejemplo: Si el valor asignado fue 5, es transformado en 4, resultado de 5-1).
- Al puntaje asignado a una pregunta par, se resta dicho puntaje de 5 (Ejemplo: Si el valor asignado fue 3, es transformado en 2, resultado de 5-3).
- Luego, todos los puntajes son sumados una vez hechas las transformaciones antes descritas.
- Y al final, se multiplica dicho valor por 2.5, provocando que las puntuaciones devueltas se encuentren en un rango entre 40 y 100 [25].

Para finalizar, este método define algunos valores o rangos de valores como límites para definir cuando el resultado de una evaluación se puede considerar buena, regular o mal, algo que se muestra en la tabla 2.3.5.2.

Tabla 2.3.5.2 Clasificación de resultados SUS

Rangos de valores	Clasificación del resultado
40.00 – 50.99	Deficiente
51.00 – 67.99	Regular
68.00 – 80.29	Buena
80.30 – 100.00	Sobresaliente

Así, una vez explicado cómo funciona este método, entonces podemos observar cómo fue ejecutado el mismo dentro del presente proyecto. Para ello, fue tomado el archivo APK generado dentro del flujo “test” descrito anteriormente y enviado a cada una de las personas que harían uso de la aplicación, en tanto, la forma para registrar

las respuestas de esta encuesta SUS descrita previamente, fue a través de un formulario realizado a través de la herramienta Microsoft Forms (para observar el cuestionario, revisar el Anexo XXI), para lo cual fue enviado el enlace generado por dicha aplicación a los usuarios para el registro de sus opiniones y resultados tras el uso de la aplicación, una vista previa del cuestionario, se puede observar en la figura 2.3.5.1.

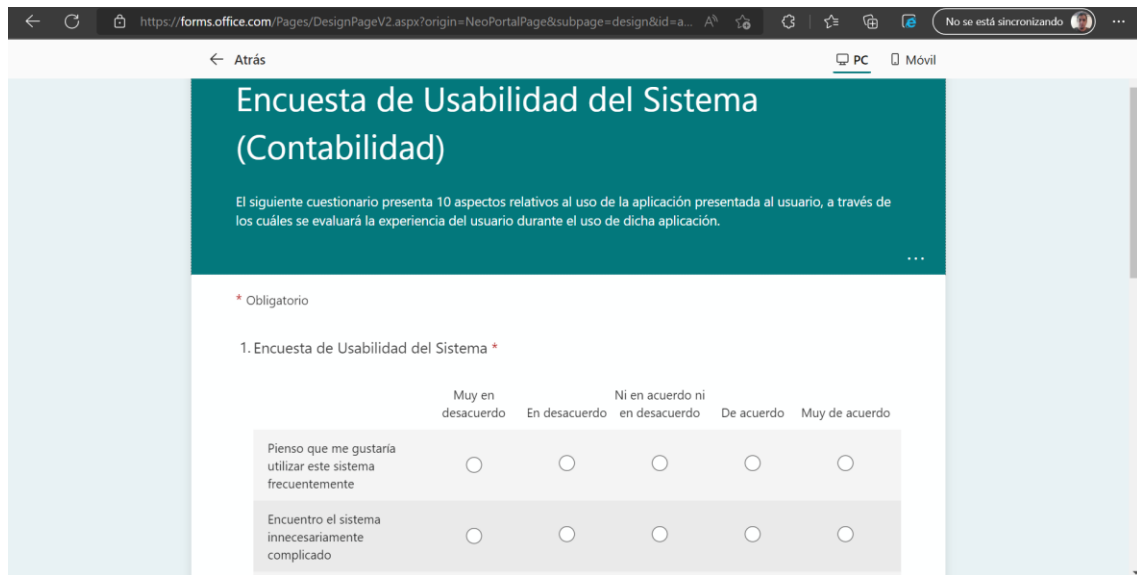


Figura 2.3.5.1 Vista previa del cuestionario SUS

Para finalizar, y tomando en cuenta el formato de la encuesta ejecutada mostrado en la figura 2.3.5.1, la tabla 2.3.5.3 muestra las equivalencias de dicho cuestionario con la escala del 1 al 5 descrito dentro del método SUS.

Tabla 2.3.5.3 Equivalencia de encuesta y escala SUS

Descripción	Escala (1 a 5)
Muy en desacuerdo	1
En desacuerdo	2
Ni en acuerdo ni en desacuerdo	3
De acuerdo	4
Muy de acuerdo	5

Resultado de las Pruebas de Usabilidad

Una vez ejecutadas las pruebas, las cuales fueron desarrolladas simplemente a través del envío del archivo APK a cada una de las personas que fueron parte de estas pruebas, 9 en total, obteniendo los resultados que se muestran a continuación.

Los resultados del primer individuo se muestran en la tabla 2.3.5.4. A partir de dichos resultados, podemos aplicar la fórmula descrita anteriormente para encontrar su calificación, que como se observa en la misma figura, otorga un puntaje de 85.00 sobre 100.00.

Tabla 2.3.5.4 Resultados del primero usuario

Número de Pregunta	Respuesta (Escala del 1 al 5)
1	4
2	1
3	5
4	2
5	5
6	2
7	5
8	2
9	4
10	2
Puntaje Total:	85.00/100.00

Los resultados del segundo individuo se muestran en la tabla 2.3.5.5. A partir de dichos resultados, podemos aplicar la fórmula descrita anteriormente para encontrar su calificación, que como se observa en la misma figura, otorga un puntaje de 87.50 sobre 100.00.

Tabla 2.3.5.5 Resultados del segundo usuario

Número de Pregunta	Respuesta (Escala del 1 al 5)
1	5
2	2

3	5
4	4
5	5
6	1
7	5
9	1
8	5
10	2
Puntaje Total:	87.50/100.00

Los resultados del tercer individuo se muestran en la tabla 2.3.5.6. A partir de dichos resultados, podemos aplicar la fórmula descrita anteriormente para encontrar su calificación, que como se observa en la misma figura, otorga un puntaje de 97.50 sobre 100.00.

Tabla 2.3.5.6 Resultados del tercer usuario

Número de Pregunta	Respuesta (Escala del 1 al 5)
1	5
2	1
3	5
4	1
5	5
6	1
7	5
8	1
9	5
10	2
Puntaje Total:	97.50/100.00

Los resultados del cuarto individuo se muestran en la tabla 2.3.5.7. A partir de dichos resultados, podemos aplicar la fórmula descrita anteriormente para encontrar su calificación, que como se observa en la misma figura, otorga un puntaje de 75.00 sobre 100.00.

Tabla 2.3.5.7 Resultados del cuarto usuario

Número de Pregunta	Respuesta (Escala del 1 al 5)
1	4
2	4
3	5
4	2
5	4
6	2
7	5
8	2
9	4
10	2
Puntaje Total:	75.00/100.00

Los resultados del quinto individuo se muestran en la tabla 2.3.5.8. A partir de dichos resultados, podemos aplicar la fórmula descrita anteriormente para encontrar su calificación, que como se observa en la misma figura, otorga un puntaje de 60.00 sobre 100.00.

Tabla 2.3.5.8 Resultados del quinto usuario

Número de Pregunta	Respuesta (Escala del 1 al 5)
1	4
2	4
3	3
4	2
5	4
6	3
7	4
8	1
9	3
10	4
Puntaje Total:	60.00/100.00

Los resultados del sexto individuo se muestran en la tabla 2.3.5.9. A partir de dichos resultados, podemos aplicar la fórmula descrita anteriormente para encontrar su calificación, que como se observa en la misma figura, otorga un puntaje de 72.50 sobre 100.00.

Tabla 2.3.5.9 Resultados del sexto usuario

Número de Pregunta	Respuesta (Escala del 1 al 5)
1	4
2	2
3	4
4	3
5	4
6	1
7	3
8	1
9	5
10	4
Puntaje Total:	72.50/100.00

Los resultados del séptimo individuo se muestran en la tabla 2.3.5.10. A partir de dichos resultados, podemos aplicar la fórmula descrita anteriormente para encontrar su calificación, que como se observa en la misma figura, otorga un puntaje de 82.50 sobre 100.00.

