

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL
ECUADOR FACULTAD DE INGENIERÍA
SISTEMAS Y COMPUTACIÓN**



**DISERTACIÓN PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERO DE
SISTEMAS DE INFORMACIÓN**

**Aplicativo Informático para la Gestión de Indicadores de Balance Social para
Entidades Cooperativas el Ecuador “AIBASOL”**

Autor: Santiago Xavier Guayasamín Díaz

Director: Ing. Oswaldo Espinoza.

QUITO FEBRERO, 2022

DEDICATORIA

El presente trabajo de titulación está dedicado a mis padres que son el pilar fundamental en este sueño desde que empezó y gracias a su apoyo incondicional he llegado hasta aquí, siendo base de amor y ejemplo a seguir con todas sus enseñanzas en el trayecto de mi vida.

A mi hermano menor para demostrarle que este logro alcanzado es muestra de que los sueños si se cumplen, que, a pesar de las adversidades, a pesar de que el camino se torne oscuro, muy difícil y se quiera abandonar el sueño, con perseverancia, lucha, compromiso y responsabilidad todo se logra.

A María José, mi compañera incansable que me acompaño en la recta final de este recorrido, quién fue y es mi motivación para seguir adelante y no rendirme a las puertas de la meta, mi compañera de las pequeñas victorias y mi fortaleza.

Y a mi familia, por ser la base inmensa de cariño y testigos de mi crecimiento al realizarme profesionalmente.

AGRADECIMIENTO

Agradezco a Dios y a la Virgencita del Quinche por siempre guiarme por el camino del bien e iluminarme cuando todo se pone oscuro, por la salud brindada para poder conseguir paso a paso los objetivos que me estoy trazando y por la salud a toda mi familia y a mis seres queridos que me han acompañado a lo largo de este viaje en busca de mi realización profesional.

Agradezco infinitamente a mi padre por haberme brindado el apoyo necesario y los recursos para seguir mis estudios en tan prestigiosa universidad y por la inmensa ayuda en a lo largo mi carrera y en especial en la recta final.

A mi madre por guiarme y acompañarme desde mis primeros pasos en este sueño, por sus ánimos y soporte cuando estaba a punto de rendirme. No me alcanzaría la vida para agradecer todo lo que han dado por mí.

A mi mis compañeros y amigos de clase, Angie, Andrea y Dennys que me acompañaron a lo largo de la carrera y durante la época de pandemia, fuimos avanzando juntos, nivel a nivel hasta llegar hasta aquí, en especial a Erick quién ha sido la persona que con su ayuda, inteligencia y conocimientos pude continuar.

A María José, quien fue mi apoyo incondicional, mi cómplice en este tramo final de mi carrera quien supo levantarme cuando más lo necesitaba y hacerme centrar para poder alcanzar mis pequeños logros y celebrarlos juntos.

RESUMEN

El presente trabajo de Disertación tiene la intención de formular e introducir una aplicación web para la compañía de Servicios informáticos Webcoopec System, empresa que al ser regulada por la Superintendencia de Economía popular y solidaria necesita implementar una herramienta que genere indicadores de balance social, basados en la resolución Nro. SEPS-INSEPS-2021-0355, publicados por el mencionado organismo de control.

El avance del aplicativo se llevará a cabo a través de la técnica Ágil Scrum, la ejecución de instrumentos Open Source como: Framework Laravel y su arquitectura MVC, Php, Mysql, Apache, Js. Mailtrap que son marcos de trabajo que la compañía utiliza.

Los clientes de la compañía son entidades financieras que trabajan con un CORE financiero el cual gestiona la totalidad de las referencias financiera y contable de cooperativas de ahorro y crédito, el aplicativo debe poder integrarse a cada uno de los CORE para poder extraer los indicadores de balance social, estos deben ser almacenados por meses.

El aplicativo al ser desarrollado mediante un framework de última generación permite heredar las características responsivas, la información para calcular los indicadores se obtiene realizando una integración a la base de datos del CORE financiero Webcoop, al cual se aplican las sentencias Sql, para obtener información de los productos de ahorros, plazo fijo, cartera e información de los representantes de las instituciones.

Finalmente, el aplicativo permite desplegar dashboard que representan de forma gráfica los indicadores para una mejor interpretación gerencial, con ello se habar cumplido con los requerido por el organismo de control evitando que los clientes incurran en multas pecuniarias por incumplimiento de las normas.

ABSTRACT

The present dissertation work intends to formulate and introduce a web application for the Webcoopec System computer services company, a company that, being supervised by the Superintendence of Popular and Solidarity Economy, needs to implement a tool that generates indicators of social balance, based on resolution no. SEPS-INSEPS-2021-0355, published by the aforementioned control body.

The advancement of the application will be carried out through the Agile Scrum technique, the execution of Open Source instruments such as: Laravel Framework and its MVC architecture, Php, Mysql, Apache, Js. Mailtrap which are frameworks that the company uses.

The company's clients are financial entities that work with a financial CORE which manages all the financial and accounting references of savings and credit cooperatives. The application must be able to integrate with each of the COREs in order to extract the balance indicators. social, these must be stored for months.

The application, being developed through a state-of-the-art framework, allows inheriting the responsive characteristics, the information to calculate the indicators is obtained by integrating the Webcoop financial CORE database, to which Sql statements are applied, to obtain information on savings products, fixed term, portfolio and information of the representatives of the institutions.

Finally, the application allows the display of a dashboard that graphically represents the indicators for a better managerial interpretation, thereby complying with those required by the control body, preventing clients from incurring financial fines for non-compliance with

the regulations.

ÍNDICE

DEDICATORIA	2
AGRADECIMIENTO	3
RESUMEN	4
ABSTRACT.....	6
ÍNDICE	8
Índice de Figuras.....	11
Índice de Tablas	11
CAPITULO I	12
1.1 Justificación.....	12
1.2 Planteamiento del Problema.....	13
1.3 Antecedentes	13
1.4 Objetivos.	15
1.4.1 Objetivo General.	15
1.4.2 Objetivos Específicos.	15
CAPITULO II	16
2.1 Metodología Ágil	16
2.2 Gestión de Procesos de Negocio (BPM).	17
2.3 Framework Laravel (MVC)	18
2.3.1 Modelo.....	19
2.3.2 La Vista	19
2.3.3 El controlador	19
2.4 Características de Laravel	19
2.5 JIRA	20

2.6 Aplicación Web.....	21
2.7 Entorno de Desarrollo	21
CAPITULO III.....	23
3.1 Metodología de desarrollo Scrum	23
3.1.1 Transparencia. –	23
3.1.2 Inspección.....	24
3.1.3 Adaptación. -	24
3.2 Definición de Roles.....	24
3.2.1 Product Owner.....	24
3.2.2 Máster Scrum.....	25
3.2.3 Development Team	27
3.3 Eventos de Scrum.....	27
3.3.1 Sprint	28
3.3.2 Scrum Diario (Daily Scrum)	28
3.4 Artefactos de Scrum.....	30
3.4.1 Producto Backlog	30
3.4.2 Sprint Backlog.....	31
3.5 PROCESO SCRUM.....	31
CAPITULO IV.....	33
4.1 DIAGRAMAS DE PROCESOS	33
4.2 EJECUCIÓN DEL PROYECTO	34
4.2.1 Roles	34
4.3 Producto Backlog	35
4.3.1 Planning Product Backlog o lista del producto	35
4.3.2 Backlog Planning (Refinamiento)	38

4.4 DESARROLLO	41
4.4.1 Sprint Seguridades.....	41
4.4.2 Sprint Parámetros	61
4.4.3 Sprint Indicadores.....	62
4.4.4 Dashboard Indicadores	80
4.5 PRUEBAS	97
4.6 IMPLEMENTACIÓN	101
CAPITULO V	102
5.1 Conclusiones	102
5.2 Recomendaciones.....	104
Referencias.....	105

Índice de Figuras

Fig. No 1. <i>Inciso 2 Constitución</i>	13
Fig. No 2. <i>Metodología Ágil</i>	17
Fig No. 3. <i>Ciclo de vida BPM</i>	18
Fig. No 4. <i>Patrón Modelo Vista Controlador</i>	19
Fig. No. 5. <i>Ingreso al sistema</i>	33
Fig. No. 6. <i>Indicadores</i>	34
Fig. No. 7. <i>Modelo ERM-Herramienta BdSchema</i>	41
Fig. No 8. <i>Pestaña de Usuario y Crear Usuario</i>	42
Fig. No. 9. <i>Pantalla de Usuarios</i>	43
Fig. No. 10. <i>Modal de creación de Usuario</i>	43
Fig. No. 11. <i>Validaciones</i>	58
Fig. No. 12. <i>Claves encriptadas</i>	61
Fig. No. 13. <i>Seleccionar Periodo</i>	64
Fig. No. 14. <i>Menú indicadores y opción depósitos</i>	67
Fig. No. 15. <i>Indicador Depósitos por sexo</i>	68
Fig. No 16. <i>Pestaña Menú Dashboards</i>	81
Fig. No 17. <i>Dashboard Depósitos Ahorros</i>	81
Fig. No 18. <i>Dashboard Depósitos Plazos Fijos</i>	82
Fig. No 19. <i>Pruebas Seguridades 1</i>	97
Fig. No 20. <i>Pruebas Seguridades 2</i>	98
Fig. No 21. <i>Pruebas Indicadores 1</i>	99
Fig. No 22. <i>Pruebas Seguridades 2</i>	100

Índice de Tablas

Tabla 1. <i>Procesos fundamentales de Scrum</i>	32
Tabla 2. <i>Roles Scrum</i>	34
Tabla 3. <i>Historias de Usuario</i>	36
Tabla 4. <i>Planning Backlog</i>	38
Tabla 5. <i>Sprint de Seguridades</i>	41
Tabla 6 <i>Sprint Parámetros</i>	61
Tabla 7 <i>Sprint Indicadores</i>	62
Tabla 8 <i>Dashboard Indicadores</i>	80

CAPITULO I

MARCO TEORICO

1.1 Justificación.

La compañía Webcoopec System Cía Ltda, es una organización encargada en la creación de software que cuenta una experiencia superior a los 25 años, especializada en la aplicación para solventar tecnológicas en el ámbito informático, administración de empresas e internet, entre sus principales productos se encuentra los CORE financieros Webcoop – Siryusweb, desarrollado para entidades cooperativas, el cual en el transcurso del tiempo ha embebido en sus estructuras la totalidad de requerimientos establecidos por los diferentes organismos de control.

La compañía es considerada de acuerdo a la Superintendencia de Economía Popular y Solidaria, mediante la resolución Nro. SEPS-INSEPS-2021-0355, mediante la cual la empresa puede ofrecer servicios auxiliares para el sector cooperativo nacional. La Superintendencia Económica de Solidaridad Popular es el organismo técnico que supervisa y controla a las Entidades del Sector Financiero de Solidaridad Popular, así como a la Organización Económica de Solidaridad Popular del Ecuador, dentro de su mandato de promover su perdurabilidad y adecuado funcionamiento encargado de la para protección de sus socios (Super Intendencia de Economía Popular y Solidaria, 2022).

De acuerdo con la resolución Nro. SEPS-IGT-IGS-INFMR-INGINT-2021-0017 del 24 de noviembre de 2021, teniendo en cuenta el artículo 11, inciso 2, de la Constitución Política de la República del Ecuador estipula el principio para la ejecución de los derechos:

Fig. No 1.

Inciso 2 Constitución

“Todos son iguales y gozan de los mismos derechos, obligaciones y oportunidades.
No discriminar a nadie por motivos de raza, lugar de nacimiento, edad, género, sexo,
identidad cultural, estado civil, idioma, religión, ideología, afiliación política, historial
judicial, estatus socioeconómico, estatus migratorio, orientación sexual, salud estado civil,
infección por VIH, discapacidad, diferencia física, ni por cualquier otra distinción individual
o colectiva, temporal o permanente, que tenga por objeto o resultado menoscabar o dejar sin
efecto el reconocimiento, goce o ejercicio de derechos. La ley debe castigar todas las formas
de discriminación”.

Fuente: (Constitución de la República del Ecuador, 2008, Art. 11)

La aplicación de los derechos e intereses de género de los socios, clientes y usuarios financieros en las finanzas inclusivas, se emiten las siguientes normas de control. Con el cual se emite los lineamientos que se deben considerar en el desarrollo de la herramienta informática que permita publicar información solicitada del organismo de control. Dicha resolución se debe implementar en cada uno de los clientes de la compañía Webcoopec System Cía. Ltda.

1.2 Planteamiento del Problema.

El sector Cooperativo del Ecuador no dispone de ninguna herramienta especializada de indicadores de balance social, por ello es necesario desarrollar e implementar una herramienta web, que permita interactuar o integrarse con las bases de datos de los CORE financieros con el propósito de obtener los indicadores requeridos en las notas técnicas de la resolución, las mismas están en una base de datos registradas y que puedan ser que representados gráficamente.

1.3 Antecedentes.

La empresa Webcoopec System Cía Ltda, es una organización de carácter privada cuyo objeto principal es el diseño, planificación y el desarrollo de software informático, actualmente

se encuentra calificada como empresa de servicios de apoyo del sector financiero popular y solidario encargado de llevar a cabo prestaciones en el ámbito de servicios de carácter software financiero y de computación al sector cooperativo nacional, calificación que fue emitida mediante RESOLUCIÓN Nro. SEPS-INSEPS-2021-0355 de la superintendencia de Economía Popular y Solidaria.

La compañía en el transcurso de su permanencia en el mercado ha implementado alrededor de 65 aplicaciones en diferentes entidades del Ecuador, manteniendo alrededor de 25 cooperativas de ahorro y crédito con el CORE financiero que son de pertenencia de la compañía.

La Superintendencia de Economía Popular y Solidaria, organismo de control para las organizaciones cooperativas de ahorro y crédito, desde el año 2012 fecha de su creación ha sostenido el discurso de incluir dentro de sus políticas de regulación la implementación de los principios de balance social en el sector cooperativo, ha pasado mucho tiempo sin que se haya tocado para nada este tema sin embargo el 24 de noviembre del 2021 publico la resolución Nro. SEPS-IGT-IGS-INFMR-INGINT-2021-0017, solicitando a las cooperativas la publicación de los indicadores de balance social aplicados en sus organizaciones.

Según ICORED en su boletín BSC del 31-12-2016 menciona “El Balance Social es considerado como una herramienta capaz de estudiar la administración de responsabilidad social con la comunidad y acatar su misión institucional.”, para ello estas organizaciones solicitan a los proveedores de software desarrollar e implementar aplicativos que permitan cumplir esta disposición, planteando como tope máximo el mes de junio del 2022.

La ley Orgánica de la Economía Popular y Solidaria considera a la responsabilidad social como la obligación de una organización para expandir los requisitos legales o económicos y alcanzar objetivos de largo plazo que generen algún tipo de beneficio a la sociedad. [7].

Los temas de equilibrio social cooperativo deben ser vistos como una herramienta para analizar y respetar los principios de los derechos del socio o cliente, que permiten la continuidad y oportunidad de la inclusión financiera enfocada al género en el tiempo, Fernández (1998) menciona que “el equilibrio social permite la realización de autoevaluaciones que revelan la evolución de las mismas a lo largo del tiempo, facilitan las comparaciones entre cooperativas, evitan la manipulación de la información y brindan información socioeconómica homogénea”.

La compañía Webcoopec System, para seguir cumpliendo con las regulaciones del organismo de control necesita desarrollar una herramienta informática que permita interactuar con el Core financiero Siryusweb V1.0 para ello acepta la propuesta de generar un proyecto por medio de la utilización de una técnica ágil para su desenvolvimiento.

1.4 Objetivos.

1.4.1 Objetivo General.

Desarrollar un aplicativo informático web de última generación, que permita la gestión y emisión de indicadores de balance social para cooperativas de ahorro y crédito.

1.4.2 Objetivos Específicos.

- Integrar el aplicativo informático especializado para la gestión de indicadores de balance social al CORE financiero Siryusweb.
- Utilizar herramientas Open Source que permita cumplir con el desarrollo del aplicativo mediante el uso de una metodología ágil.
- Implementar políticas de seguridad y gestión de indicadores.
- Establecer dashboard de los indicadores.

CAPITULO II

IDENTIFICACIÓN DE CONCEPTOS METODOLOGICOS Y MATERIAL DE DESARROLLO

Para el desarrollo del aplicativo propuesto es necesario utilizar una metodología ágil, la cual permita reducir sustancialmente el tiempo de implementación ya que la resolución del organismo de control establece el tiempo máximo de publicación de la información hasta junio del 2022.

En este capítulo se analizará y evaluarán las ventajas de utilizar herramientas Open Source utilizadas para la creación del proyecto, a fin de acatar las políticas y estándares que mantiene la compañía.

