

UNIVERSIDAD TÉCNICA DEL NORTE



Facultad De Ingeniería en Ciencias Aplicadas
Carrera de Ingeniería en Sistemas Computacionales

DESARROLLO DE UN SISTEMA WEB PARA LA GESTIÓN Y MONITOREO DE LOS PROCESOS JUDICIALES Y DE RECURSOS HUMANOS, CON INFORMES ESTADÍSTICOS DE CRÉDITOS, AGENCIAS Y DESEMPEÑO SOCIAL PARA LA COOPERATIVA DE AHORRO Y CRÉDITO MUJERES UNIDAS MEDIANTE LA UTILIZACIÓN DEL FRAMEWORK RUBY ON RAILS Y LA LIBRERÍA REACTJS.

Trabajo de grado presentado ante la Ilustre Universidad Técnica del Norte previo a la obtención del título de Ingeniero en Sistemas Computacionales

Autor:
Santiago Javier López Carranco

Directora:
Msc. Daisy Elizabeth Imbaquingo Esparza

Ibarra, 2018



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA

UNIVERSIDAD TÉCNICA DEL NORTE

1.- IDENTIFICACION DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO	
CÉDULA DE IDENTIDAD:	1004015531
APELLIDOS Y NOMBRES:	López Carranco Santiago Javier
DIRECCION:	Av. Retorno
EMAIL:	santy_lopez@live.com.ar
TELÉFONO MÓVIL:	0988848884

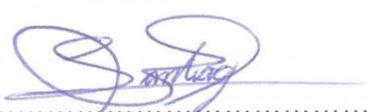
DATOS DE LA OBRA	
TÍTULO:	DESARROLLO DE UN SISTEMA WEB PARA LA GESTIÓN Y MONITOREO DE LOS PROCESOS JUDICIALES Y DE RECURSOS HUMANOS, CON INFORMES ESTADÍSTICOS DE CRÉDITOS, AGENCIAS Y DESEMPEÑO SOCIAL PARA LA COOPERATIVA DE AHORRO Y CRÉDITO MUJERES UNIDAS MEDIANTE LA UTILIZACIÓN DEL FRAMEWORK RUBY ON RAILS Y LA LIBRERÍA REACTJS
AUTOR(ES):	LÓPEZ CARRANCO SANTIAGO JAVIER
FECHA:	16/05/2018
PROGRAMA:	PREGRADO

TITULO POR EL QUE OPTA:	INGENIERO EN SISTEMAS COMPUTACIONALES
ASESOR / DIRECTOR:	MSC. DAISY IMBAQUINGO

2 - CONSTANCIAS

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 16 días del mes de mayo de 2018

EL AUTOR:

.....
Nombre: Santiago Javier López Carranco



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN

En mi calidad de Asesora de Trabajo de Grado presentado por el egresado LÓPEZ Carranco Santiago Javier, para optar por el título de INGENIERO EN SISTEMAS COMPUTACIONALES. Cuyo tema es **“DESARROLLO DE UN SISTEMA WEB PARA LA GESTIÓN Y MONITOREO DE LOS PROCESOS JUDICIALES Y DE RECURSOS HUMANOS, CON INFORMES ESTADÍSTICOS DE CRÉDITOS, AGENCIAS Y DESEMPEÑO SOCIAL PARA LA COOPERATIVA DE AHORRO Y CRÉDITO MUJERES UNIDAS MEDIANTE LA UTILIZACIÓN DEL FRAMEWORK RUBY ON RAILS Y LA LIBRERÍA REACTJS”**. Considero que el trabajo reúne los requisitos y méritos satisfactorios para ser sometido a la defensa pública y evaluación por parte del tribunal examinador.

Msc. Daísy Imbaquingo
DIRECTORA DE TESIS

DEDICATORIAS

Quiero dedicar este trabajo de grado en primer lugar a **DIOS**, por ser él quien me ha impulsado a seguir adelante a lo largo de mi vida universitaria, me ha dado fuerza para afrontar los retos que cada día se me han presentado, él ha sido mi fortaleza en los momentos de tribulación.

Dedico no solo este proyecto, sino cada uno de los días de trabajo, las largas noches de sacrificio, mis momentos de éxito y mis fracasos, mis aciertos y desaciertos a mis **PADRES** por ser un ejemplo en mí, inculcarme sus valores e ideales que me acompañan cada día, y por el sacrificio que hicieron para que pueda culminar mis estudios.

Quiero dedicar este trabajo a todas aquellas personas que durante este largo periodo y a pesar de todas las dificultades han confiado siempre en mí, a mis hermanos, y hermana Sofia, este es un logro más de los muchos que hemos conseguido y se los dedico.

Santy.

AGRADECIMIENTOS

Agradezco especialmente **DIOS** por sus bendiciones hacía mi y a mi familia para poder culminar esta vida universitaria.

Agradezco de manera especial a la **PhD. Nataly Cadena Palacios** por su incalculable apoyo durante todo el proceso de elaboración del presente proyecto, por guiarme con sus grandes conocimientos en la materia y ayudar a la culminación satisfactoria del presente.

Agradezco de manera muy especial a mi directora de Tesis, **Msc. Daisy Imbaquingo** por guiarme en todo el proceso de elaboración del presente proyecto, por sus consejos y enseñanzas.

Agradezco a mi **PADRE**, a mi **MADRE**, a toda mi **FAMILIA** y **AMIGOS** por darme el apoyo incondicional durante mis estudios por sus consejos y apoyo en los momentos difíciles.

Agradezco a la Universidad Técnica del Norte, a la carrera de Ingeniería en Sistemas Computacionales por darme la oportunidad de realizar mis estudios y agradezco a todos mis profesores por las enseñanzas y experiencias transmitidas que me ayudaron a crecer profesionalmente.

Gracias a todos ustedes.

INDICE DE CONTENIDO

AUTORIZACIÓN DE USO Y PUBLICACIÓN	ii
CESIÓN DE DERECHOS DE AUTOR	¡Error! Marcador no definido.
CERTIFICACIÓN	iv
DEDICATORIAS.....	v
AGRADECIMIENTOS	vi
INDICE DE CONTENIDO	vii
ÍNDICE DE FIGURAS	ix
INDICE DE TABLAS	xii
RESUMEN	xiv
ABSTRACT	xv
INTRODUCCIÓN	1
Antecedentes	1
Situación Actual.....	2
Definición del Problema.....	2
Objetivos.....	3
Alcance y Limitaciones	3
Justificación	5
Contexto.....	5
CAPITULO 1.....	6
1.1. Metodología Scrum	6
1.2. Herramienta de Desarrollo RubyMine	10
1.3. Patrón de diseño MVC	11
1.4. Lenguajes de Programación	13
1.5. Stack de Desarrollo	15
1.6. Lenguaje de Mercado Haml	26
1.7. Estilos CSS.....	27
1.8. Servidor NGINX.....	29
CAPITULO 2.....	34
2.1. Planificación.....	34
2.2. Sprint 1.....	39
2.3. Sprint 2.....	69

2.4. Sprint 3.....	90
2.5. Sprint 4.....	98
2.6. Sprint 5.....	103
CAPITULO 3.....	117
3.1. Funcionalidad.....	117
3.2. Satisfacción.....	118
3.3. Aceptación.....	119
3.4. Rendimiento	121
CAPITULO 4.....	125
4.1. Conclusiones	125
4.2. Recomendaciones	127
GLOSARIO	128
REFERENCIAS	130
ANEXOS	131