Tabla 2.3.5.10 Resultados del séptimo usuario

Número de Pregunta	Respuesta (Escala del 1 al 5)
1	5
2	2
3	4
4	2
5	5
6	2
7	4
8	2

9	5
10	2
Puntaje Total:	82.50/100.00

Los resultados del octavo individuo se muestran en la tabla 2.3.5.11. A partir de dichos resultados, podemos aplicar la fórmula descrita anteriormente para encontrar su calificación, que como se observa en la misma figura, otorga un puntaje de 85.00 sobre 100.00.

Tabla 2.3.5.11 Resultados del octavo usuario

Número de Pregunta	Respuesta (Escala del 1 al 5)
1	5
2	1
3	1
4	1
5	5
6	2
7	5
8	1
9	5
10	2
Puntaje Total:	85.00/100.00

Los resultados del noveno individuo se muestran en la tabla 2.3.5.12. A partir de dichos resultados, podemos aplicar la fórmula descrita anteriormente para encontrar su calificación, que como se observa en la misma figura, otorga un puntaje de 80.00 sobre 100.00.

Tabla 2.3.5.12 Resultados del noveno usuario

Número de Pregunta	Respuesta (Escala del 1 al 5)
1	4
2	1
3	5
4	2

5	3
6	1
7	4
8	2
9	5
10	3
Puntaje Total:	80.00/100.00

2.3.6 Despliegue

Una vez finalizado el proceso de implementación y desarrollo de la aplicación, entonces se puede pasar a la fase de lanzamiento o despliegue.

En este caso, y como se mencionó previamente, se hizo uso de una de las características del servicio de Google Firebase App-Distribution. A través de esta es posible lanzar y enviar el producto final, es decir, el APK hacia posibles personas que hagan la función de “Testers”, pero también permite distribuir el producto hacia los clientes, en caso de ser necesario.

En ese sentido, el uso de Firebase App-Distribution, en conjunto con el flujo de trabajo definido a través de la metodología DevOps que fue descrito en el capítulo 2.2.2 del presente trabajo en su etapa de “test”, dicho ejecutable o APK fue enviado al servicio de Firebase App-Distribution, donde, según las personas que fueron descritas o sean descritas como parte del proyecto, pueden acceder a dicho archivo y usarlo dentro de un entorno real, así, esto se muestra en la figura 2.3.5.1.

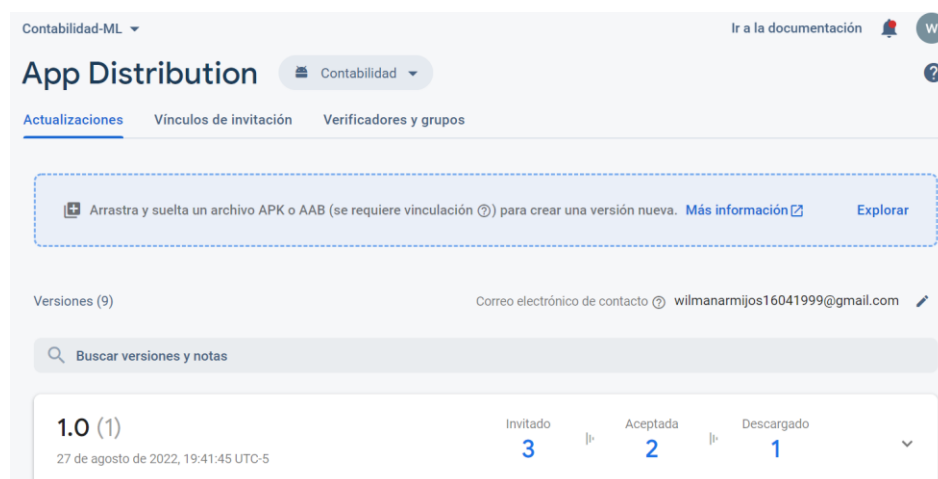


Figura 2.3.5.1 Lanzamiento de la aplicación en Firebase App-Distribution

Por último, simplemente como una forma de verificación, la figura 2.3.5.3 muestra la aplicación en ejecución dentro de un dispositivo real (Xiaomi Redmi Note 8).

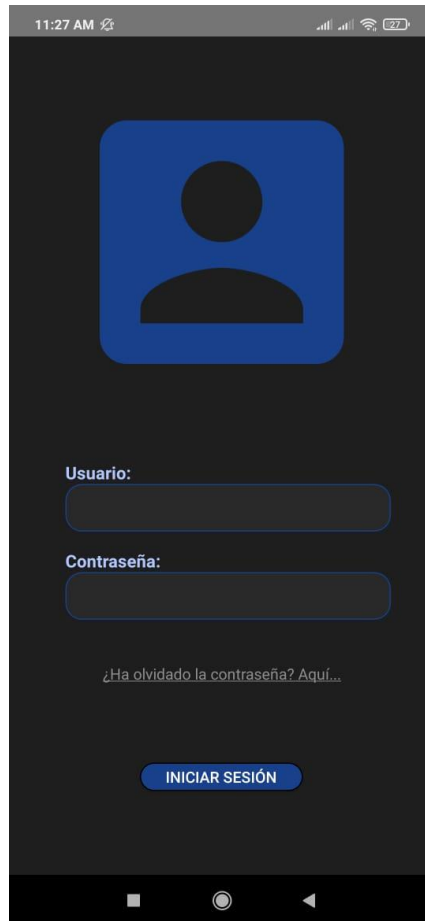


Figura 2.3.5.2 Ejecución en un dispositivo Android real

Para finalizar, si bien esta aplicación no puede ser directamente descargada o accedida dentro del repositorio descrito en el anexo I, en el anexo XXI se encuentra un enlace con el documento “Manual de Desarrollador.pdf”, en el cual se describen algunas instrucciones de configuración y uso de este producto. Además, el mismo enlace al documento se encuentra dentro del repositorio, dentro del archivo “Instrucciones.txt”, del propio anexo I.

Además, como parte del producto final, en el anexo XXII se encuentra el enlace de un video donde se ejecuta la última versión de la aplicación dentro de un dispositivo real, en relación con lo mostrado en la figura 2.3.5.2.

3 RESULTADOS

Retomando el objetivo del proyecto, centrado en la comparación entre el desarrollo a través de flujos DevOps seguros y sin el uso de estos, se presentan 4 métricas y/o modos de medir de alguna forma los resultados obtenidos, así:

- Número de flujos completados con éxito, con error o canceladas.
- Tiempo de ejecución de los flujos.
- Número de pasos necesarios para completar un flujo.
- Porcentaje de tareas completadas dentro de un Sprint en comparación con las tareas planificadas.

Con ello, se han cuantificado los resultados de los flujos DevOps ejecutados y han sido categorizados en función de dicho resultado, es decir:

- Flujos ejecutados de forma completa y exitosa.
- Flujos que han fallado, sea por errores en la configuración del flujo o por errores durante su ejecución.
- Flujos que han sido cancelados.

La tabla 3.1 muestra la cuantificación de los flujos de acuerdo sus resultados, además de una pequeña división según la etapa en la que se obtuvieron dichos resultados.

Tabla 3.1 Ocurrencias de flujos DevOps según su resultado

Etapa del Flujo (Build, Test)	Resultado del Flujo (Exitoso, Fallido, Cancelado)	Número de ocurrencias
Build	Exitoso	24
	Fallido	7
	Cancelado	3
Test	Exitoso	12
	Fallido	4
	Cancelado	1

La mayoría de los flujos fallidos resultan de las etapas tempranas de implementación de estos, de hecho, de los últimos 25 flujos ejecutados, solamente uno tuvo un resultado fallido. Con esto, podemos descubrir que si bien, la configuración y correcta

implementación de los flujos puede provocar problemas y confusiones dentro del equipo de trabajo en etapas tempranas, conforme las personas logran familiarizarse con el proceso este se transforma en un aliado muy poderoso, pues pasa de ser algo de lo que la gente se debe preocupar, a una herramienta de la cual sacar provecho.