En este capítulo se describirá el análisis que se debe realizar a los procesos del negocio cooperativo en relación a los indicadores de balance social utilizando la disciplina BPM

También se describirán las herramientas que se aplican en la generación del proyecto las cuales son utilizadas por la compañía como política institucional.

2.1 Metodología Ágil

Henry Ford (1913) utilizó en su cadena de fabricación con un enfoque ágil de trabajo para afrontar las restricciones del método en cascada que se derivaban del sistema de cadena de montaje de la época, la cual se considera el origen del enfoque ágil.

Para la década de los años 90 la mayoría de desarrolladores de software se dieron cuenta que los periodos de elaboración y metodologías de apoyo de los modelos en cascada tenían resultados desfavorables y debían esperar años para sacar una distribución de software funcional.

El costo de oportunidad tiempo y recursos hizo que la mayoría de desarrolladores indagara en la busca de alternativas de innovar el desarrollo de proyectos, es así como en el año 2001 nace la metodología ágil, a través de una agrupación de desarrolladores que dedujo

la patente ágil para el desarrollo de software (Fuentes, 2020).

Se describe cuatro características clave que deben tener prioridad sobre todos los demás problemas.

- a) Personas e interacciones antes que procesos e instrumentos
- b) Software de trabajo antes de la documentación completa
- c) Cooperación con los clientes antes de la negociación del contrato
- d) Responder a los cambios antes de ceñirse al plan.

Con la ejecución de estos términos fue posible la recolección de varios resultados de los procesos de desarrollo y las pruebas de enfoque simultáneo y continuo, donde prima la funcionalidad y no los tradicionales diagramas documentales que llevan tiempos exagerados de diseño, permitiendo que varias secuencias de los procesos de desarrollo sucedan al mismo tiempo. (www.redhat.com).

Fig. No 2.

Metodología Ágil



Nota. La Figura muestra la metodología ágil. Tomado de *Metodologías ágiles ¿qué son y para qué sirven?*, por Think (2018).

2.2 Gestión de Procesos de Negocio (BPM).

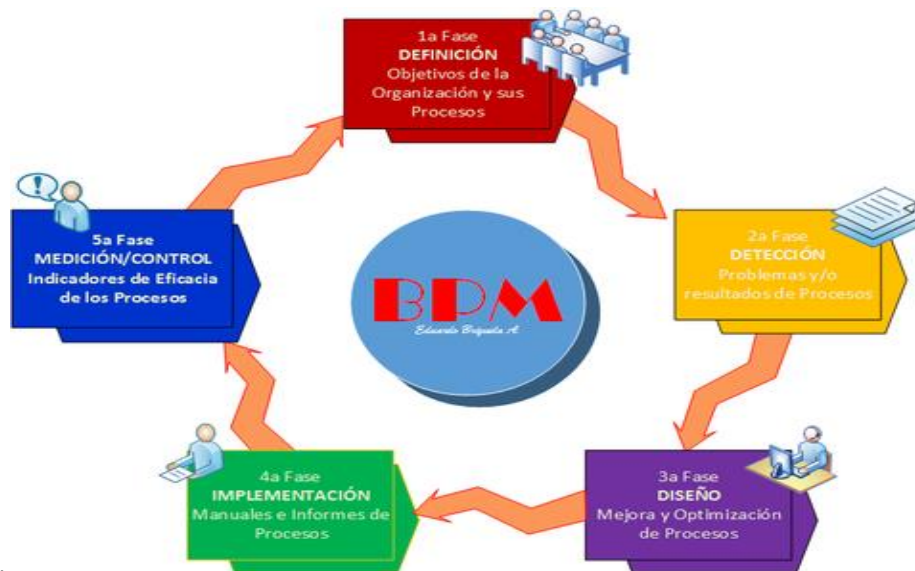
Considerada como una metodología de trabajo es un el encargado de modelar y presentar los procesos del negocio con enfoque en el cliente, sin ser una metodología de software apoya mucho en la implementación efectiva.

Esta metodología aparece en los años 80 en el modelo productivo utilizado por la empresa TOYOTA, la cual con el paso del tiempo y la estandarización de las normativas de calidad se consolidó como un sistema de gestión de procesos.

Jhoan Chavez (2019) menciona “BPM ejecuta, identifica, documenta, evalúa, controlar, diseña y mide los procesos de negocio de una entidad, contempla procesos operativos y automatizados que se orientan mucho al desarrollo del software”.

Fig No. 3.

Ciclo de vida BPM



Nota. La figura muestra el ciclo de vida BPM. Tomado de tecnología BPM, por BPM (2019), Consultora impresa

2.3 Framework Laravel (MVC)

Es un framework que utiliza el sistema de paquetes MVC (Model-View-Controller) para ayudar a desarrollar aplicaciones web escritas en este lenguaje de programación.

Laravel es bastante moderno, brinda a los desarrolladores muchas utilidades para acelerar la generación de aplicaciones web, la herramienta está diseñada para facilitar el mantenimiento y la escalabilidad, por eso es muy utilizada para proyectos ágiles pequeños, grandes y muy grandes que promueven mejores prácticas de desempeño de sistemas.

El patrón MVC utiliza la división en tres partes: el Modelo, la Vista y el controlador. Cada uno de estos recursos se complementan para armar la arquitectura utilizada por Laravel.

2.3.1 Modelo: A través de su funcionalidad para administrar datos de aplicaciones, las aplicaciones le permiten almacenar y administrar información.

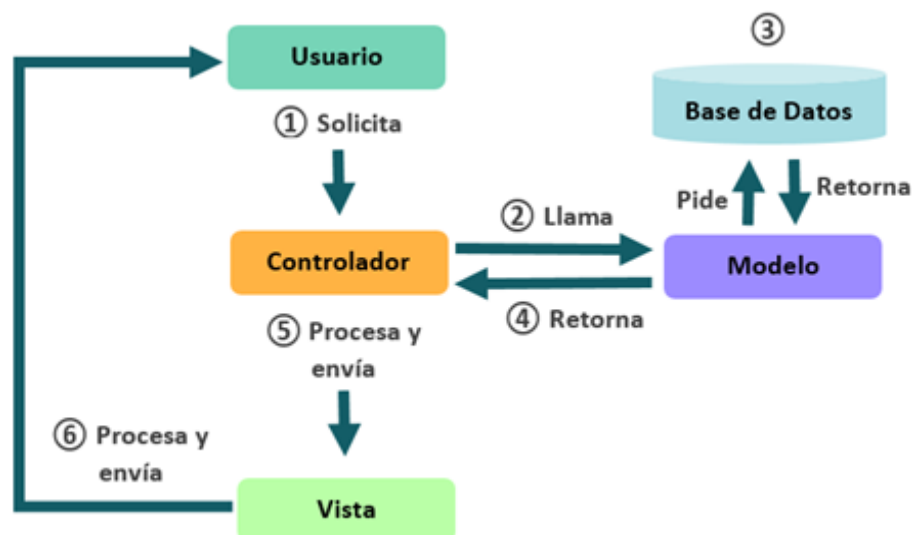
2.3.2 La Vista: Proyecta la situación actual evitando el contacto con el modelo, se considera como la parte “tonta”, su función es la de presentar los datos al usuario.

2.3.3 El controlador: Está pensado como el vínculo entre el modelo y la vista, y es responsable de "notificar" al modelo cuando el usuario manipula la vista.

El patrón MVC es bueno para la escalabilidad de la aplicación, ya que permite que el sistema se ajuste y crezca con el tiempo. Agregar funcionalidad es más fácil cuando tenemos una base sólida y estructurada (Chavez, 2019).

Fig. No 4.

Patrón Modelo Vista Controlador



Nota. La figura muestra el patrón modelo vista controlador. Tomado de ¿Qué es MVC? , por Gómez (2015).

2.4 Características de Laravel

La arquitectura utilizada por Laravel es la de carpetas avanzadas y clases las cuales promueven un adecuado control de código, que guía al equipo de desarrollo en el estilo modelo,

vista controlador (Laravel, 2018).

Entre las principales características podemos mencionar:

- Sistema de enrutamiento a través del cual puede crear y mantener fácilmente varias URL, rutas API, etc. que sean amigables para los usuarios y los motores de búsqueda.
- Sistema de abstracción de base de datos, con un ORM potente pero fácil de usar, a través del cual se pretende transformar la base de datos como objetos simples.
- Creación de colas de trabajo para enviar tareas a ejecutar en segundo plano y mejorar la productividad.
- Configuraciones para el envío de correo electrónico con varios proveedores
- Envío de notificaciones a usuarios a través de correo electrónico, base de datos y otros canales
- Utilización del mismo código para escribir datos en el proveedor de la nube y, por supuesto, en el disco del servidor.
- Gestión de sesiones
- Recordatorios de contraseñas, confirmación de cuentas, recordatorio de usuarios registrados y más.
- Capacidad de acceso en tiempo real y recibir notificaciones cuando cambien los datos en la base de datos.

2.5 JIRA

Herramienta suministrada por la empresa Atlassian, especializada en proporcionar un conjunto de productos diseñados para ayudar en el trabajo de creación de software ágil.

A esta herramienta se la denomina Jira Software considerada una potente herramienta de gestión de todas las etapas de programación, colaboración y publicación de todos los tipos de casos de uso, desde sus requerimientos hasta los casos de pruebas del desarrollo de software, totalmente compatible con la metodología Scrum.

2.6 Aplicación Web

Se definen como herramientas que los usuarios emplean desde un servidor web por medio de Internet o accediendo a una intranet a través de un navegador. Es una aplicación o software que está codificado en un idioma compatible con un navegador web y se confía en que el navegador se ejecute.

Por ello podemos mencionar que se denomina aplicación o sistema Web a toda aquella aplicación de software que puede utilizarse mediante un navegador, accediendo a través de formularios html hospedados en un servidor de publicación web, para nuestro caso de desarrollo utilizaremos el servidor web Apache.

El sistema de comunicación utilizada por estas aplicaciones es a través de los protocolos comunicación HTTP o HTTPS, que suelen ejecutarse sobre TCP/IP, sean de una intranet o de internet.

La arquitectura de los sistemas Web generalmente se encuentra diseñados por los componentes del Backend, que se refiere a los componentes o técnicas con los cuales se procesan los datos, y el Frontend, que son los componentes encargados de mostrar los datos a los usuarios.

2.7 Entorno de Desarrollo

Es la agrupación de procesos e instrumentos que se implementan en el desarrollo de un código fuente o programa. Es un artilugio encargado de desarrollar softwares utilizado en la escritura de códigos, generación, pruebas y depuración de programas. Además proporciona a los desarrolladores una interfaz de usuario común (UI) con el fin de desarrollar y depurar de diferentes maneras (Ortega, 2018).

La aplicación informática para la Gestión de Indicadores de Balance Social para Entidades Cooperativas el Ecuador “AIBASOL” vamos a trabajar dentro del entorno de

desarrollo “Visual Studio Code” que está encargado de la edición de código fuente ligero, y potencia la ejecución en el escritorio y está disponible para Windows, macOS y Linux. Viene con soporte incorporado para JavaScript, TypeScript y Node.js y tiene una amplia variedad de extensiones para otros lenguajes (como C++, C#, Java, Python, PHP, Go) y tiempos de ejecución (como .NET y Unity) que ayudan en gran medida al desarrollo.

CAPITULO III

DESCRIPCIÓN DE LA METODOLOGÍA SCRUM

3.1 Metodología de desarrollo Scrum

Implementado en la generación y la manutención de proyectos o productos complejos, esta metodología permite entregar productos de máximo valor productivo o creativo, permite que las personas puedan enfrentar problemas complejos y adaptativos, a través de la teoría scrum la cual tiene un enfoque teórico de evaluación y seguimiento de procesos empírico (Ortega, 2018).

Este marco de trabajo no es totalmente nuevo debido a que se implementa en la gestión el desarrollo de productos complejos desde principios de los años 90, donde los desarrollos de software demoraban años en la distribución de una aplicativo funcional.

El método Scrum no debe entenderse como un proceso o técnica para la construcción de un proyecto o producto, sino como un marco en el que es posible utilizar una variedad de metodologías y procedimientos, que permitan demostrar la eficacia de las prácticas de administración y desarrollo de productos para su mejora.

Los Equipos Scrum, los cuales se encuentran conformados por roles, eventos, artefactos y reglas asociadas que gobiernen las relaciones e interacciones entre ellos. Cada elemento tiene asignado un propósito determinado y es primordial para la implementación y éxito de Scrum (Holtzhausen & Klerk , 2018).

Se puede decir que Scrum es liviano, sencillo de comprender y considerado un dominio complejo, esta metodología tiene un enfoque teórico empírico de seguimiento de procesos, que asegura que la proveniencia de la información se basa de la experiencia y de la toma de decisiones. Tres pilares sustentan toda la ejecución de la evaluación empírica de procesos: transparencia, inspección y adaptación

3.1.1 Transparencia. – Hace referencia a factores importantes del proceso de desarrollo que en los roles se estipula su visibilidad ya que son causantes del resultado. La transparencia está

definida por estándares frecuentes que los observadores comprendan lo que están observando. Los participantes deben usar un lenguaje común para referirse al proceso, y quienes realizan el trabajo y quienes reciben los resultados del trabajo deben compartir una definición común de "Terminado".

3.1.2 Inspección. - Revisión con frecuencia los artefactos de Scrum y avanzar hacia los objetivos de variación, demora y mejora. Sus inspecciones no deben ser tan frecuentes como para interferir con su trabajo. Resultan beneficiosas cuando se realizan diligentemente en el sitio de trabajo por un inspector calificado.

3.1.3 Adaptación. - Si el inspector establece aspectos del proceso que se desvían de los rangos adecuados el producto final no se acepta, se reajusta los procedimientos o el material. Este ajuste debe hacerse lo más rápido posible para reducir desviaciones significativas.

3.2 Definición de Roles

Para definir los roles debemos entender primero que se necesita un equipo autoorganizado y multifuncional y este debe estar conformado por:

- Product Owner (Dueño de Producto).
- Máster Scrum
- Development Team (Equipo de desarrollo)

A este equipo se lo denomina Equipo Scrum, y según las recomendaciones, puede estar conformado mínimo de 3 miembros y un máximo de 9.

3.2.1 Product Owner

El responsable de generar aumentos de valor del producto y el trabajo del equipo de desarrollo es el dueño, puede variar mucho en las entidades, equipos Scrum e individuos.

La única persona autorizada y responsable de administrar, modificar o eliminar elementos de la cartera de productos es el propietario.

La gestión del Back Log incluye:

- Expresar claramente los elementos de la Lista de historias de usuario;
- Clasifique los elementos en el Product Backlog para lograr mejor los objetivos y misiones;
- Transformar el valor de trabajo a un nivel competitivo;
- Procurar que el Product Backlog sea visible, transparente y claro para las partes interesadas;
- Asegúrese de que el equipo de desarrollo comprenda los elementos del Product Backlog al nivel necesario.

Para que el Product Owner haga su trabajo, toda la organización debe respetar sus decisiones. Las decisiones del Product Owner se reflejan en el contenido y las prioridades del Product Backlog. Nadie debe pedirle al equipo de desarrollo que trabaje de acuerdo con un conjunto diferente de requisitos, y el equipo de desarrollo no debe actuar como dicen los demás. (Guía Scrum Reglas del Juego 2013).

3.2.2 Máster Scrum

Es el facilitador responsable de garantizar que Scrum sea manejado y adoptado por el equipo. El Scrum Master realiza esta tarea y se asegura de que el Scrum Team se adapte a la teoría, la práctica y las reglas de Scrum. Facilita que las personas internas y externas del Equipo Scrum tengan la capacidad de comprensión de las interacciones con el Equipo Scrum y determinar la utilidad de las mismas (Stormi & Laine, 2019).

El Scrum Máster sirve de apoyo para modificar interacciones que potencien el valor creado por el Equipo Scrum.

3.2.2.1 El Servicio del Scrum Máster al Dueño de Producto.- El Scrum Máster da servicio al Dueño de Producto de varias formas como:

- Encuentre técnicas para administrar de manera efectiva el Product Backlog;
- Facilita al equipo de Scrum el entendimiento de la necesidad de los factores claros y

concisos de la Lista de Producto;

- La planificación de productos es comprendida de manera sencilla con un enfoque empírico;
- Asegura de que el propietario del producto conozca cómo ordenar la cartera de productos para maximizar el valor;
- Comprender y practicar la agilidad; sí,
- Promocione los eventos de Scrum según sea necesario o deseado.

3.2.2.2 El Servicio del Scrum Máster al Equipo de Desarrollo. - El Scrum Máster da servicio al equipo de desarrollo de varias formas como:

- Encamina al equipo a ser autoorganizado y multifuncional;
- Apoya la creación de productos de alto valor;
- Suprime restricciones en el progreso;
- Facilitar los eventos de Scrum de acuerdo con los requerimientos; y,
- Dirigir al equipo de desarrollo dentro del ámbito de organizaciones en las que Scrum aún no ha sido adoptado.