ÍNDICE DE FIGURAS

Fig. 1 Proceso de la metodología Scrum	9
Fig. 2: Proceso de solicitudes de usuario con la arquitectura MVC	12
Fig. 3. Modelo de referencia que representa la arquitectura del framework	18
Fig. 4 Arquitectura de Rails en la aplicación	20
Fig. 5. Búsquedas Web por frameworks y librerías Javascript	23
Fig. 6 Arquitectura de React en la aplicación.....	24
Fig. 7 PostgreSQL y Active Record.....	26
Fig. 8. Ejemplo del marcado HAML.....	26
Fig. 9 Transpilación de Sass a Css	28
Fig. 10. Ejemplo de materiales y elevaciones de material design.....	29
Fig. 12. NGINX como un servidor proxy.....	31
Fig. 13. NGINX como balanceador de carga y terminación SSL	32
Fig. 14. Proceso Judicial para socios que poseen bienes.....	40
Fig. 15. Proceso Judicial para socios que no poseen bienes e insolvencia	40
Fig. 16. Sistema de semaforización de la herramienta.....	41
Fig. 17. Diagrama de la base de datos del módulo de cobranzas judiciales	53
Fig. 18. Login del aplicativo	56
Fig. 19. Dashboard de la aplicación	56
Fig. 20. Listado de juicios asignados al usuario.....	57
Fig. 21. Gestión de estado de juicios	57
Fig. 22. Datos de un crédito judicial	58
Fig. 23. Evaluación de resultados	58
Fig. 24. Juicios Pendientes de ingreso.....	59
Fig. 25. Ingreso de nuevos juicios	59
Fig. 26. Ingreso de nuevos juicios	60
Fig. 27. Formulario de consulta de monitoreos.....	60
Fig. 28. Reporte de monitoreo de juicios.....	61
Fig. 29. Reporte de monitoreo de juicios.....	61
Fig. 30. Reporte de monitoreo de juicios.....	62
Fig. 31. Pantalla principal Jefatura de Crédito	62
Fig. 32. Gestión de Actividades y Etapas	63
Fig. 33. Gestión de Actividades y Etapas	63
Fig. 34. Guardar monitoreos mensuales y previsualización de monitoreos.....	64
Fig. 35. Evaluación de resultados usuario de Jefatura de Créditos	64

Fig. 36. Evaluación de resultados usuario de Jefatura de Créditos	65
Fig. 37. Evaluación de resultados usuario de Jefatura de Créditos	65
Fig. 38. Gestión de nuevos juicios.....	66
Fig. 39. Autorización y asignación de nuevos juicios.....	66
Fig. 40. Reporte de monitoreos del usuario de gerencias	67
Fig. 41. Eliminación de monitoreo	67
Fig. 42. Porcentajes de colocación por semana	70
Fig. 43. Pantalla principal módulo de créditos	79
Fig. 44. Menú lateral módulo de créditos	79
Fig. 45. Formulario de consulta de la planificación de la cartera	80
Fig. 46. Planificación de cartera	80
Fig. 47. Formulario de consulta de la cartera vencida	81
Fig. 48. Créditos vencidos por agencia	81
Fig. 49. Detalles de los créditos vencidos.....	82
Fig. 50. Formulario de consulta de la eficiencia de cartera	82
Fig. 51. Eficiencia de cartera por agencia	83
Fig. 52. Detalles de los créditos de la eficiencia de cartera.....	83
Fig. 53. Formulario de consulta de las matrices de transición.	84
Fig. 54. Matriz de transición de saldos.....	84
Fig. 55. Matriz de transición de cantidades.....	85
Fig. 56. Consulta de detalles de la matriz de transición	85
Fig. 57. Formulario de consulta de cosechas	86
Fig. 58. Cosechas de la cooperativa CACMU.....	86
Fig. 59. Detalles de las cosechas.....	87
Fig. 60. Formularios de consulta de los indicadores de créditos.....	87
Fig. 61. Indicadores de género de créditos vigentes	88
Fig. 62. Indicador de género de créditos colocados	88
Fig. 63. Pantalla principal módulo de agencias.....	93
Fig. 64. Menú lateral módulo de agencias	94
Fig. 65. Reporte de indicadores por agencia	94
Fig. 66. Gráfica del indicador de activos	95
Fig. 67. Reporte de indicadores de la SEPS.....	95
Fig. 68. Gráfica del indicador de solvencia normativa	96
Fig. 69. Pantalla principal módulo de desempeño social.....	100
Fig. 70. Reporte de balance social	101
Fig. 71. Menú lateral módulo de desempeño social	101
Fig. 72. Pantalla principal módulo de recursos humanos	110

Fig. 73. Listado de empleado de la cooperativa	111
Fig. 74. Formulario de ingreso de un nuevo empleado	111
Fig. 75. Calendario global de planificación de vacaciones	112
Fig. 76. Detalles de un empleado.....	112
Fig. 77. Formulario de edición de un empleado.....	113
Fig. 78. Formulario de ingreso de permisos de un empleado	113
Fig. 79. Historial de permisos de un empleado.....	114
Fig. 80. Calendario de planificación de vacaciones de un empleado	114
Fig. 81. Formulario de ingreso de planificación de vacaciones	115
Fig. 82. Gráfico satisfacción del cliente	119
Fig. 83. Gráfico de aceptación del aplicativo	120
Fig. 84. Configuración del JMeter primera parte.....	122
Fig. 85. Configuración del JMeter segunda parte	122
Fig. 86. Configuración de tiempos JMeter primera parte.....	123

INDICE DE TABLAS

TABLA 1: Roles principales de la metodología Scrum	7
TABLA 2 Componentes de MVC	11
TABLA 3: Componentes integrados en Ruby on Rails.	17
TABLA 4 Convenciones de Ruby on Rails.....	19
TABLA 5: Roles de la metodología Scrum para el proyecto.	34
TABLA 6: Formato para elaborar el Product Backlog.....	35
TABLA 7: Product backlog del aplicativo.	35
TABLA 8: Formato para redactar los sprints.....	38
TABLA 9: Recursos y costos para el proyecto.....	39
TABLA 10: Historia de usuario 1	42
TABLA 11: Historia de usuario 2	42
TABLA 12: Historia de usuario 3	43
TABLA 13: Historia de usuario 4	43
TABLA 14: Historia de usuario 5	44
TABLA 15: Historia de usuario 6	44
TABLA 16: Historia de usuario 7	45
TABLA 17: Historia de usuario 8	45
TABLA 18: Historia de usuario 9	46
TABLA 19: Historia de usuario 10	46
TABLA 20: Historia de usuario 11	47
TABLA 21: Historia de usuario 12	47
TABLA 22: Historia de usuario 13	48
TABLA 23: Historia de usuario 14	48
TABLA 24: Historia de usuario 15	49
TABLA 25: Historia de usuario 16	49
TABLA 26: Historia de usuario 17	50
TABLA 27: Historia de usuario 18	50
TABLA 28: Historia de usuario 19	51
TABLA 29: Historia de usuario 20	51
TABLA 30: Historia de usuario 22	52
TABLA 31: Historia de usuario 23	52
TABLA 32: Sprint Backlog final del spint 1	55
TABLA 33: Matriz de transición	71
TABLA 34: Historia de usuario 24	73
TABLA 35: Historia de usuario 25	73
TABLA 36: Historia de usuario 26	74