A través de la Tabla 3.1, la figura 3.1 describe a través de porcentajes cuál fue la cantidad de flujos en clasificación de su resultado, independiente de la etapa.

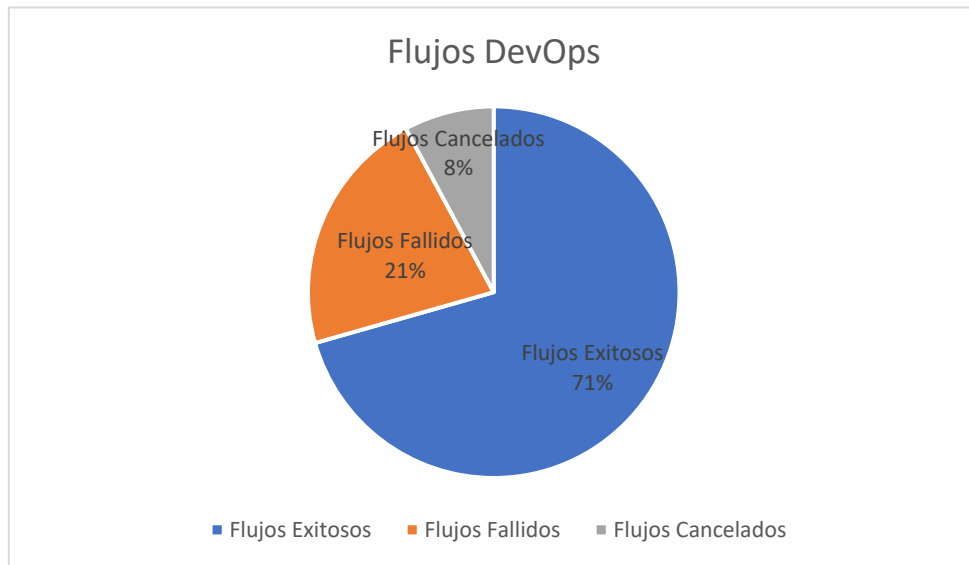


Figura 3.1. Resultados de los Flujos DevOps ejecutados

Luego, además de conocer el número, otro aspecto importante a medir representa los tiempos de ejecución de los flujos ejecutados a través de DevOps, cuyo resultado fue exitoso. Se presenta un resumen de la información de estos tiempos en la tabla 3.2.

Tabla 3.2 Tiempos de Ejecución de Flujos DevOps

Etapa del Flujo (Build, Test)	Resultado del Flujo (Exitoso, Fallido, Cancelado)	Tiempo Promedio de Ejecución [s] (Aproximación)
Build	Exitoso	340
	Fallido	143*
	Cancelado	27
Test	Exitoso	335
	Fallido	236*
	Cancelado	Sin ocurrencias

*No toma en cuenta flujos que no empezaron su ejecución.

Si bien, estos tiempos aproximados dependen mucho en calidad de la cantidad de tareas a ser ejecutadas, o de la cantidad de información a ser procesada por dicha tarea, estos ofrecen una visión general de los tiempos que pueden llegar a tomar las ejecuciones de los flujos DevOps.

Así, si tomamos el promedio de tiempos de ejecución como un valor recurrente, entonces los procesos de éxito toman alrededor de 335 a 340 segundos (algo más de 5 minutos y medio), mientras la ejecución de proceso que resultan en un error toma, entre las dos etapas, un tiempo promedio de 190 segundos o 3 minutos y medio. Mientras, los flujos cancelados toman tiempos mínimos o no existen ocurrencias, esto dado porque su cancelación requiere del momento en que el usuario decida cancelar su ejecución manualmente.

Por otro lado, respecto a lo que implica el desarrollo de un flujo DevOps, algo que se puede medir es la cantidad de operaciones esenciales a ser ejecutadas para completar un flujo, en comparación con la cantidad de operaciones que se pueden ejecutar al no usar dicho flujo. En ese sentido, la tabla 3.3 muestra un resumen de las operaciones para cada caso:

Tabla 3.3 Número de operaciones para el flujo DevOps y un flujo común

Operaciones básicas CON uso de flujo DevOps	Operaciones básicas SIN uso de flujo DevOps
Desarrollo de código	Desarrollo de código
Envío de cambios al repositorio local (computador)	-
Envío de cambios al repositorio remoto (GitLab)	-
Dentro del flujo DevOps - build: Compilación del proyecto	Compilación del proyecto
Dentro del flujo DevOps - build: Creación del archivo APK	Creación del archivo APK
Dentro del flujo DevOps – test: Envío del APK a Firebase App-Distribution	Envío del APK a Firebase App-Distribution o algún servicio
Número de operaciones: 6	Número de operaciones: 4

Como se ha mencionado previamente durante la ejecución del proyecto, el hecho de que el equipo de trabajo empleado sea conformado por una sola persona, provoca que muchas de las operaciones puedan ser omitidas dentro del flujo sin DevOps, en comparación del flujo con uso de DevOps.

Sin embargo, aunque las operaciones son menores, el tiempo destinado a la concesión de estos objetivos (a excepción del desarrollo), depende de gran manera de las capacidades del desarrollador o desarrolladores para que ejecuten dichas actividades. Si bien la inclusión de flujos DevOps seguros agiliza estas operaciones al ejecutarlas de forma automática, esto depende en gran medida del resultado esperado con dichas operaciones. Como ejemplo: la generación de un APK puede ser necesaria de ser enviada hacia un tercero, o puede bastar con guardarla dentro del computador, según el caso, puede que la implementación del flujo DevOps facilita la operación (primer caso) o la retrase o complique más de lo necesario (segundo caso), por lo que es algo que se debe medir a mayor profundidad y precisión.

Luego, un punto dentro de la implementación, se pueden describir la cantidad de avance logrado por cada Sprint, en relación con las actividades propuestas dentro de la fase de Sprint Planning. Con ello, la tabla 3.4 muestra el porcentaje de actividades completadas, de forma aproximada, en relación con lo planificado para ser ejecutado en dicho Sprint.

Tabla 3.4 Porcentaje de actividades completadas en el Sprint

Número de Sprint	Porcentaje de avance en relación con la planificación
1	90.00 %
2	40.00 %
3	70.00 %
4	90.00 %
5	60.00 %
6	70.00 %
7	85.00 %
Promedio:	72.14 %

Respecto a Scrum, de forma breve, tenemos que el porcentaje de avance promedio por cada Sprint es de alrededor de 72.14% que, si bien es un promedio decente, no

es suficiente para decir que la ejecución fue la mejor, sin embargo, en este caso, el problema recae en el desarrollador, antes que en la metodología en sí.

Ahora, respecto a las pruebas de usabilidad ejecutadas, es posible extraer información relevante de cada una de ellas y de su total. Tomando en cuenta que estas pruebas de usabilidad son una fuente de retroalimentación y medición de la capacidad de la aplicación, a continuación, se recopila la información de la prueba a través de metodología SUS.

La tabla 3.5 muestra un resumen de todos los resultados obtenidos, en relación con los puntajes calculados en base a las respuestas, además de mostrar cuál es su equivalencia de la tabla 2.3.5.2.

Tabla 3.5 Resultados totales de SUS

Encuesta	Puntaje Total	Clasificación
1	85.00	Bueno
2	87.50	Sobresaliente
3	97.50	Sobresaliente
4	75.00	Bueno
5	60.00	Regular
6	72.50	Bueno
7	82.50	Bueno
8	85.00	Sobresaliente
9	80.00	Bueno
Promedios:	80.55	Bueno

De la tabla anterior, podemos ver que la mayoría de las encuestas entrega un resultado donde la usabilidad de la aplicación es calificada como buena o sobresaliente, exceptuando los resultados del usuario número 5, cuya calificación entrega un valor regular a esta.

Del mismo modo, se pueden extraer los resultados obtenidos divididos por cada ítem descrito en la tabla 2.3.5.1, pues a través de cada uno de estos ítems será posible analizar las posibles deficiencias o fallas que presente la aplicación, para la ejecución de sus debidos ajustes o correcciones.

Con esto en mente, la figura 3.2 muestra el promedio de calificación otorgado a cada una de las preguntas o ítems de la tabla 2.3.5.1 correspondientes a la escala de la metodología SUS.