3.2.2.3 El Servicio del Scrum Máster a la Organización. - El Scrum Máster da servicio a la organización de varias formas como:

- Guiar la adopción de Scrum por parte de la organización;
- planificar la implementación de Scrum en la entidad;
- Apoyo a los trabajadores y partes interesadas para la comprensión y la creación empírica de productos;
- Inspirar variaciones que influyan en la mejora de la productividad de los equipos Scrum; sí,
- Colaborar con otros Scrum Masters para impulsar la efectividad en la aplicación Scrum en la entidad (Holtzhausen & Klerk , 2018).

3.2.3 Development Team

El equipo de desarrollo consta de personal capacitado que realizan el trabajo al final de cada sprint para entregar una vista previa del producto "terminado", posiblemente en producción. Únicamente los integrantes del equipo participan en la creación de incrementos de características, están organizados y empoderados por las entidades para administrar el trabajo propio. La sinergia resultante optimiza la eficiencia y eficacia del equipo de desarrollo.

El equipo de desarrollo tiene las siguientes características:

Se autoorganizan. Nadie (ni siquiera el Scrum Master) le dijo al equipo de desarrollo cómo convertir el elemento del Product Backlog en un incremento funcional potencialmente implementable;

- Los equipos de desarrollo son multifuncionales y se considera que tienen todas las habilidades necesarias para crear incrementos de productos;
- Scrum no es capaz de reconocer títulos de los integrantes del equipo, son desarrolladores, no hay excepciones a esta regla;
- Scrum tampoco procesa subequipos dentro de un equipo de desarrollo, independientemente de las áreas específicas que se deban considerar, no hay excepciones a esta regla; sí,
- Los integrantes individuales del equipo de desarrollo pueden tener sus habilidades profesionales y entornos máximos de enfoque.

3.3 Eventos de Scrum

Existen situaciones predeterminados en Scrum para fomentar la regularidad minimizando la dependencia de reuniones no definidas en Scrum. Todos los eventos son cajas de tiempo, por lo que todos tienen una duración máxima. La duración de un Sprint es fija y no se puede modificar. Siempre que se alcance el objetivo del evento, otros eventos pueden finalizar, lo que garantiza que se dedique el tiempo adecuado y que no se desperdicien en el

proceso. A excepción del propio Sprint como contenedor del resto de eventos, cada evento Scrum constituye una oportunidad formal para revisar y ajustar algún aspecto. Estas actividades están diseñadas específicamente para lograr una transparencia e inspección significativas. La ausencia de cualquiera de estos eventos da como resultado una menor transparencia y oportunidades perdidas para examinar y adaptar (Stormi & Laine, 2019).

3.3.1 Sprint.- Considerados el núcleo de Scrum, los Sprints son bloques de tiempo (timeboxes) de un mes o menos durante los cuales se crean incrementos de productos "terminados", utilizables e implementables. Es más conveniente si la duración del Sprint es consistente a lo largo del esfuerzo de desarrollo.

Cada nuevo Sprint inicia automáticamente después de que se completa el Sprint anterior.

Un Sprint consta de la reunión de planificación, los Scrums diarios, el trabajo de desarrollo, la revisión y la retrospectiva del mismo. Durante un Sprint:

- No se realizaron cambios que pudieran afectar el Sprint Goal;
- No se bajan los objetivos de calidad; sí,
- A medida que se aprende más, el alcance se puede aclarar y renegociar entre las partes interesadas.
- Cada Sprint se considera como un proyecto de no más de un mes de antigüedad, se implementan para hacer ciertas cosas, define qué construir, el diseño y un plan flexible que guiará la construcción, el trabajo y el producto final.

3.3.2 Scrum Diario (Daily Scrum).- Es una reunión física y el equipo de desarrollo no debe gastar como máximo 15 minutos sincronizando sus actividades y haciendo planes para las próximas 24 horas.

Esto se hace examinando el progreso realizado desde el último Daily Scrum y prediciendo lo que se puede hacer antes del próximo Daily Scrum.

Daily Scrum es ejecutado en un horario determinado continuo y en el mismo lugar todos los días por la razón de que se reduce la dificultad, cada integrante del equipo de desarrollo explica lo que sucedió durante su día y lo hace a través de las siguientes preguntas:

¿Qué hice ayer para ayudar al equipo de desarrollo a lograr el Sprint Goal?

¿Qué voy a hacer hoy para ayudar al equipo de desarrollo a lograr el Sprint Goal?

¿Veo algún obstáculo que nos impida a mí o al equipo de desarrollo alcanzar el Sprint Goal?

El Equipo de Desarrollo utiliza el Daily Scrum con el objetivo de analizar el avance hacia el Sprint Goal y evaluar la tendencia del progreso hacia la finalización del trabajo contenido en el Sprint Backlog.

Daily Scrum es el encargado de potencializar las posibilidades del equipo de desarrollo de lograr el Sprint Goal. Todos los días, el equipo debe comprender la forma en la que se pretende trabajar en conjunto como un equipo autoorganizado para cumplir los Objetivos del Sprint y crear el aumento previsto al final del Sprint.

El equipo con frecuencia se reúne nuevamente inmediatamente después del Daily Scrum para discusiones detalladas, o para ajustar o reprogramar el resto del Sprint. El Scrum Master se asegura de que el Equipo de Desarrollo se reúna, pero el Equipo de Desarrollo es responsable de ejecutar el Daily Scrum. El Scrum Master guía al equipo de desarrollo para mantener el Scrum diario dentro del límite de bloque de 15 minutos.

El Scrum Master se asegura de que solo los miembros del equipo de desarrollo participen en el Scrum diario. Daily Scrums mejora la comunicación, elimina la necesidad de reuniones adicionales, identifica y elimina los obstáculos relacionados con el desarrollo, resalta y facilita la toma de decisiones rápida y aumenta el nivel de conocimiento del equipo de desarrollo. El Daily Scrum es una importante reunión de revisión y adaptación. (Guía Scrum Reglas del Juego 2013).

3.4 Artefactos de Scrum

Representan trabajo o valor de varias maneras que ayudan a brindar transparencia y oportunidades para la inspección y el ajuste. Los artefactos definidos por Scrum están diseñados específicamente para maximizar la transparencia de la información crítica, lo cual es necesario para garantizar que todos tengan la misma comprensión del artefacto. (Guía Scrum Reglas del Juego 2013).

3.4.1 Producto Backlog

Una cartera de productos es una lista ordenada de todo lo que podría ser necesario en un producto, mediante el uso de una herramienta específica de cartera de pedidos, estas listas también se denominan historias de usuario y es la única fuente de requisitos para cualquier cambio en el producto. El propietario del producto es responsable de la acumulación de productos, incluido su contenido, disponibilidad y pedidos.

El backlog nunca está completo, su primer desarrollo refleja los requisitos más conocidos y entendidos, y la lista o backlog evoluciona a medida que evoluciona el producto y el entorno en el que se utiliza. La lista de productos es dinámica; cambia continuamente para determinar qué necesita el producto para que sea apropiado, competitivo y útil. Mientras exista un producto, también existe su lista de productos.

Product Backlog enlista todas las características, funciones, requisitos, mejoras y correcciones que constituyen cambios en el producto para versiones futuras. Los elementos del Product Backlog tienen Descripción, Pedido, Estimación y Valor como atributos. A medida que aumenta el uso y el valor del producto, así como los comentarios del mercado, la lista de pedidos del producto se convierte en una lista más larga y completa. Los requisitos nunca dejan de cambiar, por lo que el Product Backlog es un artefacto vivo. Los cambios en las necesidades comerciales, las condiciones del mercado o la tecnología pueden dar lugar a cambios en la cartera de productos.

3.4.2 Sprint Backlog

El Sprint Backlog se denomina Sprint Backlog y es un agrupación de factores o historias de usuario de la lista de productos aún no completados seleccionados para el Sprint, y un cronograma de entrega de los elementos que deben incorporarse a la lista. Un producto cuyo fin es conseguir el Sprint Goal.

El Sprint Backlog es la predicción del equipo de desarrollo de qué funcionalidad se necesita en el próximo backlog, que se requiere para entregar esa funcionalidad en un estado "terminado". El Sprint Backlog es un plan con suficiente detalle para comprender los cambios que se están realizando en el Daily Scrum.

El Equipo de Desarrollo modifica el Sprint Backlog durante el Sprint, y este Sprint Backlog aparece a lo largo del Sprint. Esto sucede cuando el Equipo de Desarrollo hace un plan y aprende más sobre el trabajo requerido para lograr el Sprint Goal. A medida que se requiere un nuevo trabajo, el equipo de desarrollo lo agrega al Sprint Backlog. Las estimaciones de trabajo restantes se actualizan a medida que se ejecutan o completan los trabajos. Cuando algún elemento del plan se considera innecesario, se elimina. Solo el Equipo de Desarrollo está autorizado para realizar cambios en su Sprint Backlog. El Sprint Backlog es una imagen visual en tiempo real de lo que el Equipo de Desarrollo planea lograr durante el Sprint y pertenece completamente al Equipo de Desarrollo (Pries-Heje & Baskerville, 2017).

3.5 PROCESO SCRUM

El método Scrum también puede verse como un proceso que permite realizar una serie de tareas de manera regular con el objetivo de trabajar colectivamente y promover el trabajo en equipo.

Un proceso Scrum implica actividades y procesos específicos de un proyecto Scrum. En cinco fases, se reúnen 19 procesos básicos de Scrum que son empleados en todos los proyectos (Holtzhausen & Klerk , 2018).

Tabla 1*Procesos fundamentales de Scrum*

FASE	PROCESO SCRUM
INICIO	1.- Generar visión del proyecto
	2.- Determinar los roles de Scrum Master y Steakholder
	3.- Creación un Equipo Scrum
	4.- Desarrollo épico
	5.- Producir un backlog de producto priorizado
	6.- Ejecutar el plan de lanzamiento
	7.- Generar historias de usuario.
PLANIFICACIÓN Y	8.- Estimación de Historias de Usuario
ESTIMACIÓN	9.- Participa en historias de usuario
	10.- Determinar la tarea
	11.- Tareas de estimación
	12.- Crear Sprint Backlog
	13.- Crear entregables
IMPLEMENTACIÓN	14.- Desarrollar Daily Standup
	15.- Cartera prioritaria de productos refinados
REVISIÓN Y	16.- Sprint de Demostración y Validación
RETROSPECTIVA	17.- Revisión de Sprint
	18.- Enviar entregables
LANZAMIENTO	19.- Revisión del Proyecto

Fuente: Recuperado de “Una guía para el Cuerpo de Conocimiento de Scrum (Guía SBOK™) – 3ra Edición”, de Satpathy, T., 2017, p. 16, Avondale, Arizona 85392 USA: SCRUMstudy. VMdu

CAPITULO IV

DESARROLLO E IMPLEMENTACIÓN DEL APLICACIÓN WEB UTILIZANDO LA METODOLOGÍA SCRUM

Se detalla las fases de desarrollo del aplicativo web, enfocada en los procesos críticos de los requerimientos establecidos por la normativa y con la descripción en lenguaje natural de los usuarios que manejaran el aplicativo.

Al considerarse un aplicativo satélite de un CORE financiero es indispensable considerar un nivel de seguridades básicas y los procedimientos necesarios para lograr la integración con la base de datos de la institución financiera, con el propósito de obtener los indicadores requeridos por los organismos de control.

Las instituciones financieras desarrollan sus proyectos basados en la gestión por procesos, motivo por el cual realizaremos los diagramas generales del proceso que se pretende automatizar con el aplicativo web.

4.1 DIAGRAMAS DE PROCESOS

Fig. No. 5.
Ingreso al sistema

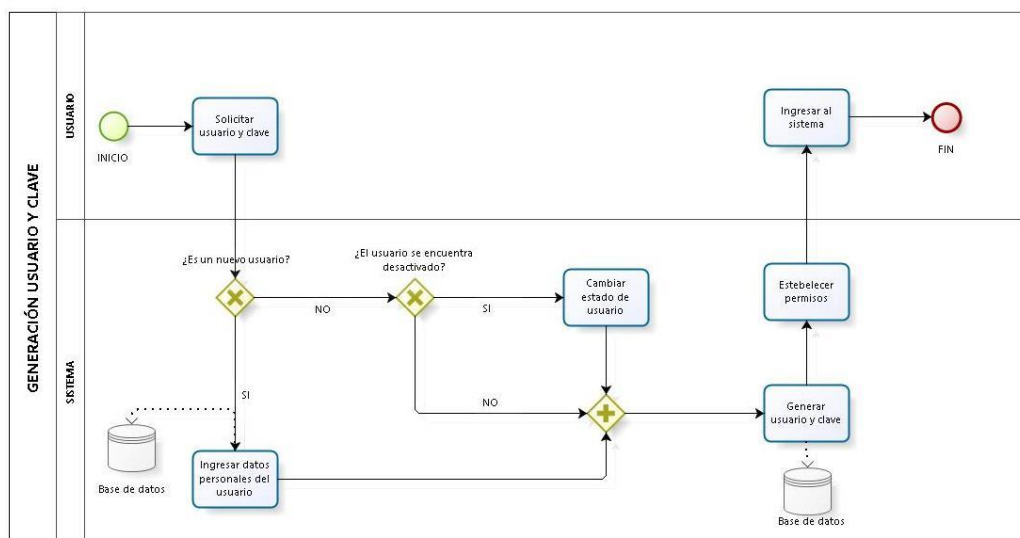
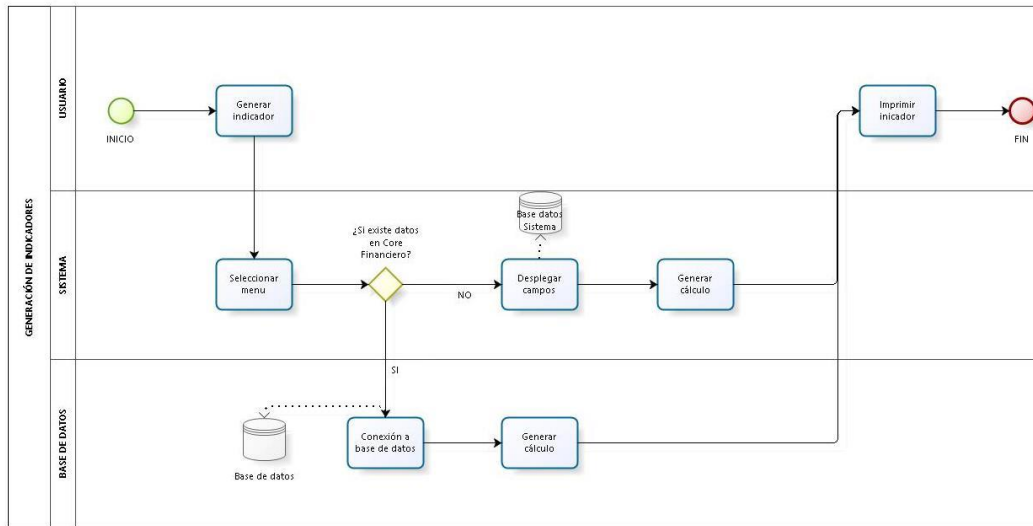


Fig. No. 6.
Indicadores



4.2 EJECUCIÓN DEL PROYECTO

La compañía Webcoopec System Cía Ltda, es un empresa de desarrollo de software la cual entre sus políticas institucionales determinan que todos los proyectos deben ser desarrollados utilizando la metodología Scrum.

4.2.1 Roles

Para el uso de esta metodología debemos empezar con la definición de los roles Scrum:

Tabla 2

Roles Scrum

ROL	DESCRIPCIÓN	RESPONSABLE
PRODUCT OWNER	Dueño de producto es quien	Adriana Betancort
	negocia requerimientos y	
	decisiones sobre el	
	producto con los usuarios	

	Líder que incentiva y promueve el uso de la metodología Scrum, encargado de realizar los deaily Scrum	Elvis Rodríguez
SCRUM MASTER		
	Responsable del diseño y desarrollo de aplicaciones web	Elvis Rodríguez Santiago Guayasamín
DEVELOPMENT TEAM		

La definición de cargos en base a la estructura organizacional de la compañía, donde la jefe de Operaciones Adriana Betancourt es la experta en la normativa de las Superintendencia de Economía Popular y Solidaria, y es considerada Product Owner de la organización, y es la encargada de presenta el diseño y modelo de solución que se ofrece a las entidades que son los clientes de la compañía, el Scrum Master Elvis Rodríguez fue designado por la administración como responsable de apoyar y de dirigir la utilización de la metodología Scrum, el equipo de desarrollo se encuentra conformado por Elvis Rodríguez y Santiago Guayasamín, encargado del diseño, la programación y las pruebas del aplicativo web.