TABLA 37: Historia de usuario 27	74
TABLA 38: Historia de usuario 28	75
TABLA 39: Historia de usuario 29	75
TABLA 40: Historia de usuario 30	76
TABLA 41: Base de datos de la herramienta.....	76
TABLA 42: Sprint Backlog final del spint 2.....	78
TABLA 43: Historia de usuario 31	91
TABLA 44: Historia de usuario 32	91
TABLA 45: Historia de usuario 33	91
TABLA 46: Sprint Backlog final del sprint 3	93
TABLA 47: Historia de usuario 34	98
TABLA 48: Historia de usuario 35	99
TABLA 49: Sprint Backlog final del sprint 4	100
TABLA 50: Historia de usuario 36	104
TABLA 51: Historia de usuario 37	104
TABLA 52: Historia de usuario 38	105
TABLA 53: Historia de usuario 39	105
TABLA 54: Historia de usuario 40	106
TABLA 55: Historia de usuario 41	106
TABLA 56: Historia de usuario 42	107
TABLA 57: Historia de usuario 43	107
TABLA 58: Historia de usuario 44	108
TABLA 59: Historia de usuario 45	108
TABLA 60: Sprint Backlog final del sprint 5	109
TABLA 61: Evaluación de la funcionalidad del sistema.....	117
TABLA 62: Encuesta de satisfacción al cliente.....	119
TABLA 63: Aceptación del aplicativo	120
TABLA 64: Resultados Apdex de JMeter módulo de créditos y recursos humanos.....	123
TABLA 65: Resultados Apdex de JMeter segunda parte	124

RESUMEN

La Cooperativa de Ahorro y Crédito Mujeres Unidas (CACMU) como parte de su labor diaria, realizar muchos procesos internos para gestionar los créditos en estado de mora, para gestionar su plantilla laboral, para obtener información estadística de su cartera, cartera en riesgo, colocaciones, balances sociales, planificaciones de recuperación de cartera al mes, y obtención de indicadores sociales o indicadores de la SEPS. Estos procesos son realizados por los empleados de manera manual lo que provoca una falta de eficiencia para la obtención de información vital para la toma de decisiones y el normal desempeño de las funciones de la cooperativa por lo que se propone automatizar y mejorar dichos procesos de administración y gestión. La implementación del sistema surge a partir de la necesidad de obtener toda esta información actualizada y de manera inmediata por parte de las gerencias y diferentes departamentos de la cooperativa, para la cual se desarrolló el sistema que cumplen todos los requerimientos y satisface las necesidades de la cooperativa. Las tecnologías utilizadas para el desarrollo de este proyecto están divididas en el front-end y el back-end en una misma aplicación monolítica, se utiliza Ruby en su versión 2.4 como lenguaje de programación del back-end junto con el framework Ruby on Rails en su versión 5.1 para la obtención de información y manejo de datos, para el front-end se utilizó el lenguaje JavaScript en su versión ES6 y React en su versión 16.0 para construir las vistas orientadas a componentes para ser mostradas al usuario final, como gestor de base de datos se utilizó PostgreSQL en su versión 9.6 y para obtener la información de la cooperativa se configuró Ruby on Rails para manejar dos gestores de bases de datos dentro de la aplicación y hacer uso de la información de la base de datos OracleDb propia de la cooperativa. Todo el proceso de desarrollo fue realizado bajo la metodología scrum generando los artefactos necesarios de la metodología, para cada iteración de la metodología que entrega una porción del producto terminado se aplicó un ciclo de desarrollo de software (Planificación, Análisis, Diseño, Codificación y Pruebas correspondientes). Finalmente se presenta la evaluación de resultados, donde se evalúa la funcionalidad, satisfacción, aceptación y rendimiento del sistema junto con conclusiones y recomendaciones.

ABSTRACT

The Cooperativa de Ahorro y Credito Unidas Unidas (CACMU) as part of their daily work, perform many internal processes to manage delinquent loans, to manage their workforce, to obtain statistical information of their portfolio, portfolio at risk, placements, social balances, portfolio recovery plans per month, and obtaining social indicators or SEPS indicators. These processes are carried out by employees in a manual way, which causes a lack of efficiency in obtaining vital information for decision-making and the normal performance of the cooperative's functions, which is why it is proposed to automate and improve these administration and management processes. The implementation of the system arises from the need to obtain all this information updated and immediately by the managements and different departments of the cooperative, for which the system was developed that gratify all the requirements and meets the needs of the cooperative. The technologies used for the development of this project are divided into the front-end and the back-end in the same monolithic application, Ruby is used in version 2.4 as the programming language of the back-end along with the Ruby on Rails framework, its version 5.1 to obtaining information and data management, for the front-end the JavaScript language was used in its version ES6 and React in its version 16.0 to build the component-oriented views to be shown to the end user, as manager of database was used PostgreSQL in its version 9.6 and to obtain the information of the cooperative Ruby on Rails was configured to manage two database managers within the application and to use of the information of the database Oracledb of the cooperative. The entire development process was carried out under the scrum methodology, generating the necessary artifacts of the methodology, for each iteration of the methodology that delivers a portion of the finished product, a software development cycle was applied (Planning, Analysis, Design, Coding and corresponding Testing). Finally, the evaluation of results is presented, where the functionality, satisfaction, acceptance and performance of the system are evaluated together with conclusions and recommendations.

INTRODUCCIÓN

Antecedentes

Antecedentes de la institución

La Cooperativa de Ahorro y Crédito Mujeres Unidas -Tantanakushka Warmikunapak- "CACMU", nace en el año 2000 como caja solidaria, bajo el impulso y apoyo de la Fundación Cooperación y Acción Comunitaria. Se legaliza como cooperativa de ahorro y crédito el 4 de diciembre del 2001 mediante Acuerdo Ministerial N° 137 del MIES para viabilizar el microcrédito de desarrollo y la voluntad de unir los pequeños esfuerzos de las Organizaciones y Grupos Solidarios de las mujeres y sus familias del norte ecuatoriano. Surge para apoyar el trabajo de las organizaciones de sectores rurales y urbanos populares potenciando la capacidad productiva de mujeres microempresarias, mestizas, negras e indígenas y sus familias, mediante la asistencia financiera y no financiera y el fortalecimiento socio-organizativo popular. ("Cooperativa de ahorro y crédito mujeres unidas", 2017).

CACMU es una Cooperativa de Ahorro y Crédito inclusiva comprometida con el mejoramiento de las condiciones de vida de las mujeres y sus familias de los sectores rurales y urbano populares del país, en un futuro se espera consolidarse como una cooperativa reconocida y referente por su gestión financiera con enfoque social y de género, principalmente rural y de fomento a la asociatividad de la economía popular y solidaria.

Antecedentes del Aplicativo

Las aplicaciones web han evolucionado en los últimos 10 años y más aún con la llegada de la web 2.0 y 3.0, en donde se busca soluciones más complejas y robustas a las necesidades tecnológicas de la sociedad actual. Esto significó el aumento de la complejidad de desarrollo de aplicaciones web, por tal motivo empezaron a surgir *frameworks* de desarrollo Web más avanzados que evolucionan constantemente y que integran módulos y herramientas específicas para realizar tareas más complejas de una manera más transparente al programador, facilitando el desarrollo de las aplicaciones web. Entre alguno de estos *frameworks* se encuentra el conocido "Framework de desarrollo ágil de aplicaciones *Ruby on Rails*".

Ruby on Rails (RoR) surgió en el año 2003 cuando se estaba trabajando en el código de una herramienta de gestión de proyectos, fue liberado en el 2004 RoR impuso una manera distinta de desarrollar aplicaciones Web, se volvió tan popular que grandes

empresas mundiales empezaron a usar esta herramienta como: Twitter, Soundcloud, Basecamp, Github, Twitch.

React también conocido como React.js o ReactJS es una biblioteca Javascript de código abierto que sirve especialmente para crear interfaces de usuario orientado a componentes con el objetivo de impulsar el desarrollo de aplicaciones de una sola página (*Single Page Applications*). La librería de React fue desarrollada por la empresa Facebook, la cual, actualmente es contribuido por Instagram y una comunidad de desarrolladores independientes y compañías. React ayuda a los desarrolladores a construir aplicaciones que usan datos que cambian todo el tiempo. Su objetivo es ser sencillo, declarativo y fácil de combinar.