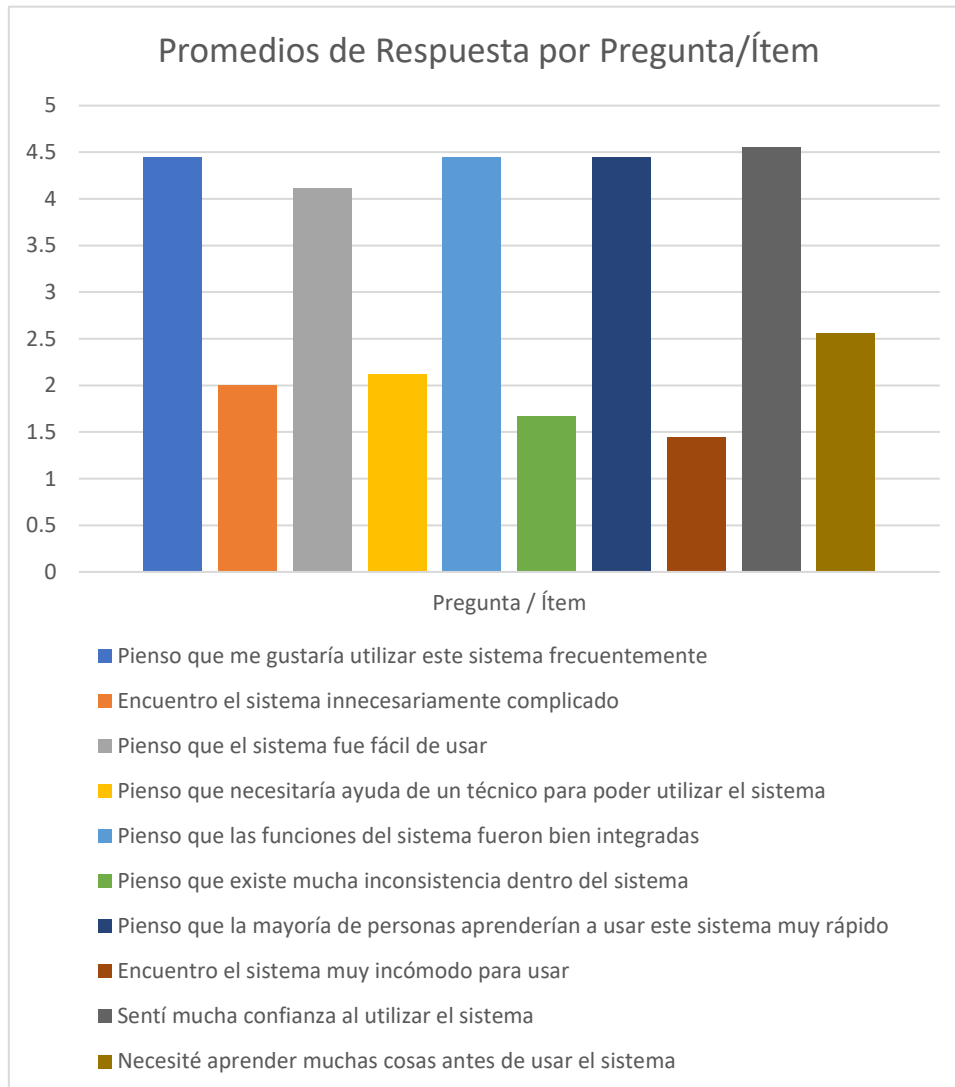


Figura 3.2. Promedios por pregunta/ítem de SUS

Manteniendo la lógica de SUS, se espera que las preguntas o ítem encontrados en posiciones impares cuenten con puntajes considerablemente elevados, al resaltar ideas positivas esperadas dentro de una aplicación; en tanto, aquellos encontrados en posiciones pares esperan contar con puntajes más bien bajos, pues se refieren a posibles defectos, fallos o malas experiencias del usuario durante el uso de la aplicación.

Con lo anterior dicho, volviendo a echar un vistazo en la figura 3.2, podemos ver que, en efecto, aquellas preguntas situadas en las posiciones impares cuentan con valores

considerablemente elevados, donde la pregunta con el menor promedio alcanza aun así un puntaje del 4.11 sobre 5, de ese modo, se puede considerar que, de momento, la aplicación cuenta con buenas características.

Sin embargo, cuando nos centramos en los resultados de las preguntas situadas en posiciones pares, es donde se puede observar un ligero problema, pues estas cuentas con puntajes relativamente elevados, que van desde un 1.44 sobre 5 hasta un valor considerable de 2.55 sobre 5, más de la mitad dentro de la escala.

Con las puntualizaciones previas, y recordando aquellas preguntas con mayor impacto negativo:

- Encuentro el sistema innecesariamente complicado,
- Pienso que necesitaría ayuda de un técnico para poder utilizar el sistema,
- Pienso que existe mucha inconsistencia dentro del sistema,
- Encuentro el sistema muy incómodo para usar, y
- Necesité aprender muchas cosas antes de usar el sistema,

Hacen que sea necesario replantear varios aspectos del funcionamiento y presentación o disposición de las características de la aplicación a los usuarios, de modo que a futuro estos pueden disminuir.

Sin embargo, respecto a estas encuestas y sus resultados, hay que tener en cuenta la complejidad de la aplicación presentada a los usuarios. Si bien, existe un rango considerablemente grande de respuestas por cada pregunta, hay que tener en cuenta que algunos de los encuestados no representan parte de los interesados o clientes objetivos de la aplicación, provocando que algunas respuestas hayan estado condicionadas, algo que no evita el hecho que existan partes de la aplicación que puedan y deban ser modificadas y mejoradas.

4 CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

Definir correctamente el o los flujos DevOps a ser implementados dentro de un proyecto de desarrollo de software depende de varios factores: el tamaño del proyecto, el tipo de proyecto, la plataforma, herramienta y sistema operativo sobre los cuales se ejecute el proyecto, la metodología de desarrollo, la metodología de gestión del proyecto, entre algunos otros. Todos estos detalles, si bien pueden impactar en mayor o menor medida, modifican la forma en cómo los flujos son desarrollados, llevando a que su implementación pueda tomar un mayor tiempo al deseado hasta que sean bien definidos e integrados dentro del proyecto.

Dentro del desarrollo de aplicación móviles, muy probablemente los flujos que quieran ser definidos tendrán como una parte fundamental la compilación del proyecto para la generación automática de un ejecutable, como lo es un APK para dispositivos Android, puesto que a partir de este ejecutable se pueden realizar pruebas en entornos reales, en entornos virtuales automatizados o manuales, o incluso la posterior distribución de la aplicación a través de algún servicio como las propias tiendas de las aplicaciones o algún servicio propio o externo.

Como si se tratase de un desarrollo de una aplicación, la implementación y las pruebas se vuelven fases cruciales de los flujos DevOps que han sido diseñados para dichos proyectos; dado que estos flujos pueden ser descritos como una serie de instrucciones a ser ejecutadas, podemos compararla con la implementación de código con un programa cualquiera, por lo que, con esto en mente, la implementación y pruebas permiten comprobar que estos flujos son ejecutados correctamente, y que los resultados obtenidos son, efectivamente, los resultados esperados al momento del diseño de estos.

La inclusión de flujos DevOps dentro del desarrollo de código no puede limitarse solamente a su ejecución de forma aleatoria o no tan bien organizada. Dado que estos flujos pretenden recoger un resultado más o menos específico, o al menos resultados que sigan una misma estructura, es importante definir en qué momento o estado del desarrollo se ejecutarán los mismos, o bajo qué condiciones serán ejecutadas, esto puede ser: en la integración de las partes de cada desarrollador, en

la culminación de un módulo completo, en el paso de desarrollo a producción, en la finalización de un Sprint o el día de trabajo (en el caso de uso de la metodología Scrum), entre otros; todos ellos, gracias a la libertad de descripción del código a ejecutarse, así como a las configuraciones que pueden describirse como condiciones para su ejecución.