4.3 Producto Backlog

4.3.1 Planning Product Backlog o lista del producto

Es un listado organizado de todos los requerimientos fundamentales para la creación del producto, la misma es proporcionada por el Product Owner, y que es tratada y depurada por el equipo de desarrollo.

A esta lista de requerimientos también se le conoce como historias de usuario las cuales constituyen pequeñas descripciones en lenguaje natural de lo que quiere el usuario, Al escribir

historias de usuarios, se deben considerar los siguientes formatos:

Como <quién> **Quiero** <qué> **Para** <objetivo>

Ejemplo: **Como** Usuario **Quiero** obtener del Core financiero la información **Para** obtener el indicador de depósitos de hombres y mujeres.

El backlog o lista de requerimientos debe ser priorizados estimados y asignados para que el equipo scrum proceda a desarrollar la funcionalidad de cada una de ellas, estas listas evolucionan a medida que el producto crece en sus funcionalidades.

Tabla 3

Historias de Usuario

HISTORIAS DE USUARIO			
ID	COMO...	QUIERO...	PARA...
BAL1	Cooperativa	Acceder a la BD del CORE	Obtener información
BAL2	Cooperativa	Seleccionar un periodo mensual	Obtener información de este periodo
BAL3	Cooperativa	Almacenar información de un periodo	Recuperar indicadores históricos
BAL4	Cooperativa	Almacenar información de autoridades en parámetros	Recuperar la información de los parámetros
BAL5	Cooperativa	Cumplir con la normativa	Publicando indicadores
BAL6	Cooperativa	Gráficos de los indicadores	Publicar los indicadores
BAL7	Usuario	Iniciar sesión	Acceder al sistema como usuario
BAL8	Usuario	Las claves encriptados	Mantener seguridades requeridas por el organismos de control
BAL9	Usuario	Recuperar mi clave	Ingresar al sistema
BAL10	Usuario	Ingresar al sistema desde cualquier localización	Acceder al CORE financiero
BAL11	Usuario	Acceder por el móvil	Acceder a los indicadores
BAL12	Usuario	No duplicar información de los directivos	Almacenar en parámetros
BAL13	Usuario	Seleccionar reportes mensuales	Publicar indicadores
BAL14	Usuario	Seleccionar información de depósitos por sexo	Generar indicador
BAL15	Usuario	Seleccionar información de depósitos por actividad económica	Generar indicador

BAL16	Usuario	Seleccionar información de depósitos por instrucción	Generar indicador
BAL17	Usuario	Seleccionar información de depósitos por grupo étnico	Generar indicador
BAL18	Usuario	Seleccionar información de depósitos por tipo de canal	Generar indicador
BAL19	Usuario	Seleccionar información de productos de tipo de crédito y por sexo	Generar indicador
BAL20	Usuario	Seleccionar información de cartera por tipo de crédito y actividad económica	Generar indicador
BAL21	Usuario	Seleccionar información de cartera por tipo de crédito y por nivel de instrucción	Generar indicador
BAL22	Usuario	Seleccionar información de cartera por tipo de cartera y actividad económica	Generar indicador
BAL23	Usuario	Seleccionar información de cartera por tipo de crédito y por provincia	Generar indicador
BAL24	Usuario	Seleccionar morosidad por sexo	Generar indicador
BAL25	Usuario	Seleccionar morosidad por sexo y actividad económica	Generar indicador
BAL26	Usuario	Seleccionar morosidad por sexo y por instrucción	Generar indicador
BAL27	Usuario	Seleccionar morosidad por sexo y por rango de edades	Generar indicador
BAL28	Usuario	Seleccionar socios por sexo de representantes	Generar indicador
BAL29	Usuario	Seleccionar socios por sexo, rango de edades de representantes	Generar indicador
BAL30	Usuario	Seleccionar información de Educación financiera por sexo	Generar indicador
BAL31	Usuario	Seleccionar información de Educación financiera por sexo y por edades	Generar indicador
BAL32	Administrador	Crear usuario	Almacenar usuarios del aplicativo
BAL33	Administrador	Asignar password	Acceder al sistema web
BAL34	Administrador	Parametrizar periodos	Seleccionar periodos mensuales
BAL35	Administrador	Realizar conexión a Bd del aplicativo	Acceder a la información de la Cooperativa
BAL36	Administrador	Ingresar parámetros	Ingresar información de representantes
BAL37	Administrador	Parametrizar dashboard	Imprimir indicadores

BAL38	Adminstrador	Menú dinámico de opciones	gráficos. Seleccionar, seguridades, indicadores y dashboard
BAL39	Adminstrador	Parámetros de representantes	Almacenar información
BAL40	Usuario	Ingresar información del consejo de administración	Almacenar información del Consejo de Administración
BAL41	Usuario	Ingresar información del consejo de Vigilancia	Almacenar información del Consejo de Vigilancia
BAL42	Usuario	Ingresar información de educación financiera	Almacenar información de la participación de la educación financiera

4.3.2 Backlog Planning (Refinamiento)

El refinamiento o la planificación del Backlog, nos permite ordenar y priorizar la lista con la participación del equipo Scrum, quienes depuran los requerimientos según el funcionamiento del producto, el equipo de desarrollo incrementa o elimina historias repetidas o no necesarias en el proceso de planificación, para con ello diseñar el modelo entidad relación que se necesita para el almacenamiento de la información.

Tabla 4

Planning Backlog

Planning Backlog				
ID	PRIORIDAD	COMO...	QUIERO...	PARA...
BAL1	12	Cooperativa	Acceder a la BD del CORE	Obtener información
BAL2	21	Cooperativa	Seleccionar un periodo mensual	Obtener información de este periodo
BAL3	13	Cooperativa	Almacenar información de un periodo	Recuperar indicadores históricos
BAL4	14	Cooperativa	Almacenar información de autoridades en parámetros	Recuperar la información de los parámetros
BAL7	5	Usuario	Iniciar sesión	Acceder al sistema como usuario
BAL8	3	Usuario	Las claves encriptados	Mantener seguridades requeridas por el organismos de control

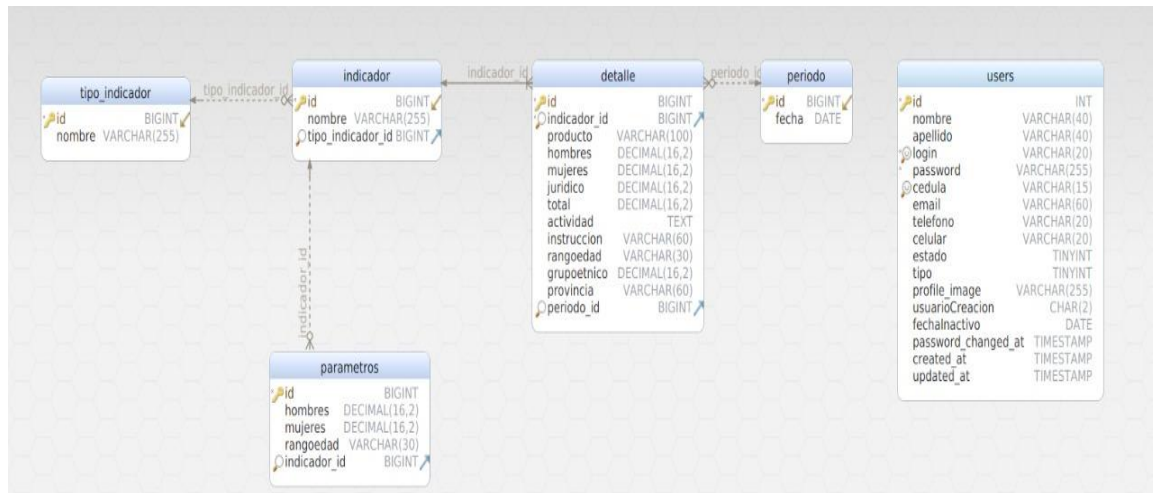
BAL9	4	Usuario	Recuperar mi clave	Ingresar al sistema
BAL10	6	Usuario	Ingresar al sistema desde cualquier localización	Acceder al CORE financiero
BAL11	7	Usuario	Acceder por el móvil	Acceder a los indicadores
BAL12	15	Usuario	No duplicar información de los directivos	Almacenar en parámetros
BAL14	22	Usuario	Seleccionar información de depósitos por sexo	Generar indicador
BAL15	23	Usuario	Seleccionar información de depósitos por actividad económica	Generar indicador
BAL16	24	Usuario	Seleccionar información de depósitos por instrucción	Generar indicador
BAL17	25	Usuario	Seleccionar información de depósitos por grupo étnico	Generar indicador
BAL18	26	Usuario	Seleccionar información de depósitos por tipo de canal	Generar indicador
BAL19	27	Usuario	Seleccionar información de productos de tipo de crédito y por sexo	Generar indicador
BAL20	28	Usuario	Seleccionar información de cartera por tipo de crédito y actividad económica	Generar indicador
BAL21	29	Usuario	Seleccionar información de cartera por tipo de crédito y por nivel de instrucción	Generar indicador
BAL22	30	Usuario	Seleccionar información de cartera por tipo de cartera y actividad económica	Generar indicador
BAL23	31	Usuario	Seleccionar información de cartera por tipo de crédito y por provincia	Generar indicador
BAL24	32	Usuario	Seleccionar morosidad por sexo	Generar indicador

BAL25	33	Usuario	Seleccionar morosidad por sexo y actividad económica	Generar indicador
BAL26	34	Usuario	Seleccionar morosidad por sexo y por instrucción	Generar indicador
BAL27	35	Usuario	Seleccionar morosidad por sexo y por rango de edades	Generar indicador
BAL28	36	Usuario	Seleccionar socios por sexo de representantes	Generar indicador
BAL29	37	Usuario	Seleccionar socios por sexo, rango de edades de representantes	Generar indicador
BAL30	38	Usuario	Seleccionar información de Educación financiera por sexo	Generar indicador
BAL31	39	Usuario	Seleccionar información de Educación financiera por sexo y por edades	Generar indicador
BAL32	1	Administrador	Crear usuario	Almacenar usuarios del aplicativo
BAL33	2	Administrador	Asignar password	Acceder al sistema web
BAL34	9	Administrador	Parametrizar periodos	Seleccionar periodos mensuales
BAL35	10	Administrador	Realizar conexión a Bd del aplicativo	Acceder a la información de la Cooperativa
BAL38	16	Administrador	Menú dinámico de opciones	Seleccionar, seguridades, indicadores y dashboard
BAL39	17	Administrador	Parámetros de representantes	Almacenar información
BAL40	18	Usuario	Ingresar información del consejo de administración	Almacenar información del Consejo de Administración
BAL41	19	Usuario	Ingresar información del consejo de Vigilancia	Almacenar información del Consejo de Vigilancia
BAL42	20	Usuario	Ingresar información de educación financiera	Almacenar información de la participación de la educación financiera

4.3.3 Diagrama ERM

Fig. No. 7.

Modelo ERM-Herramienta BdSchema



4.4 DESARROLLO

4.4.1 Sprint Seguridades

Tabla 5

Sprint de Seguridades

HISTORIAS DE USUARIO				
ID	PRIORIDAD	COMO...	QUIERO...	PARA...
BAL32	1	Administrador	Crear usuario	Almacenar usuarios del aplicativo
BAL33	2	Administrador	Asignar password	Acceder al sistema web
BAL8	3	Usuario	Las claves encriptados	Mantener seguridades requeridas por el organismos de control
BAL9	4	Usuario	Recuperar mi clave	Ingresar al sistema
BAL7	5	Usuario	Iniciar sesión	Acceder al sistema como usuario
BAL10	6	Usuario	Ingresar al sistema desde cualquier	Acceder al CORE financiero

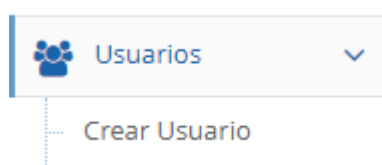
BAL11	7	Usuario	localización Acceder por el móvil	Acceder a los indicadores
BAL12	8	Usuario	No duplicar información de los directivos	Almacenar en parámetros
BAL34	9	Administrador	Parametrizar periodos	Seleccionar periodos mensuales
BAL35	10	Administrador	Realizar conexión a Bd del aplicativo	Acceder a la información de la Cooperativa

Incidencia BAL32 Crear Usuario

Se crea un menú con una opción de crear usuario.

Fig. No 8.

Pestaña de Usuario y Crear Usuario.



Código.

```
<li class="">

    <a href="#" class="dropdown-toggle">

        <i class="menu-icon fa fa-users"></i>

        <span class="menu-text"> Usuarios </span>

        <b class="arrow fa fa-angle-down"></b>

    </a>

    <b class="arrow"></b>

    <ul class="submenu">

        <li class="">
```

```

<a href="{{ route ('crearUsuario') }}">

    <i class="menu-icon fa fa-user"></i>

    Crear Usuario

</a>

<b class="arrow"></b>

</li>

```

Fig. No. 9.

Pantalla de Usuarios

Usuarios					
					Nuevo Usuario
Nombre	Apellido	Cédula	Correo Electrónico	Estado	Acciones
Santiago	Guayasamin	1721859419	santhy15c@gmail.com	ACTIVO	✎
Santiago	Guayasamin	1765656543	hola@gmail.com	ACTIVO	✎
Santiago	Guayasamin	1265656528	hola111@gmail.com	ACTIVO	✎
Santiago	Guayasamin	1212121222	santhy111c@gmail.com	INACTIVO	✎
Juan	Guayasamin	1721859411	santhy15d@gmail.com	ACTIVO	✎

En esta pantalla se muestran los usuarios creados y en la parte superior derecha se muestra el botón de “Nevo usuario” el cual nos desplegará un modal para la creación de un nuevo usuario.

Fig. No. 10.

Modal de creación de Usuario

Nuevo Usuario

✕

Nombre *

Apellido *

Cédula *

Correo Electrónico *

Contraseña *

Confirmar Contraseña *

Guardar

Se creó un modal con los campos:

- Nombre
- Apellido
- Cédula
- Correo Electrónico
- Contraseña
- Confirmar Contraseña

En donde se llenarán los campos y se guardarán al dar clic al botón “Guardar”

Código

Controlador

Para la realización del indicador de seguridades se creo un controlador con el nombre de “seguridadesController”

```
<?php
```

```
namespace App\Http\Controllers;

use App\Models\User;

use Illuminate\Http\Request;

use App\Http\Requests\dashboard\crearUsuariosRequest;

// use Illuminate\Support\Facades\Hash;

use Illuminate\Support\Facades\Crypt;

class seguridadesController extends Controller
{

    public function index (Request $request)
```

```

{
    $informacion = User::paginate(7);

    return view('Seguridades.formulario',compact ('informacion'));
}

public function guardar(crearUsuariosRequest $request)
{
    $usuario = new User;

    $usuario ->nombre = $request ->nombre;

    $usuario ->apellido = $request ->apellido;

    $usuario ->cedula = $request ->cedula;

    $usuario ->email = $request ->email;

    $usuario ->password = Crypt::encrypt($request ->password);

    $usuario ->estado = 1;

    $usuario->save();

    $request->session()->flash('success', '¡Se creó correctamente el usuario!');
}
}

```

Modelo

De igual manera se creó un modelo llamado “User” con los campos que se encuentran en la base de datos.

```

<?php

namespace App\Models;

use Illuminate\Contracts\Auth\MustVerifyEmail;

use Illuminate\Database\Eloquent\Factories\HasFactory;

use Illuminate\Foundation\Auth\User as Authenticatable;

```

```
use Illuminate\Notifications\Notifiable;
```

```
use Laravel\Sanctum\HasApiTokens;
```

```
class User extends Authenticatable implements MustVerifyEmail //Implementar funcion de
verificacion de correo
```

```
{
```

```
    use HasApiTokens, HasFactory, Notifiable;
```

```
    /**
```

```
     * The attributes that are mass assignable.
```

```
     *
```

```
     * @var array<int, string>
```

```
     */
```

```
    protected $fillable = [
```

```
        'nombre',
```

```
        'apellido',
```

```
        'cedula',
```

```
        'login',
```

```
        'email',
```

```
        'password',
```

```
        'estado'
```

```
    ];
```

```
    /**
```

```
     * The attributes that should be hidden for serialization.
```

```
     *
```

```
     * @var array<int, string>
```

```

    */

protected $hidden = [

    'password',

    'remember_token',

]

/**

 * The attributes that should be cast.

 *

 * @var array<string, string>

 */

protected $casts = [

    'email_verified_at' => 'datetime',

];

}

```

Vista

Para mostrar la pantalla de “Usuarios” y el modal de “Nuevo Usuario” se creó una vista con el nombre de: “formulario.blade.php”

```

@extends('interfaz.dashboard.plantillaGeneral')

@section('content')

@if (session('status'))

{{session('status')}}

<div class="alert alert-success" role="alert"></div>

@endif