Situación Actual

En la actualidad la Cooperativa de Ahorro y crédito Mujeres Unidas (con sus siglas CACMU) no cuenta con un sistema Web en el cual pueda gestionar y controlar los procesos judiciales de los créditos en estado de mora que ha otorgado a sus socias(os), visualizar indicadores estadísticos de créditos, desempeño social, o agencias, para que el personal administrativo pueda analizar dicha información con la finalidad de elaborar planificaciones, toma de decisiones gerenciales y de negocio financiero.

Hasta ahora todos los procesos antes mencionados y elaboración de indicadores se los viene realizando de manera manual, el control y gestión de créditos en procesos judiciales se almacena en hojas de Excel, para obtener indicadores de la cooperativa se tiene que realizar un proceso manual de obtención de información y pasarla a hojas de cálculo en tablas dinámicas lo que ocasiona que no se tenga indicadores actualizados, en el momento que se los necesita y tenga un costo elevado de tiempo elaborarlos sujeto a errores.

Actualmente llevar a cabo todos estos procesos manualmente tiene un costo alto de tiempo y recursos lo que está impidiendo el logro de los objetivos institucionales de la cooperativa con la sociedad.

Definición del Problema

La Cooperativa de Ahorro y Crédito Mujeres Unidas (CACMU) no cuenta con una manera eficiente y ágil de gestión y monitoreo de los procesos de créditos judiciales y obtención de indicadores claves de manera eficaz de agencias, créditos, desempeño social y recursos humanos.

Objetivos

Objetivo General

Desarrollar un sistema web para la Cooperativa de Ahorro y Crédito Mujeres Unidas en la cual se encuentren automatizados todos los procesos de gestión judicial y obtención de indicadores de gestión mediante la utilización del *Framework* Ruby on Rails y la librería ReactJs con la metodología Scrum.

Objetivos Específicos

- Estudiar el Framework de desarrollo ágil de aplicaciones Web Ruby on Rails para el desarrollo una plataforma Web para CACMU con la finalidad de automatizar los procesos de la cooperativa.
- Estudiar la librería *front-end* ReactJs para implementar vistas orientadas a componentes en el sistema Web a desarrollarse con la finalidad de conseguir un sistema web ágil, escalable con componentes reutilizables.
- Aplicar la metodología Scrum para el desarrollo ágil del sistema web para minimizar riesgos durante el desarrollo del proyecto y aportar calidad al software.
- Aplicar el patrón de diseño MVC para el desarrollo del sistema web con la finalidad de seguir buenas prácticas de desarrollo y obtener un sistema altamente escalable.

Alcance y Limitaciones

El sistema Web se desarrollará bajo el *Framework* Ruby on Rails en su versión 5.1.2 para el *back-end* del sistema, en cuál, se construirá toda la lógica y el flujo de nuestra aplicación. Para el *front-end* se utilizará la librería de JavaScript desarrollada por Facebook ReactJs para crear componentes interactivos y reutilizables para las interfaces de usuario, en los estilos de las vistas se utilizará preprocesadores de CSS (*Sass*) y Bootstrap Framework con sus componentes de *Material Design* y sus métricas desarrolladas por Google, para aportar en mayor grado al diseño responsivo del sistema Web.

El sistema Web que se implementará en CACMU tendrá los siguientes módulos:

- a) **Módulo de Cobranzas Judiciales:** En el cual se podrá controlar y gestionar los procesos judiciales de todos los créditos en estados de mora que tenga la cooperativa.
- b) **Módulo de Créditos:** En el que se visualizará indicadores de créditos vencidos, planificaciones de colocaciones de cartera, matrices de transiciones entre las calificaciones de los créditos, cosechas de los créditos otorgados, eficiencia de la cartera, e indicadores de créditos vigentes y colocados en un rango de fechas.

- c) **Módulo de Agencias:** En el cual se tendrá acceso a los indicadores financieros de las cuentas de la cooperativa y se podrá visualizar los indicadores de la SEPS (Superintendencia de Economía Popular y Solidaria), todos los indicadores estarán disponibles por cada una de las agencias de la cooperativa.
- d) **Módulo de Desempeño Social:** En este módulo visualizará los balances sociales de la cooperativa, extraídos de un rango de fechas, en los cuales se los podrá extraer por cualquier agencia de la cooperativa o un globalizado de toda la institución.
- e) **Módulo de Recursos Humanos:** En el cual se podrá controlar todo el proceso de control y gestión de permisos y vacaciones de la cooperativa, teniendo en cuenta leyes y normativas del estado.

Los indicadores que se obtendrán serán consultas específicas a la base de datos a los que el sistema aplicará fórmulas de análisis para presentarlas al usuario final.

Para el marcado HTML se utilizará el lenguaje HAML, el cual es un lenguaje de marcado ligero con el cual podemos generar HTML a partir de un sencillo DSL, con esto lograremos tener plantillas HTML mucho más ligeras y legibles.

Para la persistencia de datos se utilizará la base de datos *Open Source PostgreSQL* en su versión 9.6 el que estará manejado por el ORM de Ruby on Rails denominado *Active Record*; para obtener la información de créditos y cuentas de la Cooperativa, se consumirá la información de una base de datos Oracle que actualmente tiene la cooperativa.

El patrón de diseño embebido en el *Framework Ruby on Rails* con el cual se desarrollará el sistema es MVC (Modelo – Vista - Controlador). En nuestra aplicación la Vista estará a cargo de *Action View*, para la lógica de controladores se usará *Action Controller* y para los modelos se hará uso del módulo *Action Model* y el ORM *Active Record*, todos estos módulos están embebidos en el *framework* a utilizar.

El sistema Web a implementarse no reemplazará al sistema actual de la Cooperativa, este desarrollo tiene la finalidad de acoplarse a dicho sistema para funcionar en conjunto y solventar las necesidades del personal, estará únicamente al alcance de la intranet de la cooperativa y será fácilmente accedido por el resto de las sucursales mediante VPN.

Justificación

Impacto Tecnológico

El sistema web aportará a que la Cooperativa no tenga que depender de proveedores externos para la obtención de indicadores y contribuirá a continuar con el proceso de automatización tecnológica de la cooperativa para brindar servicios de calidad a sus socias(os), basándose en recursos tecnológicos para realizar labores usuales de la cooperativa de manera eficaz y eficiente.

Impacto Social

El sistema aportará al cumplimiento de los objetivos sociales de la empresa en enfocarse en los grupos más excluidos financieramente al obtener indicadores clave para que la cooperativa pueda tomar decisiones a corto plazo.

Impacto Económico

El desarrollo de este proyecto contribuirá a disminuir costos que incurre la cooperativa con proveedores externos para la obtención de información, y aumentará la recuperación de cartera gracias a un seguimiento oportuno basado en la información que el sistema entregará, a su vez eliminará costos de licencias al ser desarrollado en herramientas de código abierto.

Contexto

El desarrollo de este proyecto es completamente nuevo, CACMU actualmente no cuenta con un sistema que automatice estos procesos, teniendo en cuenta que el desarrollo de esta herramienta estará implementada a la medida de las necesidades y procesos de la cooperativa con la característica que se lo desarrollará bajo las herramientas actuales para un desarrollo Web de calidad.

CAPITULO 1

Marco Teórico

1.1. Metodología Scrum

En la actualidad el desarrollo de software ha ido cambiando y adaptándose a las nuevas necesidades de los clientes, ahora se busca estrategias para el lanzamiento de productos de software de manera temprana con resultados plenamente independientes y funcionales, que tengan una respuesta ágil e inmediata a los posibles cambios o requisitos que puedan surgir en el desarrollo del software, por esta razón es que el uso de metodologías ágiles de desarrollo de software se ha popularizado.