La integración de flujos DevOps dentro de un proyecto de desarrollo trae consigo muchas ventajas en la mayoría de los proyectos o trabajos, incluso, no solo de desarrollo; sin embargo, esto no ocurre con todos los proyectos. Como con la mayoría de las metodologías, estos necesitan de un análisis previo para decidir o no si su inclusión en el proyecto representa una ventaja o una desventaja, así mismo, permite realizar cambios o adaptaciones de estos a lo que el proyecto necesita.

El uso de flujos DevOps en proyectos demasiado pequeños, o que han sido destinados a ser desarrollados por una sola persona, puede que no sea la mejor opción. La inclusión de flujos DevOps bajo estas circunstancias podría implicar el aumento de esfuerzos y costos al proyecto, pues el desarrollador es capaz de ejecutar todo el proceso de desarrollo dentro de su computador, sin necesidad de enfocarse en esfuerzos del diseño de los flujos, su implementación, sus pruebas, o incluso su ejecución. Sin embargo, esto también depende mucho del escenario y del proyecto; así, si dentro del desarrollo una parte está destinada a la ejecución de pruebas automáticas, integración dentro o con otros servicios, es posible que estos flujos, al automatizarse, faciliten esta parte para el desarrollador.

Decidir por el uso de flujos DevOps o no, requiere de un análisis de varios componentes del proyecto que está siendo desarrollado. Tomar en cuenta cada uno de los aspectos, sean: el alcance del proyecto, el tamaño del proyecto, las herramientas o servicios utilizados, el equipo de trabajo u otros, hace que se evite pasar por todo un proceso de desarrollo de flujos DevOps cuando no es necesario, o evitar esfuerzos de los propios desarrolladores en algún proceso por la omisión de esto, según sea el caso.

4.2 Recomendaciones

Para que la ejecución de flujos DevOps centrados en la Integración Continua y/o Despliegue Continuo y sus beneficios puedan ser más fáciles de ver y entender, se esperaría su inclusión dentro de un proyecto de dos o más desarrolladores, de modo

que el flujo de codificación no sea lineal y ejecutado por una sola persona, sino que existan varios flujos de cambios en el producto que está siendo desarrollado y que la integración lógica, estructurada y correcta sea una forma de facilitar el trabajo a todo el equipo.

Del mismo modo, la definición de correctos flujos DevOps es necesaria para un buen desempeño a lo largo del proceso de desarrollo. Estos flujos deben ser configurados antes de introducirse a la fase de desarrollo del producto, de modo que sirvan como una herramienta, si bien dependiente de la plataforma, del lenguaje de programación o las herramientas utilizadas, a su vez independiente del avance o la fase en la que se encuentren los desarrolladores o el estado del producto.

5 REFERENCIAS BIBLIOGRÁFICAS

- [1]“¿Qué es ERP? | Definición de planificación de recursos empresariales | SAP Insights,” SAP, 2019. [Online]. Available: <https://www.sap.com/latinamerica/insights/what-is-erp.html>
- [2]J. Nestell and D. Olson, Successful ERP Systems : A Guide for Businesses and Executives, 1st ed. Business Expert Press, 2018 [Online]. Available: <https://bibliotecas.ups.edu.ec:2708/lib/upsal/reader.action?docID=5165162&query=978-1-63157-846-5>
- [3]K. Schwaber and J. Sutherland, “The Scrum Guide The Definitive Guide to Scrum: The Rules of the Game,” 2020 [Online]. Available: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>
- [4]S. Sharma and B. Coyne, DevOps for fummies, 2nd ed. New Jersey: John Wiley & Sons, Inc, 2015 [Online]. Available: <https://www.immagic.com/eLibrary/ARCHIVES/EBOOKS/W150421S.pdf>
- [5]M. Krief, Learning DevOps : The Complete Guide to Accelerate Collaboration with Jenkins, Kubernetes, Terraform and Azure DevOps, 1st ed. Birmingham: Packt Publishing, 2019 [Online]. Available: <https://online-pmo.com/wp-content/Education/Learning%20DevOps.pdf>
- [6]GitLab, “About GitLab,” GitLab. [Online]. Available: <https://about.GitLab.com/company/>
- [7]GitLab, “GitLab CI/CD | GitLab,” docs.GitLab.com. [Online]. Available: <https://docs.GitLab.com/ee/ci/>
- [8]GitLab, “GitLab Runner | GitLab,” docs.GitLab.com. [Online]. Available: <https://docs.GitLab.com/runner/>
- [9]GitLab, “Repository | GitLab,” docs.GitLab.com. [Online]. Available: <https://docs.GitLab.com/ee/user/project/repository/>
- [10]Git, “Git - About Version Control,” Git-scm.com, 2019. [Online]. Available: <https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control>
- [11]Git, “What is a Git Repository? | Beginner Git Tutorial,” www.gitkraken.com, 2019. [Online]. Available: <https://www.gitkraken.com/learn/git/tutorials/what-is-a-git-repository>
- [12]Git, “Git - Branches in a Nutshell,” Git-scm.com, 2019. [Online]. Available: <https://git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell>
- [13]Git, “Git - Working with Remotes,” git-scm.com, 2019. [Online]. Available: <https://git-scm.com/book/en/v2/Git-Basics-Working-with-Remotes>

- [14]Figma, "About Figma, the collaborative interface design tool.," Figma, 2011. [Online]. Available: <https://www.figma.com/about/>
- [15]Google, "Documentation | Firebase," Firebase, 2019. [Online]. Available: <https://firebase.google.com/docs>
- [16]Google, "Cloud Firestore | Firebase," Firebase, 2019. [Online]. Available: <https://firebase.google.com/docs/firestore>
- [17]Google, "Cloud Firestore Data model," Firebase. [Online]. Available: <https://firebase.google.com/docs/firestore/data-model>
- [18]Google, "Firebase App Distribution," Firebase. [Online]. Available: <https://firebase.google.com/docs/app-distribution>
- [19]Docker, "Docker overview," Docker Documentation, Apr. 09, 2020. [Online]. Available: <https://docs.docker.com/get-started/overview/>
- [20]Google Developers, "Meet Android Studio | Android Developers," Android Developers, 2019. [Online]. Available: <https://developer.android.com/studio/intro>
- [21]DeveloperAndroid, "Kotlin overview," Android Developers. [Online]. Available: <https://developer.android.com/kotlin/overview>
- [22]W3, "Extensible Markup Language (XML)," W3.org, 2016. [Online]. Available: <https://www.w3.org/XML/>
- [23]RedHat, "¿Qué es YAML? YAML y su uso en la automatización," www.redhat.com, 2021. [Online]. Available: <https://www.redhat.com/es/topics/automation/what-is-yaml>
- [24]CircleCI, "Docker Hub," hub.docker.com, 2021. [Online]. Available: <https://hub.docker.com/r/circleci/android/tags>
- [25]J. Brooke, "SUS: A quick and dirty usability scale," Usability Eval. Ind., vol. 189, 1998 [Online]. Available: https://www.researchgate.net/publication/228593520_SUS_A_quick_and_dirty_usability_scale

6 ANEXOS

ANEXO I

En el enlace se encuentra el repositorio con el código fuente de implementación del módulo desarrollado en el proyecto.