```

```
<div class="page-header"><h1>Usuarios</h1></div><!-- /.page-header -->
```

```
<!-- qui va el contenido de la tabla -->
```

```
<div class="row">
```

```
  <div class="col-xs-12">
```

```
    <div class="col-xs-12"><div>&nbsp;</div></div>
```

```
    <div class="col-xs-12">
```

```
      <div class="col-sm-6">
```

```
    </div>
```

```
      <div class="col-sm-6">
```

```
        <a href="#nuevoUsuario" role="button" class="btn btn-primary pull-right" data-  
toggle="modal">
```

```
          &nbsp;  Nuevo Usuario &nbsp; 
```

```
        </a>
```

```
    </div>
```

```
</div>
```

```
<div class="col-xs-12"><div>&nbsp;</div></div>
```

```
<div class="col-xs-12">
```

```
  <div>
```

```
    @if(Session::has('success'))
```



```

<div class="alert alert-success" >

    <button    type="button"    class="close"    data-dismiss="alert"    aria-
label="Close">&times;</button>

    {{ Session::get('success')}}

</div>

@endif

</div>

<table id="simple-table" class="table table-bordered table-hover">

    <thead>

        <tr>

            <!-- <th>id</th> -->

            <th>Nombre</th>

            <th>Apellido</th>

            <th>Cédula</th>

            <th>Correo Electrónico</th>

            <th>Estado</th>

            <th>Acciones</th>

        </tr>

    </thead>

    <tbody>

        @php $estado = [1 => 'ACTIVO', 0 => 'INACTIVO']; @endphp

        @if($informacion->count())

            @foreach($informacion as $info)

                <tr id="{{ $info->id }}">

```

```

<td>{{ $info->nombre }}</td>

<td>{{ $info->apellido }}</td>

<td>{{ $info->cedula }}</td>

<td>{{ $info->email }}</td>

<td>{{ $estado[$info->estado] }}</td>

```

```

<td>

```

```

    { { -- @php use Illuminate\Support\Facades\Crypt; @endphp

    @php    $desencriptar    =    Crypt::decryptString($info->password);

@endphp -- } }

```

```

<div class="text-center">

    <button class="btn btn-xs btn-info"

        data-id="{{ $info->id }}"

        data-nombre="{{ $info->nombre }}"

        data-apellido="{{ $info->apellido }}"

        data-cedula="{{ $info->cedula }}"

        data-email="{{ $info->email }}"

        data-password="{{ $info->password }}"

        data-estado="{{ $info->estado }}"

        data-toggle="modal"          data-target="#editUsuario"          data-

toggle="tooltip" title="Editar Usuario">

        <i class="ace-icon fa fa-pencil bigger-120"></i></button>

        <!--          <button          class="btn          btn-danger          btn-xs"

onclick="eliminarUbicacion('{{ $info->id }}')"><span class="glyphicon glyphicon-trash"

></span></button> -->

```

```

        </div>

    </td>

</tr>

@endforeach

@else

<tr>

    <td colspan="8" class="text-center"> No Existen Registros </td>

</tr>

@endif

</tbody>

</table>

{{ $informacion->links() }}

</div>

</div>

</div> <!-- /.row -->

<!-- NUEVO USUARIO -->

<div id="nuevoUsuario" class="modal fade" data-backdrop="static" tabindex="-1"
role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="false">

    <br>

    <br>

    <div class="modal-dialog">

        <div class="modal-content">

            <div class="modal-header">

```

```
        <button      type="button"      class="close"      data-dismiss="modal"      aria-  
hidden="true">&times;</button>
```

```
    <h3 class="smaller lighter blue no-margin"> Nuevo Usuario</h3>
```

```
</div>
```

```
<div class="modal-body ">
```

```
    <div class="row">
```

```
        <div class="col-xs-12">
```

```
            <form class="form-horizontal" id="formNuevoUsuario" role="form" >
```

```
                <div id="div-nombre" class="form-group">
```

```
                    <label class="col-sm-5 control-label no-padding-right" > Nombre *  
</label>
```

```
                    <div class="col-sm-7" id="div-nombre-error">
```

```
                        <input      type="text"      id="nombre"      name="nombre"  
placeholder="Nombre" class="col-xs-10 col-sm-5" />
```

```
                    </div>
```

```
                </div>
```

```
            <div id="div-apellido" class="form-group">
```

```
                <label class="col-sm-5 control-label no-padding-right" > Apellido *  
</label>
```

```
                <div class="col-sm-7" id="div-apellido-error">
```

```
                    <input      type="text"      id="apellido"      name="apellido"  
placeholder="Apellido" class="col-xs-10 col-sm-5" />
```

```
                </div>
```

```
            </div>
```

```

<div id="div-cedula" class="form-group">

    <label class="col-sm-5 control-label no-padding-right" > Cédula *

</label>

    <div class="col-sm-7" id="div-cedula-error">

        <input type="text" id="cedula" name="cedula" placeholder="Cédula"

class="col-xs-10 col-sm-5" />

    </div>

</div>

<div id="div-email" class="form-group">

    <label class="col-sm-5 control-label no-padding-right" > Correo

Electrónico * </label>

    <div class="col-sm-7" id="div-email-error">

        <input type="text" id="email" name="email" placeholder="Correo

Electrónico" class="col-xs-10 col-sm-5" />

    </div>

</div>

<div id="div-password" class="form-group">

    <label class="col-sm-5 control-label no-padding-right" > Contraseña *

</label>

    <div class="col-sm-7" id="div-password-error">

        <input type="password" id="password" name="password"

placeholder="Contraseña" class="col-xs-10 col-sm-5" />

    </div>

</div>

```

```

<div id="div-password_confirm" class="form-group">

    <label class="col-sm-5 control-label no-padding-right" > Confirmar
    Contraseña * </label>

    <div class="col-sm-7" id="div-password_confirm-error">

        <input
            type="password"
            id="password_confirm"
            name="password_confirm" placeholder="Confirmar Contraseña" class="col-xs-10 col-sm-5"
        />

    </div>

</div>

</div>

</div>

</div>

</div>

<div class="modal-footer center">

    <button type="submit" class="btn btn-success" id="guardarBoton" >

        <i class="ace-icon fa fa-floppy-o "></i>

        Guardar

    </button>

</div>

</form>

</div><!-- /.modal-body -->

</div><!-- /.modal-content -->

</div><!-- /.modal-dialog -->

</div>

{{-- EDITAR Usuario --}}

<div class="modal fade" name="editUsuario" id="editUsuario" data-backdrop="static"

```

```

tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="false">

<div class="modal-dialog">

  <div class="modal-content">

    <div class="modal-header">

      <button      type="button"      class="close"      data-dismiss="modal"      aria-
hidden="true">&times;</button>

      <h3 class="smaller lighter blue no-margin"> Editar Usuario</h3>

    </div>

    <div class="modal-body ">

      <div class="row">

        <div class="col-xs-12">

          <form class="form-horizontal"  id="formEditUsuario" role="form" >

            <input type="hidden" name="id" id="edit-id">

            <div id="div-edit-nombre" class="form-group ">

              <label  class="col-sm-3  control-label  no-padding-right"  >  Nombre*

</label>

              <div class="col-sm-9">

                <input type="text" id="edit-nombre" name="nombre" class="col-xs-10
col-sm-5" />

              </div>

            </div>

          </div>

          <div id="div-edit-apellido" class="form-group">

            <label  class="col-sm-3  control-label  no-padding-right"  >  Apellido*

</label>

            <div class="col-sm-9" id="div-edit-apellido-error">

```

```
        <input type="text" id="edit-apellido" name="apellido" class="col-xs-
10 col-sm-5" />

    </div>

</div>

<div id="div-edit-cedula" class="form-group">

    <label class="col-sm-3 control-label no-padding-right" > Cédula*

</label>

    <div class="col-sm-9" id="div-edit-cedula-error">

        <input type="text" id="edit-cedula" name="cedula" class="col-xs-10
col-sm-5" />

    </div>

</div>

<div id="div-edit-email" class="form-group">

    <label class="col-sm-3 control-label no-padding-right" > Correo
Electrónico* </label>

    <div class="col-sm-9" id="div-edit-email-error">

        <input type="text" id="edit-email" name="email" class="col-xs-10 col-
sm-5" />

    </div>

</div>

<div id="div-edit-password" class="form-group">

    <label class="col-sm-3 control-label no-padding-right" > Contraseña*

</label>

    <div class="col-sm-9" id="div-edit-password-error">

        <input type="text" id="edit-password" name="password" class="col-
```



```
xs-10 col-sm-5" />
```

```
</div>
```

```
</div>
```

```
<div id="div-edit-estado" class="form-group">
```

```
<label class="col-sm-3 control-label no-padding-right" > Estado* </label>
```

```
<div class="col-sm-9" id="div-edit-estado-error">
```

```
<input type="text" id="edit-estado" name="estado" class="col-xs-10
```

```
col-sm-5" />
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="modal-footer center">
```

```
<button type="submit" class="btn btn-success" id="editarBoton" >
```

```
<i class="ace-icon glyphicon glyphicon-refresh" :class="submitting ? 'fa-  
spinner' : 'fa-refresh'"></i> Actualizar
```

```
</button>
```

```
</div>
```

```
</form>
```

```
</div>
```

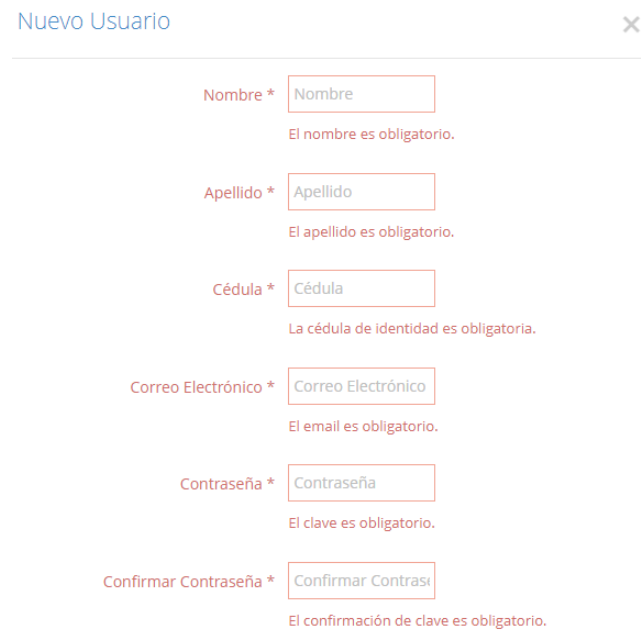
```
</div>
```

```
</div>
```

```
@endsection
```

Fig. No. 11.

Validaciones



The screenshot shows a web form titled "Nuevo Usuario" with a close button (X) in the top right corner. The form contains six input fields, each with a red border and a red asterisk indicating a required field. Below each input field is a red error message. The fields and messages are:

- Nombre ***: Input field contains "Nombre". Error message: "El nombre es obligatorio."
- Apellido ***: Input field contains "Apellido". Error message: "El apellido es obligatorio."
- Cédula ***: Input field contains "Cédula". Error message: "La cédula de identidad es obligatoria."
- Correo Electrónico ***: Input field contains "Correo Electrónico". Error message: "El email es obligatorio."
- Contraseña ***: Input field contains "Contraseña". Error message: "El clave es obligatorio."
- Confirmar Contraseña ***: Input field contains "Confirmar Contrase". Error message: "El confirmación de clave es obligatorio."

Para la creación de las validaciones se creó con el framework de laravel “crearUsuarioRequest” en donde se realizan todas las validaciones.

Código

```
<?php
namespace App\Http\Requests\dashboard;
use Illuminate\Foundation\Http\FormRequest;
use Illuminate\Validation\Rules;
class crearUsuariosRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
```

```

*/

public function authorize()

{

    return true;

}

/**

 * Get the validation rules that apply to the request.

 *

 * @return array

 */

public function rules()

{

    return [

        'nombre' => 'required|max:40',

        'apellido' => 'required|max:40',

        'cedula' => 'required|regex:/^\d*$|unique:users|size:10',//cedula no se repita y sea de
10 caracteres

        'email' => 'required|unique:users|email:rfc',

        //'password' => ['required','confirmed', Rules\Password::defaults()],

        'password' => 'required|min:8',

        'password_confirm' => 'required|min:8',

    ];

}

public function messages()

{


```

```

return [

    'required'      => 'El :attribute es obligatorio.',

    'max'           => 'El :attribute debe tener máximo :max caracteres.',

    'unique'        => 'El :attribute ya existe.',

    'cedula.unique'  => 'La :attribute ya existe.',

    'cedula.required' => 'La :attribute es obligatoria.',

    'login'         => 'El :attribute es obligatorio.',

    'email'         => 'El :attribute no tiene el formato correcto.',

    'min'           => 'El :attribute debe tener al menos :min caracteres.',

    'cedula.regex'   => 'La :attribute debe tener solamente caracteres numéricos.',

    'cedula.size'    => 'La :attribute debe tener :size caracteres.',

    'regex'         => 'El :attribute debe tener solamente caracteres numéricos'

];

}

public function attributes()
{
    return [

        'cedula'      => 'cédula de identidad',

        'login'       => 'usuario',

        'password'     => 'clave',

        'password_confirm' => 'confirmación de clave'

    ];

}

}

```

BAL8 Claves encriptadas

Fig. No. 12.

Claves encriptadas

```
password
$2y$10$rlOCtgmCFs3MB8Hzjs0Q0ehBsYsahu2V5s.j6L5luCV...
eyJpdil6lk1tUGtycVF4QjE3aG9nbmNQeIFvT0E9PSIsInZhbH...
eyJpdil6lk9Kbk1ZTGFGTTN4bkZwOWxhK2hNU2c9PSIsInZhbH...
eyJpdil6ljZ5K0VOWFd0akJpOFZMSTBVYWxlZ1E9PSIsInZhbH...
```

Para la encriptación de las claves se utilizó la función criptográfica Hash que en laravel se lo llama a través del controlador mediante la clase “Crypt”.

Código

```
$usuario ->password = Crypt::encrypt($request ->password);
```

4.4.2 Sprint Parámetros

Tabla 6

Sprint Parámetros

HISTORIAS DE USUARIO					
ID	PRIORIDAD	COMO...	QUIERO...	PARA...	
BAL38	11	Administrador	Menú dinámico de opciones	Seleccionar, seguridades, indicadores y dashboard	
BAL1	12	Cooperativa	Acceder a la BD del CORE	Obtener información	
BAL3	13	Cooperativa	Almacenar información de un periodo	Recuperar indicadores históricos	
BAL4	14	Cooperativa	Almacenar información de autoridades en parámetros	Recuperar la información de los parámetros	
BAL12	15	Usuario	No duplicar información de los directivos	Almacenar parámetros	en

BAL38	16	Administrador	Menú dinámico de opciones	Seleccionar, seguridades, indicadores y dashboard
BAL39	17	Administrador	Crear menú de Parámetros de representantes	Seleccionar opciones para almacenar información
BAL40	18	Usuario	Ingresar información del consejo de Administración	Almacenar información del Consejo de Administración
BAL41	19	Usuario	Ingresar información del consejo de Vigilancia	Almacenar información del Consejo de Vigilancia
BAL42	20	Usuario	Ingresar información de educación financiera	Almacenar información de la participación de la educación financiera

4.4.3 Sprint Indicadores

Tabla 7

Sprint Indicadores

Sprint Indicadores				
ID	PRIORIDAD	COMO...	QUIERO...	PARA...
BAL2	21	Cooperativa	Seleccionar un periodo mensual	Obtener información de este periodo
BAL14	22	Usuario	Seleccionar información de depósitos por sexo	Generar indicador
BAL15	23	Usuario	Seleccionar información de depósitos por actividad económica	Generar indicador
BAL16	24	Usuario	Seleccionar información de depósitos por instrucción	Generar indicador
BAL17	25	Usuario	Seleccionar información de depósitos por grupo étnico	Generar indicador
BAL18	26	Usuario	Seleccionar información de depósitos por tipo de	Generar indicador

BAL19	27	Usuario	canal Seleccionar información de productos de tipo de crédito y por sexo	Generar indicador
BAL20	28	Usuario	Seleccionar información de cartera por tipo de crédito y actividad económica	Generar indicador
BAL21	29	Usuario	Seleccionar información de cartera por tipo de crédito y por nivel de instrucción	Generar indicador
BAL22	30	Usuario	Seleccionar información de cartera por tipo de cartera y actividad económica	Generar indicador
BAL23	31	Usuario	Seleccionar información de cartera por tipo de crédito y por provincia	Generar indicador
BAL24	32	Usuario	Seleccionar morosidad por sexo	Generar indicador
BAL25	33	Usuario	Seleccionar morosidad por sexo y actividad económica	Generar indicador
BAL26	34	Usuario	Seleccionar morosidad por sexo y por instrucción	Generar indicador
BAL27	35	Usuario	Seleccionar morosidad por sexo y por rango de edades	Generar indicador
BAL28	36	Usuario	Seleccionar socios por sexo de representantes	Generar indicador
BAL29	37	Usuario	Seleccionar socios por sexo, rango de edades de representantes	Generar indicador
BAL30	38	Usuario	Seleccionar información de Educación financiera por sexo	Generar indicador
BAL31	39	Usuario	Seleccionar información de Educación financiera por sexo y por edades	Generar indicador

Incidencia BAL2

Fig. No. 13.