Scrum es un modelo de trabajo ágil que se caracteriza por optar por un desarrollo incremental en lugar de una planificación completa y secuencial de un producto, dando flexibilidad al desarrollo y permitiendo incrementar valor al producto en cada incremento(sprint) de la metodología, por tal motivo, Scrum es un modelo en donde se va construyendo un producto de software al mismo tiempo que se puede introducir o modificar los requisitos del producto (Menzinsky, López, & Palacio, 2016).

En esta metodología se parte con una visión general de todo el producto de software que se desea desarrollar, y a partir de esta visión general se detallan y especifican todas las funcionalidades que se desean obtener en primer lugar para así tener una lista priorizada de requerimientos del software. (Schwaber & Sutherland, 2017).

Cada iteración o ciclo de desarrollo también conocido como sprint, comienza con la recolección y especificación de los requerimientos a desarrollarse en dicho sprint y finaliza con la entrega de una parte operativa del producto que se puede usar, inspeccionar y evaluar, la duración de cada sprint se recomienda que no se exceda de un mes.

Esta metodología es ideal cuando el cliente no tiene claro los requerimientos del software, lo que puede provocar cambios en los mismos durante el desarrollo del sistema y muchas veces requerirá que se realice cambios radicales en partes ya entregadas y funcionalidades del sistema para que este se acople perfectamente a las necesidades del cliente, scrum nos da la esta flexibilidad de agregar o cambiar requerimientos mientras se está desarrollando el proyecto, en CACMU se tiene claro únicamente determinados módulos del sistema por lo que

se usará scrum para minimizar los riesgos y garantizar la satisfacción del cliente en la entrega del producto final.

Roles de la metodología Scrum

Todas las personas que tienen relación directa o indirecta con el proyecto se clasifican en dos grupos: comprometidos e implicados. (Menzinsky, López, 2016).

TABLA 1: Roles principales de la metodología Scrum

COMPROMETIDOS	IMPLICADOS
Propietario del producto	Otros interesados (dirección, gerencias, comerciales, marketing, etc.)
Miembros del equipo	

Propietario del Producto

El propietario del producto también conocido como **“product owner”** es la persona quien toma las decisiones del cliente y es el responsable de lograr el mayor valor de producto para los clientes, usuarios e implicados.(Menzinsky et al., 2016).

Equipo de Desarrollo

El equipo de desarrollo está formado por el grupo de profesionales que se encargan de realizar el incremento de cada sprint, son un equipo de trabajo multifuncional que tienen responsabilidad compartida, ya que, la responsabilidad completa del incremento de cada sprint recae sobre el equipo de desarrollo en conjunto.

Menzinsky, López, (2016) afirman que en el equipo todos los integrantes deben:

- Conocer la metodología Scrum
- Respetar las opiniones y aportes de los demás
- Comprenden la visión del propietario del producto
- Comparten de forma conjunta el objetivo de cada sprint

Scrum Master

El Scrum Master es la persona que es responsable del cumplimiento de las reglas dentro del marco de trabajo de Scrum, proporciona asesoría necesaria al equipo y propietario del producto, a su vez se encarga de revisar y validar la pila del producto y actúa como moderador en las reuniones. (Menzinsky et al., 2016).

Artefactos de la metodología Scrum

Pila del Producto (*Product Backlog*)

La Pila del Producto es la lista ordenada por prioridad de los requisitos que usuario quiere para el producto final, la cual crece y evoluciona durante el desarrollo (Schwaber & Sutherland, 2017).

La pila del producto nunca se da por completada ya que está en constante evolución y actualización, cuando se comienza con el proyecto se inician con los requerimientos más claros para el usuario y se van agregando nuevos requerimientos conforme vaya avanzando el desarrollo. (Menzinsky et al., 2016).

Los requerimientos en la pila del producto estarán ordenados prioritariamente, en donde el propietario del producto (*product owner*) decide cual requerimiento es más importante por el valor que dicho requerimiento aporte al producto o sea de suma importancia para la continuación del desarrollo.

Para establecer prioridades se puede utilizar el método *MoSCow*:

- **M – Must**, se debe completar este requerimiento para finalizar el proyecto.
- **S – Should**, se debe completar por todos los medios, pero el éxito del proyecto no depende de él.
- **C – Could**, su implementación no afecta a la consecución de los objetivos principales
- **W – Would**, se implementará si sobra tiempo.

Pila del Sprint (*Sprint Backlog*)

La pila del sprint es el listado de tareas para construir las historias de usuario que se van a realizar en un sprint, es decir, la pila del sprint descompone las historias de usuario en unidades más pequeñas para poder monitorizar los avances diarios y medir riesgos y problemas en el desarrollo. (Menzinsky, López, 2016).

Incremento

El incremento es aquella parte del producto final producida en un sprint, en donde su principal característica es que es completamente funcional y entregable al cliente una vez que esta sea probada. (Menzinsky, López, 2016).

Eventos de la metodología Scrum

Reunión de Planificación del sprint

En esta reunión se toman como base las prioridades y necesidades de negocio del cliente, y se determinan cuáles y cómo van a ser las funcionalidades que se incorporarán al producto en el siguiente sprint. (Menzinsky et al., 2016).

Scrum Diario

Reunión diaria breve, de no más de 15 minutos, en la que el equipo sincroniza el trabajo y establece el plan para las 24 horas siguientes.

Revisión del sprint

Reunión realizada al final del sprint para comprobar el incremento.

En la figura 1 se aprecia el proceso completo de la metodología Scrum desde la recolección de requerimientos de usuarios finales, clientes o equipos de trabajo, artefactos, y personas implicadas hasta la entrega del producto terminado.

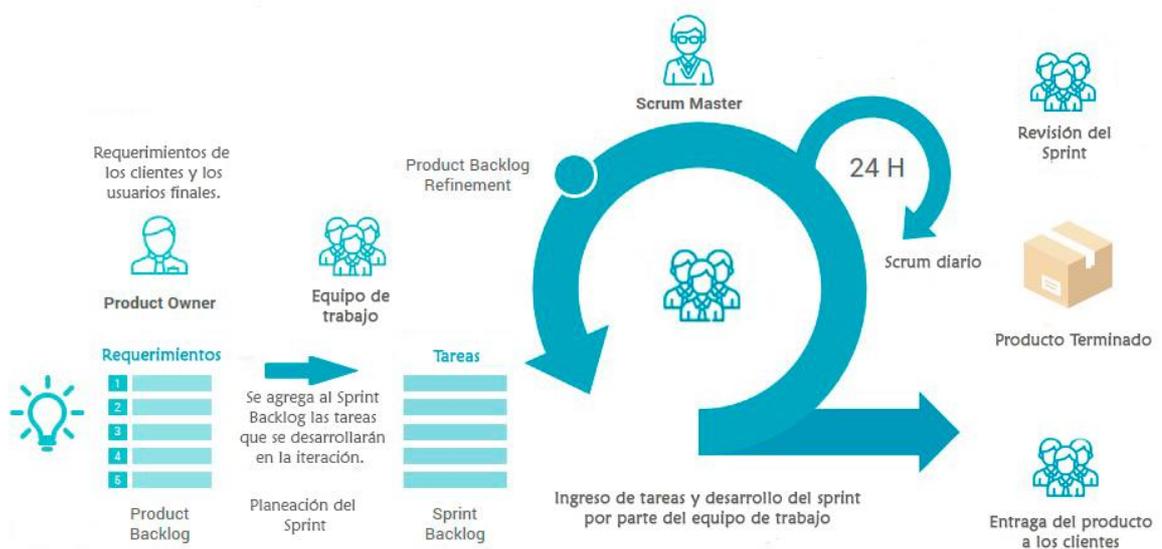


Fig. 1 Proceso de la metodología Scrum

1.2. Herramienta de Desarrollo RubyMine

Definición

Rubymine es un entorno de desarrollo integrado (IDE) inteligente y multiplataforma para todo tipo de proyectos de Ruby y Ruby on Rails con un editor de código inteligente, desarrollado por la empresa JetBrains. IDE ofrece una amplia gama de herramientas esenciales para desarrolladores de Ruby, estrechamente integrado en conjunto para crear un ambiente propicio para el desarrollo productivo de Ruby y desarrollo Web con Ruby on Rails. (“RubyMine”, 2018)