<https://GitLab.com/manticore-labs/trabajo/proyectos/trabajo-de-titulacion/cp-armijos/-/tree/main>

ANEXO II

Historia de Usuario # XXX

Versión: X.X.X

Historia de Usuario

Antecedentes: XXX

- **Escenario: XXX**
 - **Cuando XXX**
Entonces XXX
- **Criterios de aceptación**
 - XXX
 - **XXX**
 - XXX
- **Comentarios (opcional)**
 - XXX
- **Mockup referencial (opcional)**
 - XXX
- **Tabla relacional referencial (opcional)**
 - XXX
- **Tabla no relacional referencial (opcional)**
 - XXX

- **Título** (opcional)
 - Valor

ANEXO III

Historia de Usuario # 001

Versión: 0.1.0

Gestión de Cuentas de la Organización

Antecedentes: Yo, como usuario, ingreso al sistema con mis credenciales y me dirijo a la opción de la página principal “Cuentas”

- **Escenario:** Visualizar lista de cuentas de la organización
 - **Cuando** he ingresado a la opción “Cuentas”
Entonces se mostrará una lista con las cuentas de la organización y una lista con las cuentas de terceros existentes
- **Criterios de aceptación**
 - Lista de recuadros de información
 - **Campos primarios**
 - Organización, Identificación, Tipo, Responsable
 - **Campos secundarios**
 - Ninguna
- **Comentarios** (opcional)
 - Ninguno
- **Escenario:** Registrar una cuenta de la organización
 - **Cuando** he ingresado a la opción “Cuentas”
Entonces se mostrará una lista con las cuentas de la organización y una lista con las cuentas de terceros existentes
Y Cuando he seleccionado la opción “Agregar Nueva Cuenta” en la

sección de la lista de con las cuentas de la organización

Entonces se ha mostrado un formulario de creación de cuenta

Y Cuando he llenado correctamente el formulario y seleccionado la opción “Registrar Información”

Entonces se ingresará la información correspondiente en la aplicación

- **Criterios de aceptación**

- Mensaje de Verificación

- Informando el resultado de la operación.

- Lista de recuadros de información

- **Campos requeridos**

- ORGANIZACIÓN – Nombre de la organización
- IDENTIFICACIÓN – Caracteres alfanuméricos y caracteres especiales de 10 caracteres, único por cada Cuenta
- TIPO – Lista de Valores (Activo, Pasivo, ...)
- RESPONSABLE – Caracteres alfabéticos entre 2 y 100 caracteres

- **Comentarios** (opcional)

- Ninguno

- **Escenario: Actualizar una cuenta de la organización**

- **Cuando** he ingresado a la opción “Cuentas de la Organización”

Entonces se mostrará una lista con las cuentas de la organización y una lista con las cuentas de terceros existentes

Y Cuando he seleccionado una cuenta existente de la lista con las cuentas de la organización

Entonces se ha mostrado un formulario de actualización de cuenta

Y Cuando he llenado correctamente el formulario y seleccionado la opción “Registrar Información”

Entonces se ingresará la información correspondiente en la aplicación

- **Criterios de aceptación**

- Mensaje de Verificación

- Informando el resultado de la operación.

- Lista de recuadros de información

- **Campos requeridos**

- ORGANIZACIÓN – Nombre de la organización

- IDENTIFICACIÓN – Caracteres alfanuméricos y caracteres especiales de 10 caracteres, único por cada Cuenta
 - TIPO – Lista de Valores (Activo, Pasivo, ...)
 - RESPONSABLE – Caracteres alfabéticos entre 2 y 100 caracteres
- **Comentarios** (opcional)
 - Ninguno
- **Escenario: Eliminar una cuenta de la organización**
 - **Cuando** he ingresado a la opción “Cuentas de la Organización”
Entonces se mostrará una lista con las cuentas de la organización y una lista con las cuentas de terceros existentes
Y Cuando he seleccionado una cuenta existente de la lista con las cuentas de la organización
Entonces se ha mostrado un formulario de actualización de cuenta
Y Cuando he seleccionado la opción “Eliminar Cuenta”
Entonces se eliminará la información correspondiente en la aplicación
- **Criterios de aceptación**
 - Mensaje de Verificación
 - Informando el resultado de la operación.
- **Comentarios** (opcional)
 - Ninguno

Antecedentes: Yo, como usuario, ingreso al sistema con mis credenciales, selecciono el periodo actual y elijo la opción “Agregar Transacción”.

- **Escenario: Registrar una cuenta de la organización desde el registro de una transacción**
 - **Cuando** he ingresado a la opción “Agregar Transacción”
Entonces se ha mostrado un formulario de registro de transacción
Y Cuando he seleccionado la opción “Créala aquí” en la parte de Información de la Organización
Entonces se ha mostrado un formulario de creación de cuenta
Y Cuando he llenado correctamente el formulario y seleccionado la

opción “Registrar Información”

Entonces se ingresará la información correspondiente en la aplicación

- **Criterios de aceptación**
 - Lista de recuadros de información
 - **Campos primarios**
 - Organización, Identificación, Tipo, Responsable
 - **Campos secundarios**
 - Ninguna
- **Comentarios** (opcional)
 - Ninguno

ANEXO IV

Historia de Usuario # 002

Versión: 0.1.0

Gestión de Cuentas de Terceros

Antecedentes: Yo, como usuario, ingreso al sistema con mis credenciales y me dirijo a la opción de la página principal “Cuentas”.

- **Escenario:** Visualizar lista de cuentas de terceros
 - **Cuando** he ingresado a la opción “Cuentas”

Entonces se mostrará una lista con las cuentas de la organización y una lista con las cuentas de terceros existentes
- **Criterios de aceptación**
 - Lista de recuadros de información
 - **Campos primarios**
 - Organización, Identificación, Tipo, Responsable
 - **Campos secundarios**
 - Ninguna
- **Comentarios** (opcional)
 - Ninguno

- **Escenario: Registrar una cuenta de un tercero**
 - **Cuando** he ingresado a la opción “Cuentas”

Entonces se mostrará una lista con las cuentas de la organización y una lista con las cuentas de terceros existentes

Y Cuando he seleccionado la opción “Agregar Nueva Cuenta” en la sección de la lista de con las cuentas de terceros

Entonces se ha mostrado un formulario de creación de cuenta

Y Cuando he llenado correctamente el formulario y seleccionado la opción “Registrar Información”

Entonces se ingresará la información correspondiente en la aplicación
- **Criterios de aceptación**
 - Mensaje de Verificación
 - Informando el resultado de la operación.
 - Lista de recuadros de información
 - **Campos requeridos**
 - ORGANIZACIÓN – Nombre de la organización
 - IDENTIFICACIÓN – Caracteres alfanuméricos y caracteres especiales de 10 caracteres, único por cada Cuenta.
 - TIPO – Lista de Valores (Activo, Pasivo, ...)
 - RESPONSABLE – Caracteres alfabéticos entre 2 y 100 caracteres.
- **Comentarios (opcional)**
 - Ninguno
- **Escenario: Actualizar una cuenta de un tercero**
 - **Cuando** he ingresado a la opción “Cuentas”

Entonces se mostrará una lista con las cuentas de la organización y una lista con las cuentas de terceros existentes

Y Cuando he seleccionado una cuenta existente de la lista con las cuentas de terceros

Entonces se ha mostrado un formulario de actualización de cuenta

Y Cuando he llenado correctamente el formulario y seleccionado la opción “Registrar Información”

Entonces se ingresará la información correspondiente en la aplicación
- **Criterios de aceptación**

- Mensaje de Verificación
 - Informando el resultado de la operación.
- Lista de recuadros de información
 - **Campos requeridos**
 - ORGANIZACIÓN – Nombre de la organización
 - IDENTIFICACIÓN – Caracteres alfanuméricos y caracteres especiales de 10 caracteres, único por cada Cuenta.
 - TIPO – Lista de Valores (Activo, Pasivo, ...)
 - RESPONSABLE – Caracteres alfabéticos entre 2 y 100 caracteres.
- **Comentarios** (opcional)
 - Ninguno
- **Escenario: Eliminar una cuenta de un tercero**
 - **Cuando** he ingresado a la opción “Cuentas”

Entonces se mostrará una lista con las cuentas de la organización y una lista con las cuentas de terceros existentes

Y Cuando he seleccionado una cuenta existente de la lista con las cuentas de terceros

Entonces se ha mostrado un formulario de actualización de cuenta

Y Cuando he seleccionado la opción “Eliminar Cuenta”

Entonces se eliminará la información correspondiente en la aplicación
- **Criterios de aceptación**
 - Mensaje de Verificación
 - Informando el resultado de la operación.
- **Comentarios** (opcional)
 - Ninguno

Antecedentes: Yo, como usuario, ingreso al sistema con mis credenciales, selecciono el periodo actual y elijo la opción “Agregar Transacción”.