Seleccionar Periodo

Periodo

SELECCIONE FECHA PARA ALMACENAR BALANCE SOCIAL

NUEVA FECHA DE REGISTRO



FECHA DE CONSULTA

Seleccione una fecha

✓

Enviar Fecha

Código:

Se creo una vista en donde se seleccionará el periodo para poder realizar las consultas y posteriormente mostrar la información.

```
@extends('Depositos.plantillaGeneral')
```

```
@section('content')
```

```
@if (session('status'))
```

```
{{session('status')}}  
<div class="alert alert-success" role="alert"></div>
```

```
@endif
```

```
<div class="page-header"><h1>Periodo</h1></div><!-- /.page-header -->
```

```
<!-- qui va el contenido de la tabla -->
```

```
<div class="row">
```

```
<div class="col-xs-12">
```

```
<!-- PAGE CONTENT BEGINS -->
```



```

<!-- INICIO FECHA DE NUEVO REGISTRO -->

<div class="widget-box"><!-- widget-box -->

    <div class="widget-header"><!-- widget-header -->

        <h5 class="widget-title text-primary">SELECCIONE FECHA PARA
ALMACENAR BALANCE SOCIAL</h5>

        <span class="widget-toolbar">

            <a href="#" data-action="collapse">

                <i class="ace-icon fa fa-chevron-up bigger-125"></i>

            </a>

        </span>

    </div>

    <div class="widget-body"><!-- widget-body -->

        <div class="widget-main"><!-- widget-main -->

            <!-- INICIO DEL CUERPO -->

            <div class="row">

                <div class="col-md-offset-4 col-md-4">

                    { {-- @foreach ($totaldep as $item)

                        <tr>{ { $item } }</tr>

                    @endforeach --} }

                <label class="col-md-offset-3 text-primary">NUEVA FECHA DE
REGISTRO</label>

                <div class="input-group">

                    <span class="input-group-addon">

                        <i class="fa fa-calendar bigger-110"></i>

```

```

        </span>

        <input class="form-control date-picker" id="date-picker" type="text"
name="date-picker" value="" data-date-format="dd-mm-yyyy" readonly />

    </div>

</div>

<div class="row">

    <div class="col-md-offset-4 col-md-4">

        <label      class="col-md-offset-3      text-primary">FECHA      DE
CONSULTA</label>

        <select  class="chosen-select"  data-placeholder="Fecha  de  Consulta..."
style="width:350px;" tabindex="1" name="fechaConsulta" id="fechaConsulta" >

            <option value="">Seleccione una fecha</option>

        </select>

    </div>

</div>

<div class="space space-8"></div>

<div class="row">

    <div class="col-md-offset-4 col-md-4">

        <button      class="btn      btn-success      col-md-12"      type="submit"
id="btnRegistrarFecha" disabled >

            <i class="ace-icon fa fa-check bigger-110"></i>

```

```

        Enviar Fecha

    </button>

</div>

</div>

<!-- FIN DEL CUERPO -->

</div><!-- ./widget-main -->

</div><!-- ./widget-body -->

</div><!-- ./widget-box -->

<!-- FIN FECHA DE NUEVO REGISTRO -->

<!-- PAGE CONTENT ENDS -->

</div><!-- /.col -->

</div><!-- /.row -->

@endsection

```

Incidencia BAL14

Fig. No. 14.

Menú indicadores y opción depósitos



Fig. No. 15.

Indicador Depósitos por sexo

Depósitos

SALDO DE DEPÓSITOS POR SEXO				
PRODUCTOS	TOTAL DEPÓSITOS HOMBRES	TOTAL DEPÓSITOS MUJERES	TOTAL DEPÓSITOS JURÍDICOS	SALDO TOTAL DEPÓSITOS
DEPÓSITO AHORROS	571407.18	672498.11	69018.2	1312923.49

Código

Se creó un código para poder mostrar el menú de los indicadores y la opción de depósitos.

```
<li class="">
```

```
  <a href="#" class="dropdown-toggle">
```

```
    <i class="menu-icon fa fa-user"></i>
```

```
    <span class="menu-text"> Indicadores </span>
```

```
    <b class="arrow fa fa-angle-down"></b>
```

```
  </a>
```

```
  <b class="arrow"></b>
```

```
  <ul class="submenu">
```

```
    <li class="">
```

```
      <a href="{ { route ('indexDepositos') } }">
```

```
        <i class="menu-icon fa fa-caret-right"></i>
```

```
        Depósitos
```

```
      </a>
```

<b class="arrow">

Controlador

Se creó un controlador llamado “depositosController” donde se colocan las funciones necesarias para realizar la consulta y posteriormente mostrar la información.

<?php

```
namespace App\Http\Controllers;
```

```
use App\Models\Depositos;
```

```
use Illuminate\Support\Facades\DB;
```

```
use Illuminate\Http\Request;
```

```
class depositosController extends Controller
```

```
{
```

```
    public function indexDepositos (Request $request)
```

```
    {
```

```
        // $informacion = User::paginate(7);
```

```
//se buscan los productos de la cooperativa se encera la matriz
```

```
$productos = DB::connection('uniotavalo')->table('cuenta')
```

```
->join('producto', 'cuenta.serial_pro', '=', 'producto.serial_pro')
```

```
->whereRaw("estado_cta IN ('A','T')")
```

```

->whereRaw('cuenta.serial_pro NOT IN (23)')

->selectRaw('cuenta.serial_pro, nombre_pro')

->groupByRaw('cuenta.serial_pro')

->orderByRaw('serial_pro, producto.nombre_pro asc')

->get();

foreach ($productos as $key => $value)
{
    $nompro  [$value->serial_pro]      = $value->nombre_pro;

    $cantdep [$value->serial_pro]['hombres'] = 0;

    $cantdep [$value->serial_pro]['mujeres'] = 0;

    $cantdep [$value->serial_pro]['juridico'] = 0;

    $montodep[$value->serial_pro]['hombres'] = 0;

    $montodep[$value->serial_pro]['mujeres'] = 0;

    $montodep[$value->serial_pro]['juridico'] = 0;

    $montodep[$value->serial_pro]['total']   = 0;
}

//se obtiene todas las cuentas

// $cuentas = Load::model('cuenta')->obtenerCuentas();

$cuentas = DB::connection('uniotavalo')->table('cliente')

->join('cuentaCliente', 'cuentaCliente.serial_cli', '=', 'cliente.serial_cli')

->join('cuenta', 'cuenta.serial_cta', '=', 'cuentaCliente.serial_cta')

->whereRaw('"estado_cta IN ('A','T') AND cuentaCliente.tipo_ccl = 'T' AND
cuenta.serial_pro NOT IN (23)')

->selectRaw('serial_pro, cliente.serial_cli, naturaleza_cli, valorDisponibleAhorros_cta,
valorRetencionAhorros_cta, totalBloqueoAhorros_cta, encaje_cta')

```

```

->get();

//se recorren las cuentas

foreach ($cuentas as $value)
{
    if($value->naturaleza_cli == 'N')
    {
        // $representante = Load::model('representante')-
>obtenerRepresentante($value->serial_cli);

        $representante = DB::connection('uniotavalo')->table('cliente')

        ->join('representanteCliente', 'representanteCliente.serial_cli', '=',
'cliente.serial_cli')

        ->join('representante', 'representante.serial_rep', '=',
'representanteCliente.serial_rep')

        ->whereRaw("representanteCliente.estado_rcl = 'A' AND cliente.serial_cli =
$value->serial_cli limit 1")

        ->selectRaw('representante.serial_rep, sexo_rep, representante.serial_ciu')

        ->get();

        if (empty($representante[0]))
        {
            $cantdep[$value->serial_pro]['mujeres'] += 1;
        }

        else if ($representante[0]->sexo_rep == "M")
        {

```

```

        $cantdep [$value->serial_pro]['hombres'] += 1;

        $montodep[$value->serial_pro]['hombres'] += ($value-
>valorDisponibleAhorros_cta + $value->valorRetencionAhorros_cta + $value-
>totalBloqueoAhorros_cta + $value->encaje_cta);

        $montodep[$value->serial_pro]['total'] += ($value-
>valorDisponibleAhorros_cta + $value->valorRetencionAhorros_cta + $value-
>totalBloqueoAhorros_cta + $value->encaje_cta);

    }

    else if($representante[0]->sexo_rep == 'F')
    {

        $cantdep [$value->serial_pro]['mujeres'] += 1;

        $montodep[$value->serial_pro]['mujeres'] += ($value-
>valorDisponibleAhorros_cta + $value->valorRetencionAhorros_cta + $value-
>totalBloqueoAhorros_cta + $value->encaje_cta);

        $montodep[$value->serial_pro]['total'] += ($value-
>valorDisponibleAhorros_cta + $value->valorRetencionAhorros_cta + $value-
>totalBloqueoAhorros_cta + $value->encaje_cta);

    }

}

if($value->naturaleza_cli == 'J')
{

    $cantdep [$value->serial_pro]['juridico'] += 1;

    $montodep[$value->serial_pro]['juridico'] += ($value-
>valorDisponibleAhorros_cta + $value->valorRetencionAhorros_cta + $value-

```



```

>totalBloqueoAhorros_cta + $value->encaje_cta);

        $montodep[$value->serial_pro]['total'] += ($value-
>valorDisponibleAhorros_cta + $value->valorRetencionAhorros_cta + $value-
>totalBloqueoAhorros_cta + $value->encaje_cta);

    }

}

$depositodpf = "DEPÓSITO PLAZO FIJO";

$montodpf['hombres'] = 0;

$montodpf['mujeres'] = 0;

$montodpf['juridico'] = 0;

$montodpf['total'] = 0;


//se obtiene todos los depositos

$depositos = DB::connection('uniotavalo')->table('deposito')

->select( 'cliente.serial_cli', 'naturaleza_cli','deposito.valor_dep')

->join('agencia', 'agencia.serial_age', '=', 'deposito.serial_age')

->join('depositoCliente', 'deposito.serial_dep', '=', 'depositoCliente.serial_dep')

->join('cliente', 'depositoCliente.serial_cli', '=', 'cliente.serial_cli')

->whereRaw("tipo_dcl='T' and estado_dep IN('A','V','N')")

->orderByRaw('agencia.serial_age')

->get();


//se recorren todos los depositos

foreach ($depositos as $value)

{

```

```
if($value->naturaleza_cli == 'N')
{
    if (empty($representante[0]))
    {
        $montodpf['mujeres'] += $value->valor_dep;
        $montodpf['total'] += $value->valor_dep;
    }
    else if ($representante[0]->sexo_rep == "M")
    {
        $montodpf['hombres'] += $value->valor_dep;
        $montodpf['total'] += $value->valor_dep;
    }
    else if($representante[0]->sexo_rep == 'F')
    {
        $montodpf['mujeres'] += $value->valor_dep;
        $montodpf['total'] += $value->valor_dep;
    }
}

if($value->naturaleza_cli == 'J')
{
    $montodpf['juridico'] += $value->valor_dep;
    $montodpf['total'] += $value->valor_dep;
}
```

```
}
```

```
$depositoahorro = "DEPÓSITO AHORROS";
```

```
$totaldep['hombres'] = 0;
```

```
$totaldep['mujeres'] = 0;
```

```
$totaldep['juridico'] = 0;
```

```
$totaldep['total'] = 0;
```

```
foreach ($montodep as $key => $value)
```

```
{ foreach ($value as $key1 => $valor)
```

```
{
```

```
    if($key1 == 'hombres')
```

```
    {
```

```
        $totaldep['hombres'] += $valor;
```

```
        $totaldep['total'] += $valor;
```

```
    }
```

```
    if($key1 == 'mujeres')
```

```
    {
```

```
        $totaldep['mujeres'] += $valor;
```

```
        $totaldep['total'] += $valor;
```

```
    }
```

```
    if($key1 == 'juridico')
```

```
    {
```

```
        $totaldep['juridico'] += $valor;
```

```
        $totaldep['total'] += $valor;
```

```
    }
```

```
}  
  
}
```

Vista

Se creo una vista para desplegar la información obtenida de la consulta.

```
@extends('Depositos.plantillaGeneral')  
  
@section('content')  
  
@if (session('status'))  
  
{{session('status')}}  
  
<div class="alert alert-success" role="alert"></div>  
  
@endif  
  
  
  
  
<div class="page-header"><h1>Depósitos</h1></div><!-- /.page-header -->  
  
<!-- qui va el contenido de la tabla -->  
  
<div class="row">  
  
    <div class="col-xs-12">  
  
        <!-- PAGE CONTENT BEGINS -->  
  
  
  
  
        <!-- Saldos de depósitos por sexo -->  
  
        <div class="widget-box" id="">  
  
            <div class="widget-header">  
  
                <h5 class="widget-title text-primary">SALDO DE DEPÓSITOS POR  
SEXO</h5>  
  
            <div class="widget-toolbar">  
  
                <a href="#" data-action="collapse">
```

<i class="ace-icon fa fa-chevron-up bigger-110"></i>

</div>

</div>

<div class="widget-body">

<div class="widget-main">

<!-- Depositos por sexo -->

<table id="" class="table table-bordered table-hover">

<thead>

<tr>

<th>PRODUCTOS</th>

<th>TOTAL DEPÓSITOS HOMBRES</th>

<th>TOTAL DEPÓSITOS MUJERES</th>

<th>TOTAL DEPÓSITOS JURÍDICOS</th>

<th>SALDO TOTAL DEPÓSITOS</th>

</tr>

</thead>

<tbody>

<tr>

<td>{{ \$depositoahorro }}</td>

@foreach (\$totaldep as \$item)

<td class="text-right">{{ \$item }}</td>

```

        @endforeach

    </tr>

    <tr>

        <td>{{ $depositodpf }}</td>

        @foreach ($montodpf as $item)

            <td class="text-right">{{ $item }}</td>

        @endforeach

    </tr>

</tbody>

</table>

</div>

</div>

</div>

<!-- Fin Saldos de depósitos por sexo -->

<!-- Saldos de depósitos por sexo y tipo de cuenta-->

<div class="widget-box" id="">

    <div class="widget-header">

        <h5 class="widget-title text-primary">SALDO DE DEPÓSITOS POR
SEXO Y TIPO DE CUENTA</h5>

        <div class="widget-toolbar">

            <a href="#" data-action="collapse">

                <i class="ace-icon fa fa-chevron-up bigger-110"></i>

            </a>

        </div>

    </div>

</div>

```

```

<div class="widget-body">

<div class="widget-main">

<!-- Depósitos por sexo y tipo de cuenta-->

<table id="" class="table table-bordered table-hover">

<thead>

<tr>

<th class="text-center"><b>PRODUCTO</b></th>

<th class="text-center"><b>TOTAL HOMBRES</b></th>

<th class="text-center">TOTAL MUJERES</th>

<td class="text-center"><b>TOTAL JUR&Iacute;DICOS</b></td>

<td class="text-center"><b>TOTAL POR PRODUCTO</b></td>

</tr>

</thead>

<tbody>

@foreach ($montodep as $key => $item)

<tr>

<td class="text-right">{{ $nompro[$key]}}</td>

<td class="text-right">{{ $item["hombres"]}}</td>

<td class="text-right">{{ $item["hombres"]}}</td>

<td class="text-right">{{ $item["mujeres"]}}</td>

<td class="text-right">{{ $item["juridico"]}}</td>

<td class="text-right">{{ $item["total"]}}</td>

</tr>

@endforeach

```

```

        </tbody>

    </table>

</div>

</div>

</div>

<!-- Saldos de depósitos por sexo y tipo de cuenta-->

<!-- PAGE CONTENT ENDS -->

</div><!-- /.col -->

</div><!-- /.row -->

```

@endsection

4.4.4 Dashboard Indicadores

Tabla 8

Dashboard Indicadores

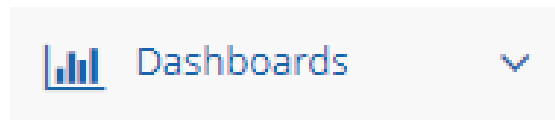
| Dashboard Indicadores | | | | |
|-----------------------|-----------|---------------|------------------------------------|-------------------------------------|
| ID | PRIORIDAD | COMO... | QUIERO... | PARA... |
| BAL37 | 40 | Administrador | Parametrizar menú de los dashboard | Imprimir indicadores gráficos. |
| BAL2 | 41 | Cooperativa | Seleccionar un periodo mensual | Obtener información de este periodo |
| BAL6 | 42 | Cooperativa | Gráficos de los | Publicar los |

| | |
|-------------|-------------|
| indicadores | indicadores |
|-------------|-------------|

Incidencia BAL37

Fig. No 16.