Características

(JetBrains, 2018) empresa creadora de la herramienta afirma que las principales características de RubyMine para el desarrollo de aplicaciones Ruby on Rails son:

- Autocompletado de métodos nativos de Ruby on Rails, métodos definidos por rutas con nombre y rutas de recursos, autocompletado de código asociado a la base de datos y modelos.
- Generadores de Ruby on Rails integrados en el IDE lo que nos permite generar controladores, vistas y modelos desde una interfaz de usuario.
- Es totalmente compatible con los tipos de marcado ERB, HAML y SLIM entregando corrección de sintaxis y terminación de código inteligente para cualquier marcado HTML de Ruby on Rails.
- Permite ver una vista panorámica de los modelos de nuestro proyecto con sus atributos para el análisis de la estructura del proyecto.
- Tiene una consola de Ruby on Rails integrada para ejecutarla directamente desde el IDE, además proporciona un resaltado de sintaxis y un historial de comandos.
- Tiene soporte YAML para los archivos de configuración de bases de datos e internacionalización.
- Lanzador de prueba incorporado, completamente integrado con el IDE, capaz de ejecutar pruebas y suministrar estadísticas sobre la marcha en las pruebas en ejecución.

Ventajas

Utilizar el IDE RubyMine aumenta la productividad en el desarrollo de aplicaciones RoR debido a que ayuda a producir código de alta calidad de manera más eficiente, gracias a la sintaxis y el resaltado de errores, formateo de código, completado de código inteligente, navegación de gran alcance.

La principal ventaja de RubyMine es que maneja un análisis exhaustivo del código del proyecto y ofrece a los desarrolladores la mejor visión de su código con sugerencias de finalización, característica que ningún otro IDE de Ruby y Ruby on Rails consigue, se basa en el conocimiento del *framework* RoR para proporcionar una variedad de refactorizaciones de código que simplifica en gran parte los cambios de código. (Comprasoft, 2018).

1.3. Patrón de diseño MVC

El MVC o Modelo-Vista-Controlador es un patrón de arquitectura de software que, utilizando tres componentes (Vistas, Modelos y Controladores) separa la lógica de la aplicación de la lógica de la vista en una aplicación. La mayoría de los *frameworks* modernos utilizan MVC (o alguna adaptación del MVC) para la arquitectura, entre ellos podemos mencionar a Ruby on Rails, Django, AngularJS y muchos otros más. (Hernandez, 2015a).

El patrón de arquitectura MVC está estructurado en tres componentes principales como se muestra en la tabla 2.

TABLA 2 Componentes de MVC

Componentes de MCV		
Modelo	Vista	Controlador
El modelo es la capa que se encarga de los datos, consultando la Base de datos en la mayoría de las ocasiones y ejecutando código Sql, no obstante, cabe recalcar que en muchas ocasiones se trabaja con librerías como algún PDO o ORM como Doctrine, que mapean la base de datos a objetos.	Son la representación visual de los datos que nos envía el controlador, todo lo que tenga que ver con la interfaz gráfica UI va aquí. Ni el modelo ni el controlador se preocupan de cómo se verán los datos, esa responsabilidad es únicamente de la vista. (Hernandez, 2015).	Esta capa se encarga de controlar, es una capa intermedia entre el modelo y la vista, esta recibe las peticiones del usuario generalmente desde una dirección Url y se encarga de solicitar los datos al modelo y de comunicárselos a la vista.

En el ambiente web, el MVC funciona como se muestra en la figura 2. Cuando el usuario manda una petición al navegador, el controlador responde a la solicitud y pide al modelo la información acerca de este, el modelo, que se encarga de los datos de la aplicación, consulta la base de datos, el modelo responde al controlador con los datos que solicitó, una vez que

el controlador tiene los datos del recurso, se los manda a la vista, organiza la información y construye la página que se ve en el navegador.

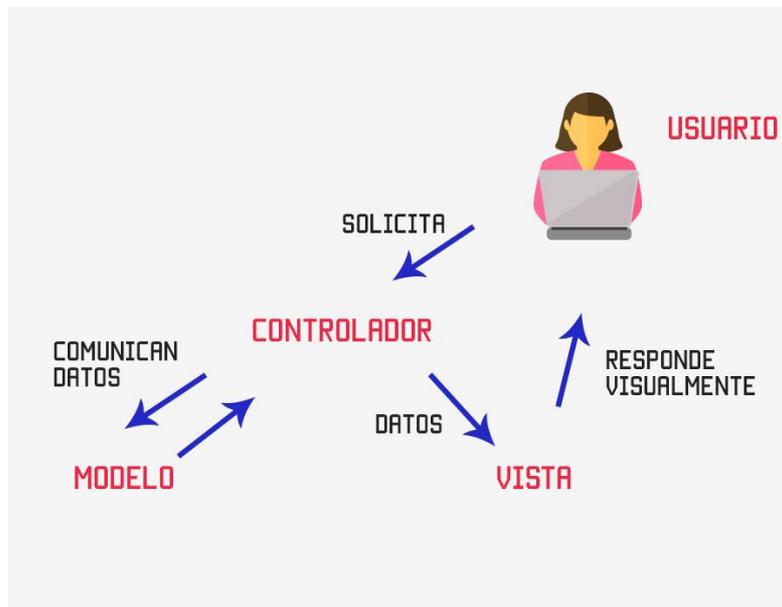


Fig. 2: Proceso de solicitudes de usuario con la arquitectura MVC

Fuente: [(Hernandez, 2015)]

Ventajas

Utilizar el patrón de arquitectura MVC nos aporte ventajas:

- Separar los componentes de la aplicación, esto significa que cuando se realiza un cambio en el código no afectará a otra parte del mismo, ayudando a la escalabilidad.
- Ayuda a respetar el principio de responsabilidad única. (Hernandez, 2015a).
- Facilita el mantenimiento en caso de errores.
- Código bien organizado, legible y mantenible.
- Cualquier modificación a la aplicación, como aumentar métodos, datos o funcionalidades, implica una modificación sólo en el componente (Modelo, Vista o Controlador) implicado. (Andalucía, n.d.)

Desventajas

Como desventajas del patrón de arquitectura MVC tenemos:

- Para desarrollar una aplicación bajo el patrón de arquitectura MVC es necesario una mayor dedicación etapas iniciales del desarrollo, implementar este patrón exige al programador desarrollar un mayor número de clases que, en otros entornos de desarrollo, no son necesarias. (Andalucía, n.d.).
- La cantidad de archivos para mantener se incrementa de manera considerable.
- La curva de aprendizaje del patrón es más alta.

La simple utilización del patrón MVC en la aplicación aporta por defecto un conjunto de buenas prácticas de desarrollo de software, que ayuda a la legibilidad y mantenibilidad de código, al tener determinado código de la aplicación distribuido en archivos específicos e identificables, es decir, tendremos código separado en scripts responsables de una única funcionalidad lo que evita, por ejemplo, tener código de acceso y gestión de datos en las vistas. La reutilización de código es otra ventaja de utilizar MVC debido a que podemos tener múltiples vistas que se basan en un mismo modelo o un mismo controlador que responde con diferentes vistas.

1.4. Lenguajes de Programación

Ruby

Definición

“ Ruby es un lenguaje de programación genuinamente orientado a objetos, es decir, todo lo que se manipula dentro del lenguaje Ruby es un objeto y como resultado de estas manipulaciones también obtendremos un objeto, Ruby creado por el programador japonés Yukihiro "Matz" Matsumoto, quien comenzó a trabajar en Ruby en 1993, y lo presentó públicamente en 1995.”(Thomas, Fowler, & Hunt, 2013).