- **Escenario: Registrar una cuenta de un tercero desde el registro de una transacción**
 - **Cuando** he ingresado a la opción “Agregar Transacción”

Entonces se ha mostrado un formulario de registro de transacción
Y Cuando he seleccionado la opción “Créala aquí” en la parte de Información del Cliente

Entonces se ha mostrado un formulario de creación de cuenta
Y Cuando he llenado correctamente el formulario y seleccionado la opción “Registrar Información”

Entonces se ingresará la información correspondiente en la aplicación

- **Criterios de aceptación**
 - Lista de recuadros de información
 - **Campos primarios**
 - Organización, Identificación, Tipo, Responsable
 - **Campos secundarios**
 - Ninguna
- **Comentarios** (opcional)
 - Ninguno

ANEXO V

Historia de Usuario # 003

Versión: 0.1.0

Gestión de Periodos Contables

Antecedentes: Yo, como usuario, ingreso al sistema con mis credenciales y selecciono un periodo contable de la página principal.

- **Escenario:** Visualizar lista de periodos contables
 - **Cuando** he ingresado a la página principal
Entonces se mostrará una lista con los periodos contables registrados
- **Criterios de aceptación**
 - Lista de recuadros de información
 - **Campos primarios**
 - Fecha de inicio del periodo - Fecha de fin del periodo, Año contable del periodo, Valor del DEBE del periodo, Valor del HABER del periodo, Valor del SALDO del periodo

Entonces se ingresará la información correspondiente en la aplicación

- **Criterios de aceptación**
 - Mensaje de Verificación
 - Informando el resultado de la operación.
 - Lista de recuadros de información
 - **Campos requeridos (llenos por defecto)**
 - FECHA DE INICIO DEL PERIODO
 - FECHA DE FIN DEL PERIODO
 - AÑO CONTABLE CORRESPONDIENTE
- **Comentarios** (opcional)
 - Ninguno

ANEXO VI

Historia de Usuario # 004

Versión: 0.1.0

Gestión de Transacciones Contables

Antecedentes: Yo, como usuario, ingreso al sistema con mis credenciales, me dirijo a la opción de la página principal y selecciono el periodo contable actual.

- **Escenario:** Visualizar lista de transacciones contables de un mismo periodo contable
 - **Cuando** he seleccionado al periodo contable actual
Entonces se mostrará un resumen de información del periodo y una lista de las transacciones contables del mismo
- **Criterios de aceptación**
 - Lista de recuadros de información (Lista de transacciones)
 - **Campos primarios**
 - Fecha de registro de la transacción, Cliente, Tipo de Transacción, Grupo Contable, Saldo.
 - **Campos secundarios**
 - Ninguna
 - Resumen de información del periodo contable

- **Cuando** he seleccionado al periodo contable actual
Entonces se mostrará un resumen de información del periodo y una lista de las transacciones contables del mismo
Y Cuando he seleccionado una transacción contable
Entonces se mostrará un resumen de información de la transacción y una lista de los documentos relacionados con la transacción
- **Criterios de aceptación**
 - Resumen de información de la transacción
 - **Campos primarios**
 - Ingresos de la transacción, Egresos de la transacción, Saldo de la transacción, Código de la cuenta de la Empresa sobre la que se ejecutó la transacción, Código de la cuenta del cliente sobre la que se ejecutó la transacción
 - **Campos secundarios**
 - Ninguna
 - Lista de recuadros de información (Lista de documentos)
 - **Campos primarios**
 - Tipo de Documento, Número de documento, Valor total del documento.
 - **Campos secundarios**
 - Ninguna
- **Comentarios** (opcional)
 - Ninguno
- **Escenario: Visualizar documento de una transacción contable**
 - **Cuando** he seleccionado al periodo contable actual
Entonces se mostrará un resumen de información del periodo y una lista de las transacciones contables del mismo
Y Cuando he seleccionado una transacción contable
Entonces se mostrará un resumen de información de la transacción y una lista de los documentos relacionados con la transacción
Y Cuando he seleccionado un documento
Entonces se mostrará un resumen de información del documento
- **Criterios de aceptación**
 - Lista de recuadros de información

- **Campos primarios**
 - Tipo de Documento, Número de Documento, Facturador, Entidad, Método de Pago, Valor.
 - **Campos secundarios**
 - Ninguna
- **Comentarios** (opcional)
 - Ninguno

- **Escenario: Agregar un documento en el registro de una transacción contable**
 - **Cuando** he seleccionado al periodo contable actual
Entonces se mostrará un resumen de información del periodo y una lista de las transacciones contables del mismo
Y Cuando he ingresado a la opción “Agregar Transacción”
Entonces se ha mostrado un formulario de registro de transacción
Y Cuando he seleccionado la opción “Agregar Documento”
Entonces se ha mostrado un formulario de registro de documento
Y Cuando he llenado correctamente el formulario y seleccionado la opción “Registrar Información”
Entonces se ingresará la información correspondiente en la aplicación
- **Criterios de aceptación**
 - Lista de recuadros de información
 - **Campos primarios**
 - TIPO DE DOCUMENTO – Lista de Valores (Nota de Débito, Nota de Crédito, Factura)
 - NÚMERO DE DOCUMENTO – Cadena de caracteres de 15 dígitos.
 - TIPO DE TRANSACCIÓN – Lista de Valores (Ingreso, Egreso)
 - GRUPO CONTABLE – Lista de Valores (Pago a Proveedores, Pago a Empleados, Pago de Viáticos, ...)
 - CUENTA CONTABLE – Lista de Valores (Pago a Proveedores, Pago a Empleados, Pago de Viáticos, ...)
 - VALOR – Número flotante mayor a 0 con precisión de 2 dígitos decimales
 - **Campos secundarios**
 - Ninguna

- **Comentarios** (opcional)
 - Ninguno

- **Escenario: Registrar una transacción contable**
 - **Cuando** he seleccionado al periodo contable actual
Entonces se mostrará un resumen de información del periodo y una lista de las transacciones contables del mismo
Y Cuando he ingresado a la opción “Agregar Transacción”
Entonces se ha mostrado un formulario de registro de transacción
Y Cuando he seleccionado la opción “Agregar Documento”
Entonces se ha mostrado un formulario de registro de documento
Y Cuando he llenado correctamente el formulario y seleccionado la opción “Registrar Información”
Entonces se ingresará la información correspondiente en la aplicación

- **Criterios de aceptación**
 - Lista de recuadros de información
 - **Campos primarios**
 - NÚMERO DE CUENTA DE LA ORGANIZACIÓN – cadena de caracteres numérica o alfanumérica según criterio de búsqueda (Identificador único de cuenta interno o número de cuenta)
 - NÚMERO DE CUENTA DE LA ORGANIZACIÓN – cadena de caracteres numérica o alfanumérica según criterio de búsqueda (Identificador único de cuenta interno o número de cuenta)
 - DOCUMENTO – agregados a través de formularios (ver escenario anterior)
 - **Campos secundarios**
 - Ninguna

- **Comentarios** (opcional)
 - Ninguno

ANEXO VII

Historia de Usuario # 005

Versión: 0.1.0

Generación de Informes Contables

Antecedentes: Yo, como usuario, ingreso al sistema con mis credenciales y me dirijo a la opción de la página principal “Informes Contables”.