Pestaña Menú Dashboards



Código

```
<li class="">
  <a href="#" class="dropdown-toggle">
    <i class="menu-icon fa fa-bar-chart"></i>
    <span class="menu-text"> Dashboards </span>
    <b class="arrow fa fa-angle-down"></b>
  </a>
  <b class="arrow"></b>
```

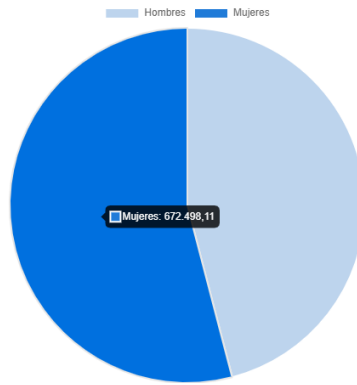
Incidencia BAL6

El aplicativo permite desplegar un menú dinámico de los siguientes indicadores.

Dashboard Depósitos Ahorros.

Fig. No 17.

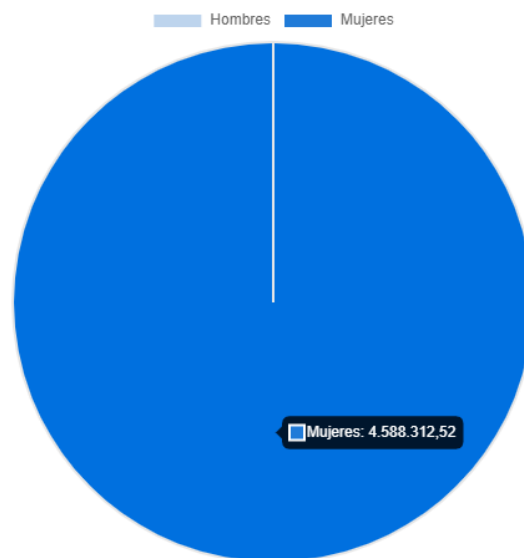
Dashboard Depósitos Ahorros



Dashboard Depósitos Plazos fijos.

Fig. No 18.

Dashboard Depósitos Plazos Fijos.



Código.

Para la creación de este dashboard se implementó un controlador llamado:

“DashboardDepositosController.php” en donde se encuentran los oficios primordiales que llaman a la información de la base de datos y poder almacenarlos.

Código del controlador.

```
namespace App\Http\Controllers;
```

```
use App\Models\Depositos;
```

```
use Illuminate\Support\Facades\DB;
```

```
use Illuminate\Http\Request;
```

```
class DashboardDepositosController extends Controller
```

```
{
```

```
    public function indexDashboardDepositos (Request $request)
```

```
    {
```

```
        $productos = DB::connection('uniotavalo')->table('cuenta')
```

```
        ->join('producto', 'cuenta.serial_pro', '=', 'producto.serial_pro')
```

```
        ->whereRaw('"estado_cta IN ('A','I')")
```

```
        ->whereRaw('cuenta.serial_pro NOT IN (23)')
```

```
        ->selectRaw('cuenta.serial_pro, nombre_pro')
```

```
        ->groupByRaw('cuenta.serial_pro')
```

```
        ->orderByRaw('serial_pro, producto.nombre_pro asc')
```

```
        ->get();
```

```
        foreach ($productos as $key => $value)
```

```
        {
```

```
            $nompro  [$value->serial_pro]      = $value->nombre_pro;
```

```
            $cantdep [$value->serial_pro]['hombres'] = 0;
```

```
            $cantdep [$value->serial_pro]['mujeres'] = 0;
```

```
            $cantdep [$value->serial_pro]['juridico'] = 0;
```

```
            $montodep[$value->serial_pro]['hombres'] = 0;
```

```
            $montodep[$value->serial_pro]['mujeres'] = 0;
```

```
            $montodep[$value->serial_pro]['juridico'] = 0;
```

```

        $montodep[$value->serial_pro]['total']    = 0;
    }

    //se obtiene todas las cuentas

    // $cuentas = Load::model('cuenta')->obtenerCuentas();

    $cuentas = DB::connection('uniotavalo')->table('cliente')

    ->join('cuentaCliente', 'cuentaCliente.serial_cli', '=', 'cliente.serial_cli')

    ->join('cuenta', 'cuenta.serial_cta', '=', 'cuentaCliente.serial_cta')

    ->whereRaw("estado_cta IN ('A','T') AND cuentaCliente.tipo_ccl = 'T' AND
cuenta.serial_pro NOT IN (23)")

    ->selectRaw('serial_pro, cliente.serial_cli, naturaleza_cli, valorDisponibleAhorros_cta,
valorRetencionAhorros_cta, totalBloqueoAhorros_cta, encaje_cta')

    ->get();

    //se recoren las cuentas

    foreach ($cuentas as $value)
    {
        if($value->naturaleza_cli == 'N')
        {
            // $representante = Load::model('representante')-
>obtenerRepresentante($value->serial_cli);

            $representante = DB::connection('uniotavalo')->table('cliente')

            ->join('representanteCliente', 'representanteCliente.serial_cli', '=',
'cliente.serial_cli')

            ->join('representante', 'representante.serial_rep', '=',

```

```

'representanteCliente.serial_rep')

->whereRaw("representanteCliente.estado_rcl = 'A' AND cliente.serial_cli =
$value->serial_cli limit 1")

->selectRaw('representante.serial_rep, sexo_rep, representante.serial_ciu')

->get();

if (empty($representante[0]))
{
    $cantdep[$value->serial_pro]['mujeres'] += 1;
}
else if ($representante[0]->sexo_rep == "M")
{

    $cantdep [$value->serial_pro]['hombres'] += 1;

    $montodep[$value->serial_pro]['hombres'] += ($value-
>valorDisponibleAhorros_cta + $value->valorRetencionAhorros_cta + $value-
>totalBloqueoAhorros_cta + $value->encaje_cta);

    $montodep[$value->serial_pro]['total'] += ($value-
>valorDisponibleAhorros_cta + $value->valorRetencionAhorros_cta + $value-
>totalBloqueoAhorros_cta + $value->encaje_cta);

}
else if ($representante[0]->sexo_rep == 'F')
{

    $cantdep [$value->serial_pro]['mujeres'] += 1;

    $montodep[$value->serial_pro]['mujeres'] += ($value-

```

```

>valorDisponibleAhorros_cta + $value->valorRetencionAhorros_cta + $value-
>totalBloqueoAhorros_cta + $value->encaje_cta);

        $montodep[$value->serial_pro]['total'] += ($value-
>valorDisponibleAhorros_cta + $value->valorRetencionAhorros_cta + $value-
>totalBloqueoAhorros_cta + $value->encaje_cta);

    }

}

if($value->naturaleza_cli == 'J')
{
    $cantdep [$value->serial_pro]['juridico'] += 1;

    $montodep[$value->serial_pro]['juridico'] += ($value-
>valorDisponibleAhorros_cta + $value->valorRetencionAhorros_cta + $value-
>totalBloqueoAhorros_cta + $value->encaje_cta);

        $montodep[$value->serial_pro]['total'] += ($value-
>valorDisponibleAhorros_cta + $value->valorRetencionAhorros_cta + $value-
>totalBloqueoAhorros_cta + $value->encaje_cta);

    }

}

```

```

$depositodpf = "DEPÓSITO PLAZO FIJO";

```

```

$montodpf['hombres'] = 0;

```

```

$montodpf['mujeres'] = 0;

```

```

$montodpf['juridico'] = 0;

```

```
$montodpf['total'] = 0;
```

```
//se obtiene todos los depositos
```

```
$depositos = DB::connection('uniotavalo')->table('deposito')
```

```
->select( 'cliente.serial_cli', 'naturaleza_cli','deposito.valor_dep')
```

```
->join('agencia', 'agencia.serial_age', '=', 'deposito.serial_age')
```

```
->join('depositoCliente', 'deposito.serial_dep', '=', 'depositoCliente.serial_dep')
```

```
->join('cliente', 'depositoCliente.serial_cli', '=', 'cliente.serial_cli')
```

```
->whereRaw("tipo_dcl='T' and estado_dep IN('A','V','N')")
```

```
->orderByRaw('agencia.serial_age')
```

```
->get();
```

```
//se recorren todos los depositos
```

```
foreach ($depositos as $value)
```

```
{
```

```
    if($value->naturaleza_cli == 'N')
```

```
    {
```

```
        if (empty($representante[0]))
```

```
        {
```

```
            $montodpf['mujeres'] += $value->valor_dep;
```

```
            $montodpf['total'] += $value->valor_dep;
```

```
        }
```

```
    else if ($representante[0]->sexo_rep == "M")
```

```
    {
```

```
        $montodpf['hombres'] += $value->valor_dep;
```

```

        $montodpf['total'] += $value->valor_dep;
    }

    else if($representante[0]->sexo_rep == 'F')
    {
        $montodpf['mujeres'] += $value->valor_dep;
        $montodpf['total'] += $value->valor_dep;
    }

}

if($value->naturaleza_cli == 'J')
{
    $montodpf['juridico'] += $value->valor_dep;
    $montodpf['total'] += $value->valor_dep;
}

}

$depositoahorro = "DEPÓSITO AHORROS";

$totaldep['hombres'] = 0;

$totaldep['mujeres'] = 0;

$totaldep['juridico'] = 0;

$totaldep['total'] = 0;

foreach ($montodep as $key => $value)
{
    foreach ($value as $key1 => $valor)
    {

```



```

        if($key1 == 'hombres')
        {
            $totaldep['hombres'] += $valor;

            $totaldep['total'] += $valor;
        }

        if($key1 == 'mujeres')
        {
            $totaldep['mujeres'] += $valor;

            $totaldep['total'] += $valor;
        }

    }

}

return

view('DashboardsIndicadores.DashboardDepositos.dashboardDepositos',compact('totaldep',
'montodpf', 'montodep', 'nompro', 'depositodpf', 'depositoahorro'));

}

}

```

Código de la vista del gráfico de ahorros.

```

@extends('Depositos.plantillaGeneral')

@section('content')

@if (session('status'))

{{session('status')}}

```

<div class="alert alert-success" role="alert"></div>

@endif

<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/3.8.0/chart.min.js" integrity="sha512-sW/w8s4RWTdFFSduOTGt4isV1+190E/GghVffMA9XczdJ2MDzSzLEubKAs5h0wzgSJO QTRYyaz73L3d6RtJSg==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>

<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/3.8.0/chart.esm.js" integrity="sha512-YM18yiANXJFpbiOZjLzUrK/INfTiBcwtTLeAntG4B8dJY+NdUDjxfPNGPEMuOdXIT7U/uT+zbIvbQYAEFog+MA==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/3.8.0/chart.esm.min.js" integrity="sha512-yPOQ2pPoQ9JtP0/jDKpXiKyWNCJWT5OI+6r1EqZmTg+afKQOBpy08VYboeq+Tt9kl9/FOCueEhH6cmHN3nAdJA==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/3.8.0/chart.js" integrity="sha512-5m2r+g00HDHnhXQDbRLAfZBwPpPCaK+wPLV6lm8VQ+09ilGdHfXV7IVyKPkLOTfi4vTTUVJnz7ELs7cA87/GMA==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/3.8.0/helpers.esm.js" integrity="sha512-dSutS1n8KEMUnQMa9YGa6CxAmoUfaZdxL2+s2xBgEq7WHaWdtjna/rzGsjqkT27GxKB DLT0Fr3C/TzzHvBRaAg==" crossorigin="anonymous" referrerpolicy="no-

```

referrer"></script>

<script      src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/3.8.0/helpers.esm.min.js"
integrity="sha512-
vxCPccgWacJoW2HlxhlKKtczdzvcg0r1UuB9LfNGt6vsDbgLfSFxKlOlUS2mqKNXrOK5b9
3S45309T+V5BhueA==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>

<script>                                                                                               src=
"https://cdn.jsdelivr.net/combine/npm/chart.js@3.8.0/dist/chart.esm.min.js,npm/chart.js@3.8.
0/dist/chart.min.js,npm/chart.js@3.8.0,npm/chart.js@3.8.0/dist/helpers.esm.min.js"</script>

<script> src= "https://cdn.jsdelivr.net/npm/chart.js@3.8.0/dist/chart.esm.min.js"</script>
<script> src= "https://cdn.jsdelivr.net/npm/chart.js@3.8.0/dist/chart.min.js"</script>
<script> src= "https://cdn.jsdelivr.net/npm/chart.js@3.8.0/dist/chart.min.js"</script>
<script>src= "https://cdn.jsdelivr.net/npm/chart.js@3.8.0/dist/helpers.esm.min.js"</script>

<div class="page-header"><h1>Depósitos</h1></div><!-- /.page-header -->

<!-- qui va el contenido de la tabla -->

<div class="row" >

  <div class="col-xs-3">

</div>

  <div class="col-xs-5" >

    <div>

      <canvas id="myChart"></canvas>

    </div>

    <script>

      var depositos=@json($totaldep);

```

```
const labels = ['Hombres','Mujeres'];

const data = {

  labels: labels,

  datasets: [{

    label: 'Usuarios Activos',

    backgroundColor: ['#bdd4ed','#207bd8'],

    borderColor: 'white',

    data: [depositos['hombres'],depositos['mujeres']],

  }]

};

const config = {

type: 'pie',

data: data,

};

</script>

<script>

const myChart = new Chart(

  document.getElementById('myChart'),

  config

);

</script>

</div><!-- /.col -->
```

<div class="col">

</div>

</div><!-- /.row -->

@endsection

Código de la vista del grafico de depósitos a plazos fijos.

@extends('Depositos.plantillaGeneral')

@section('content')

@if (session('status'))

{{session('status')}}</div>

<div class="alert alert-success" role="alert"></div>

@endif

<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/3.8.0/chart.min.js"

integrity="sha512-

sW/w8s4RWTdFFSduOTGtk4isV1+190E/GghVffMA9XczdJ2MDzSzLEubKAs5h0wzgSJO

QTRYyaz73L3d6RtJSg==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>

<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/3.8.0/chart.esm.js"

integrity="sha512-

YM18yiANXJFpbiOZjLzUrK/INfTiBcwtTLeAntG4B8dJY+NdUDjxfPNGPEMuOdXIT7U/

uT+zbIvbQYAEFog+MA==" crossorigin="anonymous" referrerpolicy="no-

referrer"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/3.8.0/chart.esm.min.js"

integrity="sha512-

yPOQ2pPoQ9JtP0/jDKpXiKyWNCJWT5OI+6r1EqZmTg+afKQOBpy08VYboeq+Tt9kl9/F

OCueEhH6cmHN3nAdJA==" crossorigin="anonymous" referrerpolicy="no-

```
referrer"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/3.8.0/chart.js" integrity="sha512-
5m2r+g00HDHnhXQDbRLAfZBwPpPCaK+wPLV6lm8VQ+09ilGdHfXV7IVyKPkLOTfi4
vTTUVJnz7ELs7cA87/GMA==" crossorigin="anonymous" referrerpolicy="no-
referrer"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/3.8.0/helpers.esm.js"
integrity="sha512-
dSutS1n8KEMUnQMa9YGa6CxAmoUfaZdxL2+s2xBgEq7WHaWdtjna/rzGs jqkT27GxKB
DLT0Fr3C/TzzHvBRaAg==" crossorigin="anonymous" referrerpolicy="no-
referrer"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/3.8.0/helpers.esm.min.js"
integrity="sha512-
vxCPccgWacJoW2HlxhlKKtczdzvcg0r1UuB9LfNGt6vsDbgLfSFxKlOlUS2mqKNXrOK5b9
3S45309T+V5BhueA==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>

<script> src=
"https://cdn.jsdelivr.net/combine/npm/chart.js@3.8.0/dist/chart.esm.min.js,npm/chart.js@3.8.
0/dist/chart.min.js,npm/chart.js@3.8.0,npm/chart.js@3.8.0/dist/helpers.esm.min.js"</script>

<script> src= "https://cdn.jsdelivr.net/npm/chart.js@3.8.0/dist/chart.esm.min.js"</script>
<script> src= "https://cdn.jsdelivr.net/npm/chart.js@3.8.0/dist/chart.min.js"</script>
<script> src= "https://cdn.jsdelivr.net/npm/chart.js@3.8.0/dist/chart.min.js"</script>
<script>src= "https://cdn.jsdelivr.net/npm/chart.js@3.8.0/dist/helpers.esm.min.js"</script>

<div class="page-header"><h1>Depósitos</h1></div><!-- /.page-header -->

<!-- qui va el contenido de la tabla -->

<div class="row" >

  <div class="col-xs-3">
```

```
</div>
```

```
<div class="col-xs-5" >
```

```
<div>
```

```
<canvas id="myChart"></canvas>
```

```
</div>
```

```
<script>
```

```
var depositos=@json($montodpf);
```

```
const labels = ['Hombres','Mujeres'];
```

```
const data = {
```

```
  labels: labels,
```

```
  datasets: [{
```

```
    label: 'Usuarios Activos',
```

```
    backgroundColor: ['#bdd4ed','#207bd8'],
```

```
    borderColor: 'white',
```

```
    data: [depositos['hombres'],depositos['mujeres']],
```

```
  ]
```

```
};
```

```
const config = {
```

```
  type: 'pie',
```

```
  data: data,
```

```
};
```

```
</script>
```

```
<script>
```

```
const myChart = new Chart(  
  document.getElementById('myChart'),  
  config  
);
```

```
</script>
```

```
</div><!-- /.col -->
```

```
<div class="col">
```

```
</div>
```

```
</div><!-- /.row -->
```

```
@endsection
```


4.5 PRUEBAS

Fig. No 19.