En lenguaje Ruby se caracteriza por preocuparse más por el desarrollador que por la máquina, esto quiere decir que el lenguaje es muy amigable y se lee como un lenguaje natural, Ruby busca ser natural y no simple, esto quiere decir que es un lenguaje muy expresivo, por dicho motivo Ruby inspiró la creación de otros lenguajes como Elixir y CoffeScript. (Hernandez, 2015b).

Ruby se usa principalmente en aplicaciones web, aunque esta no es su único campo de aplicabilidad, ya que Ruby es un lenguaje multiplataforma, con el que también podemos crear aplicaciones móviles nativas con el compilador RubyMotion, aplicaciones de escritorio con jRuby o utilidades de la terminal como Homebrew o SASS originalmente.

Características

- **Tipado Dinámico:** En Ruby las variables no tienen un tipo de dato estático y cada tipo de dato primitivo es un objeto dentro del lenguaje, esto quiere decir que podemos ejecutar métodos dentro de los tipos primitivos.
- **Sobre escritura del lenguaje:** Ruby es considerado un lenguaje flexible, ya que permite a sus usuarios alterarlo libremente. Las partes esenciales de Ruby pueden

ser quitadas o redefinidas a placer. Se puede agregar funcionalidad a partes ya existentes. Ruby intenta no restringir al desarrollador. (RubyOnRails, n.d.)

- **Manejo de hilos (*threading*):** Tiene un manejo de hilos independiente del sistema operativo. De esta forma, tienes soporte multi-hilo en todas las plataformas en las que corre Ruby, sin importar si el sistema operativo lo soporta o no, incluso en MS-DOS. (RubyOnRails, n.d.).
- **Ruby es fácilmente portable:** se desarrolla mayoritariamente en GNU/Linux, pero corre en varios tipos de UNIX, Mac OS X, Windows, DOS, BeOS, OS/2. (RubyOnRails, n.d.)
- **Recolector de basura:** Un verdadero *mark-and-sweep garbage collector* para todos los objetos de Ruby, basada en dos operaciones básicas, la primera detecta todos los objetos que ya no son requeridos por el software (*mark*) y la segunda limpia dichos objetos y libera el espacio de la memoria(*sweep*) dejándolos disponibles para otros objetos.(RubyOnRails, n.d.).

Javascript

Javascript con sus siglas JS es un lenguaje de programación muy popular utilizado principalmente en los navegadores web para crear aplicaciones web modernas, aplicaciones con las que puede interactuar directamente, sin tener que volver a cargar una página para cada acción, permitiendo manipular el contenido de las páginas Web mediante el DOM (*Document Object Model*), manipular datos con AJAX, crear gráficos con Canvas y también permite interactuar con el dispositivo mediante APIs (*Application Programming Interface*), pero a menudo también se utiliza este lenguaje en el lado del servidor con Node JS. (Rauschmayer, 2014).

Recientemente, JavaScript ha ganado terreno como un lenguaje de *back-end* en el lado del servidor con el éxito de Node JS. Esta plataforma provee un entorno completo de ejecución JavaScript fuera del navegador, capaz de ser usado en cualquier plataforma (Linux, MacOS y Windows). Node JS es la plataforma más usada para ejecutar JavaScript fuera del navegador, esto permite el uso de JavaScript como un lenguaje de scripting para automatizar ciertos procesos en la PC, elaborar APIs y servidores Web Sockets. (mozilla developer, 2017)

Características

- Es un lenguaje interpretado es decir que no se compila para poder ejecutarse y está orientado a objetos y eventos.
- Es multiplataforma, ya que se puede utilizar en Windows, Linux o Mac o en cualquier navegador.
- Es liviano y prototipado, debido a que usa prototipos en vez de clases para el uso de herencia. (Giampiere Julio, n.d.).

1.5. Stack de Desarrollo

Ruby on Rails

Ruby on Rails (RoR) es un *framework* para crear aplicaciones web basado en el lenguaje de programación Ruby, Rails ayuda a crear sitios web potentes de forma rápida, con un código limpio y fácil de mantener.

Ruby on Rails está diseñado para facilitar la programación web al realizar suposiciones acerca de lo que cada programador necesita para comenzar, a su vez permite escribir menos código mientras logra más que muchos otros lenguajes y *frameworks*, todo esto hace que desarrollar aplicaciones en Ruby on Rails sea más fácil y divertido.

Rails desde sus inicios estuvo pensado en la productividad y en enfocarse más en los desarrolladores que en las máquinas, por eso supone que hay una “mejor” manera de hacer las cosas, y está diseñado para alentar esa manera, siguiendo mejores prácticas de desarrollo.

Rails está basado en su filosofía y esta consta de dos principios fundamentales, los cuales son:

- ***Don't Repeat Yourself:*** Con sus siglas DRY, es un principio de desarrollo de software que establece que cada pieza de conocimiento debe tener un lugar único e inequívoco dentro de un sistema, lo que elimina la redundancia de código dentro de un sistema. Al no escribir la misma información una y otra vez, nuestro código es más fácil de mantener, más escalable y menos problemático.
- ***Convention over Configuration:*** También conocido como CoC, Rails tiene su opinión acerca de la mejor manera de realizar muchas cosas en una aplicación web, y predetermina este conjunto de convenciones, con esto se busca minimizar el número de decisiones que un desarrollador necesita tomar durante la programación, por lo que se gana simplicidad sin perder flexibilidad.

Patrón de Arquitectura MVC en Ruby on Rails

Rails es un *framework* de aplicaciones web que incluye todo lo necesario para crear aplicaciones web respaldadas por bases de datos de acuerdo con el patrón *Model-View-Controller* (MVC).

- **Modelo:** El modelo encapsula la lógica de negocios y datos específica de la aplicación, en Rails los modelos son controlados por el módulo de *ActiveRecord*, el cual, es un ORM (*Object Relational Mapping*) que se encarga de mapear las tablas en la base de datos de la aplicación y crea una base de datos orientada a objetos virtual, sobre la base de datos relacional. Esto posibilita el uso de las características propias de la orientación a objetos (básicamente herencia y polimorfismo).
- **Vista:** La capa de las vistas se compone de plantillas que son los responsables de presentar representaciones adecuadas a las peticiones del usuario, las plantillas pueden venir en una variedad de formatos, pero la mayoría de las plantillas de vista son HTML con código Ruby incorporado (archivos ERB). Las vistas se representan normalmente para generar una respuesta de un controlador o para generar el cuerpo de un correo electrónico, en Rails, la generación de vistas está manejada por el módulo *Action View*.
- **Controlador:** La capa de controlador en Rails es responsable de gestionar las solicitudes HTTP entrantes y proporcionar una respuesta adecuada a la petición del usuario, por lo general, esto significa devolver HTML, pero los controladores de Rails también pueden generar XML, JSON, PDF, XSLX, vistas específicas de dispositivos móviles entre otros. Los controladores cargan y manipulan los modelos, y renderizan plantillas de vista para generar la respuesta HTTP apropiada, en Rails, las solicitudes entrantes son enrutadas por el módulo de *Action Dispatch* a un controlador apropiado, y las clases de controlador se derivan de *ActionController*. Estos dos módulos están agrupados en el módulo *Action Pack*.

Componentes de Ruby on Rails

TABLA 3: Componentes integrados en Ruby on Rails.