- **Escenario: Generar informe contable sin filtro de ningún tipo**
 - **Cuando** he ingresado a la opción “Informes Contables”
Entonces se mostrará un resumen de información contable y una lista con todas las transacciones
- **Criterios de aceptación**
 - Resumen de información del informe
 - **Campos primarios**
 - Debe, Haber, Saldo
 - **Campos secundarios**
 - Ninguna
 - Lista de recuadros de información (Lista de transacciones)
 - **Campos primarios**
 - Fecha de Transacción, Tipo de Transacción, Valor por Transacción
 - **Campos secundarios**
 - Ninguna
- **Comentarios** (opcional)
 - Ninguno
- **Escenario: Generar informe contable filtrado por fecha**
 - **Cuando** he ingresado a la opción “Informes Contables”
Entonces se mostrará un resumen de información contable y una lista con todas las transacciones
Y Cuando seleccione la opción de “Filtrar Resultados”

Entonces se mostrará un formulario para filtrar los resultados del informe contable

Y Cuando configure el filtro de fecha correctamente

Entonces se mostrará la lista de todas las transacciones según el filtro aplicado

- **Criterios de aceptación**

- Resumen de información del informe

- **Campos primarios**

- Debe, Haber, Saldo

- **Campos secundarios**

- Ninguna

- Lista de recuadros de información (Lista de transacciones)

- **Campos primarios**

- Fecha de Transacción, Tipo de Transacción, Valor por Transacción

- **Campos secundarios**

- Ninguna

- Lista de recuadros de información (Filtro de resultados)

- **Campos requeridos**

- FECHA INICIO – selección
- FECHA FIN – selección

- **Comentarios** (opcional)

- Ninguno

- **Escenario: Generar informe contable filtrado por cliente**

- **Cuando** he ingresado a la opción “Informes Contables”

Entonces se mostrará un resumen de información contable y una lista con todas las transacciones

Y Cuando seleccione la opción de “Filtrar Resultados”

Entonces se mostrará un formulario para filtrar los resultados del informe contable

Y Cuando configure el filtro de fecha correctamente

Entonces se mostrará la lista de todas las transacciones según el filtro aplicado

- **Criterios de aceptación**

- Resumen de información del informe

- **Campos primarios**
 - Debe, Haber, Saldo
 - **Campos secundarios**
 - Ninguna
 - Lista de recuadros de información (Lista de transacciones)
 - **Campos primarios**
 - Fecha de Transacción, Tipo de Transacción, Valor por Transacción
 - **Campos secundarios**
 - Ninguna
 - Lista de recuadros de información (Filtro de resultados)
 - **Campos requeridos**
 - CLIENTE – cadena de caracteres numérica o alfanumérica que representa identificación, número de cuenta o nombre del cliente.
- **Comentarios** (opcional)
 - Ninguno
- **Escenario: Generar informe contable filtrado por cuenta contable**
 - **Cuando** he ingresado a la opción “Informes Contables”
Entonces se mostrará un resumen de información contable y una lista con todas las transacciones
Y Cuando seleccione la opción de “Filtrar Resultados”
Entonces se mostrará un formulario para filtrar los resultados del informe contable
Y Cuando configure el filtro de fecha correctamente
Entonces se mostrará la lista de todas las transacciones según el filtro aplicado
- **Criterios de aceptación**
 - Resumen de información del informe
 - **Campos primarios**
 - Debe, Haber, Saldo
 - **Campos secundarios**
 - Ninguna
 - Lista de recuadros de información (Lista de transacciones)
 - **Campos primarios**

- Fecha de Transacción, Tipo de Transacción, Valor por Transacción
 - **Campos secundarios**
 - Ninguna
 - Lista de recuadros de información (Filtro de resultados)
 - **Campos requeridos**
 - CUENTA CONTABLE – cadena numérica o alfanumérica que representa el número o código para la cuenta contable de la empresa.
- **Comentarios** (opcional)
 - Ninguno

ANEXO VIII

En el enlace se encuentra el diagrama de Interfaces realizado a través de la herramienta Figma.

<https://www.figma.com/file/JDrOGUFfI2XWTM8UIFweYC/Interfaces?node-id=0%3A1>

ANEXO IX

En el enlace se encuentra el modelo de Base de Datos realizado a través de la herramienta Figma.

<https://www.figma.com/file/LpK3khzN2BlpNN5N7Vr7TR/Base-de-Datos>

ANEXO X

En el enlace se encuentra un video demostrando la funcionalidad de la pantalla de Inicio de Sesión.

https://epnecuador-my.sharepoint.com/:v:/g/personal/wilman_armijos_epn_edu_ec/EcLI051Xs-1AniiOJTDyuy8BXIqWTL9Ku21Seh5fBdM5xw?e=37BEQA

ANEXO XI

En el enlace se encuentra un video demostrando la funcionalidad de la Página Principal.

https://epnecuador-my.sharepoint.com/:v:/g/personal/wilman_armijos_epn_edu_ec/EWxXpwXtKZdFjeysrrnlqABKkDENMjIKsL0XjHVCmdLMg?e=QqHGPT

ANEXO XII

En el enlace se encuentra un video demostrando la funcionalidad de la pantalla de Cuentas.

https://epnecuador-my.sharepoint.com/:v:/g/personal/wilman_armijos_epn_edu_ec/EYLEcJJSz_NHnIYRg9RsQcgBss1KhiAZcz3Uv1WF7I0WCw?e=ehI7xN

ANEXO XII

En el enlace se encuentra un video demostrando la funcionalidad de la pantalla de Cuentas de la Organización.

https://epnecuador-my.sharepoint.com/:v:/g/personal/wilman_armijos_epn_edu_ec/EW3XON-kRf1Ho2dfDSX65EMB3Mu3862RAsXqlqXryV_uQ?e=ERiPhS

ANEXO XIV

En el enlace se encuentra un video demostrando la funcionalidad de la pantalla de Cuentas de Terceros.

https://epnecuador-my.sharepoint.com/:v:/g/personal/wilman_armijos_epn_edu_ec/Ec1VqiE3uFFPpt8ixG3QzmwBt8BdktVYWrDaYhPsEMkvhA?e=KljcB1

ANEXO XV

En el enlace se encuentra un video demostrando la funcionalidad de la pantalla de Periodo Contable.

https://epnecuador-my.sharepoint.com/:v:/g/personal/wilman_armijos_epn_edu_ec/EZmCEZr-_QhCpZujYosiRLEBBfvdv_m85_OEowjigB7k8Q?e=BJaiW0

ANEXO XVI

En el enlace se encuentra un video demostrando la funcionalidad de la pantalla de Transacción.

https://epnecuador-my.sharepoint.com/:v:/g/personal/wilman_armijos_epn_edu_ec/EcwAGaOcw9ZMoSErNfOani4BzRTKw5Rtrvjc5kV4tuVkgA?e=zF5ZVz

ANEXO XVII

En el enlace se encuentra un video demostrando la funcionalidad de la pantalla de Documento en Transacción.

https://epnecuador-my.sharepoint.com/:v:/g/personal/wilman_armijos_epn_edu_ec/EZfZaqrGLjtMrkTTDV717P4BvD-wsCy4Tlw6A9ghtfUtlw?e=GyO2B1

ANEXO XVIII

En el enlace se encuentra un video demostrando la funcionalidad de la pantalla de Informe Contable.

https://epnecuador-my.sharepoint.com/:v:/g/personal/wilman_armijos_epn_edu_ec/EbPyEjEGdeBMkj9vfGsYLi4BqkBXWyJ_FPjmJORd6kcuag?e=p8lJ6J

ANEXO XIX

En el enlace se encuentra un video demostrando la funcionalidad de la pantalla de Detalle de Transacción.

https://epnecuador-my.sharepoint.com/:v:/g/personal/wilman_armijos_epn_edu_ec/EbujiVhCyctPhplk6ajEJkoBiY1oOq3i0uzTMZ0Zn9h_9A?e=vGzdCF

ANEXO XX

En el enlace se encuentra un video demostrando la funcionalidad de la pantalla de Detalle de Documento.

https://epnecuador-my.sharepoint.com/:v:/g/personal/wilman_armijos_epn_edu_ec/EZ6VZ2LvhNtJv2hPX0wOpSoBapZo5wGVytKzRVK0RWEhxQ?e=6SIYIH

ANEXO XXI

En el enlace se encuentra el documento "Manual de Desarrollador.pdf", el cuál contiene algunas instrucciones para la construcción y uso del producto final.

https://epnecuador-my.sharepoint.com/:f:/g/personal/wilman_armijos_epn_edu_ec/EgWNouoYUYtGvxjua4nh1gBOzNxYK6pqb5lcy727_b8Sg?e=i0Maen

ANEXO XXII

En el enlace se encuentra un video demostrando la funcionalidad completa de la aplicación dentro de un dispositivo real.

https://epnecuador-my.sharepoint.com/:f/g/personal/wilman_armijos_epn_edu_ec/EvQoXbHa1mhGrqr6daWLtrEBvk0Zm-7ZxVYv1mOBHsto8w?e=GJxZRE