Pruebas Seguridades 1.

| | | | |
|--|--|---------------------------|------------|
| 
WEBCOOPEC SYSTEM
SERVICIOS CORPORATIVOS WEB | | HOJA | 1 DE 5 |
| | | PROCESO-PRUEBAS | QA |
| | | VERSIÓN | 1.0 |
| | | FECHA | JUNIO 2022 |
| Formato FO-TE-PRU-01 Plan de Pruebas | | Guía de Interoperabilidad | |

| INFORMACIÓN GLOBAL DEL CASOS DE PRUEBAS SEGURIDADES | | | | | | |
|--|--|--|--|----------|----|---|
| CASO DE PRUEBA No. | 001-PRUEBAS DE SEGURIDADES | VERSIÓN DE EJECUCIÓN | Versión 1.0 | | | |
| | | FECHA EJECUCIÓN | 2022-06-03 | | | |
| CASO DE USO: | INGRESO DE USUARIOS | SISTEMA AIBASOL | SEGURIDADES | | | |
| Descripción del caso de prueba: | Probar el ingreso de usuarios, se pretende probar el uso y el correcto funcionamiento de las opciones del proceso de seguridades del aplicativo. | | | | | |
| 1. CASOS DE PRUEBA | | | | | | |
| a. Precondiciones | | | | | | |
| Internet necesario para el logueo.
Usuario y clave de acceso.
Correo electrónico para verificar el sistema de mensajería | | | | | | |
| b. Pasos de la prueba | | | | | | |
| Con el usuario asignado loguear al sistema http://aibasoc.webcoopec.com
Proceso de recuperar clave
Crear un usuario
Asignar estado | | | | | | |
| DATOS DE ENTRADA | | | RESPUESTA ESPERADA DE LA APLICACIÓN | FUNCIONO | | RESPUESTA DEL SISTEMA |
| CAMPO | VALOR | TIPO ESCENARIO | | SI | NO | |
| Link de acceso | URL | Despliega pantalla de AIBASOL | Presentación de la pantalla principal del Aplicativo | X | | Fondo de pantalla de Aibasol |
| Botón de ingreso | ON | Despliega pantalla de ingreso al sistema | Presentación de la pantalla de usuario y claves | X | | Pantalla de Ingreso al sistema |
| Olvido contraseña | ON | opción de Restablecer contraseña | Formulario de restablecer contraseña | X | | Envío de correo con nueva clave |
| Logueo | ON | Menú principal | Despliega el menú lateral la opción de usuarios | X | | Se puede navegar por las opciones de menú |
| Búsquedas | ON | Realiza las búsquedas | Despliega el registro según el criterio de la búsqueda | X | | Realiza la búsqueda de todos los procesos |
| Crear Usuario | Datos | Crear usuario | Datos almacenados correctamente | X | | Se despliega el nuevo registro |
| Encriptación | ON | Genera encriptación | Despliega datos encriptados | | X | Se encripta a nivel de base de datos |

Fig. No 20.

Pruebas Seguridades 2.

| | | | | | | |
|--|--|--|--|---------------------------|--|------------|
| 
WEBCOOPEC SYSTEM
SERVICIOS CORPORATIVOS WEB | | | | HOJA | | 2 DE 5 |
| | | | | PROCESO-PRUEBAS | | QA |
| | | | | VERSIÓN | | 1.0 |
| | | | | FECHA | | JUNIO 2022 |
| Formato FO-TE-PRU-01 Plan de Pruebas | | | | Guía de Interoperabilidad | | |

| | | | | | | |
|-----------------------------|-------|----------------------------------|---------------------------------|---|--|---------------------------------------|
| Cambio de Estado de usuario | Botón | Activar o desactivar | Prende o apaga el icono off on | X | | Se almacena la fecha de inactivación |
| Editar usuario | Datos | Modifica información del usuario | Se editan los campos de usuario | X | | Se actualizan solo campos permitidos. |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

c. Post condiciones

El acceso es lento por causa del internet
Los correos no llegan, se utiliza mailtrap, para realizar pruebas de envío.

2. RESULTADOS DE LA PRUEBA

| Defectos y desviaciones | Veredicto |
|--|---|
| <i>El aplicativo y sus procesos de seguridad y logueo esta funcional y debe ser liberado</i> | Paso ☐
Falló ☐ |

| Observaciones | Probador |
|---|---|
| <i>El aplicativo esta funcional en los procesos de logueo y seguridades</i> | 
Firma:
Nombre: Santiago Guayasamin Díaz
Fecha: 3 - junio - 2022 |

Fig. No 21.

Pruebas Indicadores 1.

| | | |
|---|-----------------|---------------------------|
|  | HOJA | 3 DE 5 |
| | PROCESO-PRUEBAS | QA |
| | VERSION | 1.0 |
| FECHA | | JUNIO 2022 |
| Formato FO-TE-PRU-01 Plan de Pruebas | | Guía de Interoperabilidad |

| INFORMACIÓN GLOBAL DE CASOS DE PRUEBAS INDICADORES | | | | | | |
|--|---|--|--|----------|----|--|
| CASO DE PRUEBA No. | 001-PRUEBAS DE INDICADORES | VERSIÓN DE EJECUCIÓN | Versión 1.0 | | | |
| | | FECHA EJECUCIÓN | 2022-06-07 | | | |
| CASO DE USO: | GENERACIÓN DE INDICADORES | SISTEMA AIBASOL | INDICADORES | | | |
| Descripción del caso de prueba: | Probar la generación de los indicadores financieros de balance social, solicitados por la compañía, así como la generación de dashboard de los mismos | | | | | |
| 3. CASOS DE PRUEBA | | | | | | |
| d. Precondiciones | | | | | | |
| Despliegue del menú de opciones.
Selección del periodo mensual.
Selección de indicadores, calculo y presentación de depósitos por genero
Selección de indicadores, calculo y presentación de créditos por genero
Selección de indicadores, calculo y presentación de morosidad por genero | | | | | | |
| e. Pasos de la prueba | | | | | | |
| Con el usuario asignado loguear al sistema http://aibasoc.webcopec.com
Selección del periodo mensual a reportar
Verificar si la conexión a la base de datos del CORE financiera es correcta
Verificar si los indicadores están correctos en sus cálculos
Generar grafico de Iso indicadores generales. | | | | | | |
| DATOS DE ENTRADA | | | RESPUESTA ESPERADA DE LA APLICACIÓN | FUNCIONO | | RESPUESTA DEL SISTEMA |
| CAMPO | VALOR | TIPO ESCENARIO | | SI | NO | |
| Link de acceso | URL | Despliega pantalla de AIBASOC | Presentación de la pantalla principal del Aplicativo | X | | Fondo de pantalla de Aibasoc |
| Botón de ingreso | ON | Despliega pantalla de ingreso al sistema | Presentación de la pantalla de usuario y claves | X | | Pantalla de Ingreso al sistema |
| Logueo | ON | Menú principal | Despliega el menú lateral la opción de usuarios | X | | Se puede navegar por las opciones de menú |
| Seleccionar el periodo | Listado | Lista las fechas almacenadas | Despliega el registro de fechas generadas en el aplicativo | X | | Realiza la búsqueda de información del periodo |
| Crear Periodo | Datos | Crear Periodo | Presenta en el formulario la fecha de almacenamiento | X | | Se despliega el nuevo registro |

Fig. No 22.

Pruebas Seguridades 2.

| | | | | | | |
|--|--|--|--|---------------------------|--|------------|
| 
WEBCOOPEC SYSTEM
SERVICIOS CORPORATIVOS WEB | | | | HOJA | | 4 DE 5 |
| | | | | PROCESO-PRUEBAS | | QA |
| | | | | VERSION | | 1.0 |
| | | | | FECHA | | JUNIO 2022 |
| Formato FO-TE-PRU-01 Plan de Pruebas | | | | Guía de Interoperabilidad | | |

| | | | | | | |
|---------------------------------------|------|-----------------------------------|---|--|---|---|
| Selección de indicador depósitos | Link | Genera formularios de indicadores | Despliega datos del indicador de depósitos por genero. | | X | Se calcula y presenta información del indicador |
| Selección de indicador crédito | Link | Genera formularios de indicadores | Despliega datos del indicador de créditos por genero. | | X | Se calcula y presenta información del indicador |
| Selección de indicador de morosidad | Link | Genera formularios de indicadores | Despliega datos del indicador de morosidad de cartera por genero. | | X | Se calcula y presenta información del indicador |
| Selección de Dashboard de indicadores | Link | Genera gráfico indicadores | Despliega datos del indicador de forma gráfica de los indicadores | | X | Se calcula y presenta información gráfica del indicador |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

f. Post condiciones

*El acceso esta correctos y permite generar los indicadores
Los gráficos se están generando mediante las librerías chart.js, y se despliegan muy grandes.*

4. RESULTADOS DE LA PRUEBA

| Defectos y desviaciones | Veredicto |
|---|---|
| <i>El aplicativo y sus procesos de cálculo y generación de indicadores esta funcional y debe ser liberado</i> | Paso ☐
Falló ☐ |

| Observaciones | Probador |
|---|--|
| <i>El aplicativo esta funcional en los procesos de cálculo y presentación de información en montos, porcentajes, el aplicativo permite desplegar los gráficos de los indicadores.</i> | 
Firma:
Nombre: Santiago Guayasamín Díaz
Fecha: 7 - junio - 2022 |

4.6 IMPLEMENTACIÓN

La compañía utiliza la herramienta JIRA para el control y ejecución del proyecto, para ello se crea un proyecto denominado Balance Social.

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

Se definen las particularidades y recomendaciones que se presentaron en el proceso de desarrollo del aplicativo, sobre todo por la metodología utilizada debido a que los periodos determinados por la Superintendencia de Economía Popular y Solidaria son muy cortos y la compañía necesita implementar el aplicativo de manera urgente en sus clientes, con el propósito de evitar multas pecuniarias por el incumplimiento de la normativa vigente.

5.1 Conclusiones

- La implementación de una técnica Ágil scrum, que estipula como principio el empirismo y sobrepone la funcionalidad a la documentación exhaustiva se logro concluir con el desarrollo del aplicativo web en los tiempos establecidos por el organismo de control.
- Las herramientas del framework Laravel utilizadas en el desarrollo del sistema permitieron que el aplicativo sea de características responsivas, con lo cual puede ser utilizado a través de dispositivos móviles o tabletas.
- El aplicativo tiene la capacidad de almacenar e integrarse a la base de datos del CORE financiero Webcoop de propiedad de la compañía, para obtener los indicadores solicitados en las notas técnicas publicadas por el organismo de control en su primera fase.

- Es importante mencionar que el aplicativo denominado AIBASOL V1.0 se considera una solución para los clientes de la compañía Webcopec System, debido a que permite la obtención y generación de los indicadores solicitados por la Superintendencia, mediante una integración directa con el registro histórico de la base de datos del CORE financiero, la misma que permitirá obtener la información que debe ser publicada por las Cooperativas de Ahorro y Crédito del Ecuador.
- Por la misma tecnología se pudo crear de manera ágil los denominados dashboard requeridos por la compañía, los cuales brindan una adecuada presentación de los indicadores calculados por el aplicativo.

5.2 Recomendaciones

- Es importante mencionar que el aplicativo se desarrolló con el objetivo de integrarse al CORE financiero de la compañía, sin embargo la herramienta por su tecnología puede integrarse mediante webservices a cualquier CORE financiero, logrando obtener y publicar los indicadores de cooperativas que no sean clientes de la compañía, motivo por el cual recomendamos trabajar en el webservice para la siguiente versión del aplicativo.
- Se recomienda realizar una adecuada capacitación de las técnicas y artilugios empleadas en el desarrollo, debido a que el aplicativo se encuentra ajustado a los indicadores solicitados en la normativa en su primera fase, y la superintendencia puede a cualquier momento publicar la fecha de cumplimiento de la segunda fase de los indicadores de balance social.
- Es importante considerar la integración del aplicativo con los CORE financieros desde un ambiente CLOUD, para ello la compañía tiene infraestructura IaaS de AWS, con la cual se puede realizar la implementación de un solo aplicativo que maneje una gama de conexiones para los clientes de la compañía, claro que es necesario trabajar en los temas de confidencialidad del uso de la información.
- Se recomienda implementar una adecuada capacitación del uso del aplicativo, para que la interpretación de los indicadores sea fácil.

Referencias

- Altamirano, R. (28 de octubre de 2021). *¿Qué es una cooperativa de ahorro y crédito? ¿Qué es una cooperativa de ahorro y crédito?:* <https://rebajatuscuentas.com/pe/blog/que-es-una-cooperativa-de-ahorro-y-credito>
- Arteaga, I., & Mouriño, G. (16 de octubre de 2018). *Metodologías ágiles ¿qué son y para qué sirven?* Technology Consulting:
<https://www.tithink.com/es/2018/10/16/metodologias-agiles-que-son-y-para-que-sirven/>
- Chavez, J. (2019). *Implementación de una aplicación web para optimizar la gestión de la óptica Chavez, Lima – 2018*. Universidad Nacional Daniel Alcides Carrión.
- Fuentes, M. (15 de enero de 2020). *¿Qué es la metodología ágil?* Obtenido de *¿Qué es la metodología ágil?:* <https://www.redhat.com/es/devops/what-is-agile-methodology>
- Gómez, R. (11 de noviembre de 2015). *¿Qué es MVC?* Web developer:
<http://rodrigogr.com/blog/modelo-vista-controlador/>
- Holtzhausen, N., & Klerk, J. (2018). Servant leadership and the Scrum team's effectiveness. *Leadership & Organization Development Journal*, 39(7), 873-882.
doi:<https://doi.org/10.1108/LODJ-05-2018-0193>
- Instituto Nacional de Economía Popular y Solidaria-IEPS. (2011). *Ley de la Economía Popular y Solidaria*. Instituto Nacional de Economía Popular y Solidaria-IEPS.
- Laravel. (15 de septiembre de 2018). *Laravel*. Laravel:
<https://desarrolloweb.com/home/laravel#track33>
- Ledesma, E. (03 de septiembre de 2020). *Proyectum*. SCRUM: Cómo escribir historias de usuarios sin morir en el intento: <https://www.proyectum.com/sistema/blog/scrum-como-escribir-historias-de-usuarios-sin-morir-en-el-intento/>

- Ortega, J. (21 de octubre de 2018). *¿Qué es Entorno de desarrollo?* ¿Qué es Entorno de desarrollo?: <https://www.arimetrics.com/glosario-digital/entorno-de-desarrollo>
- Pries-Heje, J., & Baskerville, R. (2017). The translation and adaptation of agile methods: a discourse of fragmentation and articulation. *Information Technology & People*, 30(2), 396-423. <https://doi.org/10.1108/ITP-08-2013-0151>
- Red de Integración Ecuatoriana de Cooperativas de Ahorro y Crédito. (2016). *BSC Balance Social Cooperativo*. Red de Integración Ecuatoriana de Cooperativas de Ahorro y Crédito.
- Secretariado de ICURN c/o Consejo Mundial de Cooperativas de Ahorro y Crédito. (2013). *Principios directrices para mejorar la gobernabilidad de las instituciones cooperativas financieras*. Red Internacional de Reguladores de Cooperativas de Ahorro y Crédito.
- Stormi, K., & Laine, T. (2019). Agile performance measurement system development: an answer to the need for adaptability? *Journal of Accounting & Organizational Change*, 15(2), 231-256. doi:<https://doi.org/10.1108/JAOC-09-2017-0076>
- Super Intendencia de Economía Popular y Solidaria. (22 de junio de 2022). *Super Intendencia de Economía Popular y Solidaria*. Estudios sobre EPS: <https://www.seps.gob.ec/>
- Visual Studio Code. (11 de abril de 2022). *Visual Studio Code*. Visual Studio Code in Action: <https://code.visualstudio.com/docs>

Anexos