Componentes de Ruby on Rails	
Tiempo Real	
Action Cable	Es un <i>framework</i> para integrar <i>WebSockets</i> con una aplicación de Rails.
Correos Electrónicos	
Action Mailer	Es una librería que se encarga de generar y enviar correos electrónicos.
Manejo y respuesta a solicitudes web	
Action Pack	Action Dispatch: Este componente analiza la información sobre las solicitudes web, maneja el enrutamiento tal como fue definido por el programador y realiza un procesamiento avanzado relacionado con HTTP, como parámetros de decodificación en cuerpos POST, PATCH o PUT, manejo de la lógica, cookies y sesiones de caché HTTP.
Vistas de usuario	
Action View	Es un <i>framework</i> para manejar la búsqueda de plantillas y representación de las mismas, y proporciona <i>helpers</i> de visualización que ayudan cuando se crean formularios HTML.
Trabajos en segundo plano	
Active Job	Es un <i>framework</i> para declarar trabajos en segundo plano y ayuda a que se ejecuten, estos trabajos pueden ser desde limpiezas programadas hasta envíos de correos o cargos de facturación.
Base de datos	
Active Record	Es un ORM que se encarga de mapear las tablas de la base de datos en clases, las filas de una tabla en objetos y las columnas en atributos
Almacenamiento	
Active Storage	Es una librería que ayuda a subir y referenciar archivos locales o en la nube como Amazon S3, Google Cloud Storage o Microsoft Azure Storage y adjuntar dichos archivos a las aplicaciones Rails.
Modelos	
Active Model	Es una biblioteca que contiene varios módulos e interfaces utilizados para crear clases que necesitan algunas características presentes en Active Record.
Utilidades	
Active Support	Es una colección de clases de utilidad y extensiones de biblioteca estándar que son útiles para Rails, que también se pueden usar independientemente fuera de Rails. (RubyOnRails, n.d.)

Los componentes que intervienen en el flujo de proceso de Ruby on Rails desde que se recepa la petición de un cliente ya sea un usuario final u otra aplicación que consuma los servicios de la aplicación se muestra en la figura 3.

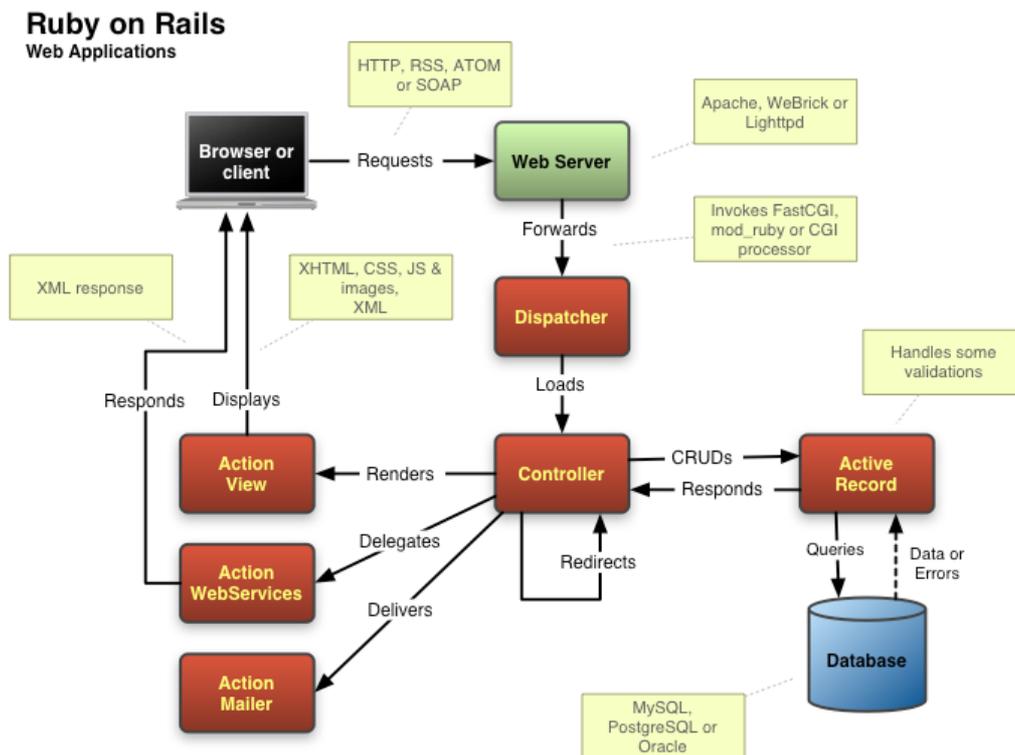


Fig. 3. Modelo de referencia que representa la arquitectura del framework

Fuente: [(Mejia, 2011)]

En base al estudio realizado Ruby on Rails tiene su propia manera de hacer las cosas y gracias a su filosofía de convención sobre configuración hace posible un desarrollo web ágil al ahorrar tiempo al no tener que realizar diferentes configuraciones que son importantes para el correcto funcionamiento de cualquier aplicación Web, para esta configuración Ruby hace “suposiciones” sobre lo que se quiere hacer y la manera de realizarlo en lugar de especificar cada pequeña configuración en un sin fin de archivos, con esto Rails ayuda a centrarse únicamente en la programación de la aplicación en sí y obviar pasos anteriores de configuraciones, es decir, si el desarrollador sigue las convenciones de Rails el *framework* es capaz de hacer gran parte del trabajo, mediante esto se logra crear proyectos Web de calidad en tiempos muy competitivos usando metodologías de programación ágiles. La lista de convenciones de Ruby on Rails se muestran en la tabla 4.

TABLA 4 Convenciones de Ruby on Rails

Convenciones de Ruby on Rails	
Convención	Definición
Convenciones en los modelos	
Pluralización en los modelos	Rails espera que las tablas en la base de datos se encuentren en plural: Productos, Coches. Y que su respectivo modelo este nombrado singularmente: Producto, Coche con la nomenclatura CamelCase, con el fin de establecer las relaciones entre los scripts de modelos con la base de datos.
Nombres reservados en los campos de las tablas.	Si dentro de una tabla existen los campos <code>created_at</code> o <code>updated_at</code> , el framework se encargará de insertar automáticamente las fechas de inserción y edición en dichos campos respectivamente.
Nomenclatura de campos de las tablas y <i>Schemas</i> .	Rails espera que en las tablas de la base de datos exista un campo id como clave primaria, con el fin de generar automáticamente en el Active Record las relaciones de claves primarias y foráneas entre las tablas.
Convenciones en los controladores	
Nombrado de clases y acciones.	Los nombres de las clases de los controladores deben estar en CamelCase y las acciones en snake_case, las acciones <i>index</i> , <i>show</i> , <i>new</i> , <i>update</i> y <i>destroy</i> , están automáticamente ligadas a los verbos HTTP y rutas estándar que crea el framework.
Convenciones en las vistas	
Nombrado de controladores y acciones.	Si Rails posee un controlador llamado <code>workers_controller.rb</code> con una acción denominada <code>list_workers</code> , el framework espera que exista un archivo <code>list_workers.html.erb</code> dentro de una carpeta <code>workers</code> en los archivos de las vistas.
Convenciones en las rutas	
Escritura de las rutas.	Los nombres de las rutas están en escritas en snake_case con el nombre del controlador en plural con el fin de mapear las rutas y asignarlas a los controladores.

Otra ventaja que aporta Rails, es que está desarrollo bajo el lenguaje Ruby, que es un lenguaje plenamente orientado a objetos y muy expresivo, ya que con pocas líneas de código se puede lograr muchas cosas, con lo que la legibilidad del código escrito es mayor

y la probabilidad de errores se reduce. Otro aspecto positivo de Rails es que cuenta con un sistema de *plugins* denominado gemas que permiten modificar o ampliar el comportamiento del *framework* de manera fácil, estas gemas se pueden crear de manera sencilla y cuentan con una gran comunidad, las gemas que son muy usadas se las acaba por incorporarlas en el núcleo de Rails; y al contrario, si una parte del *framework* no es muy utilizada se lo separa del *framework* como gema, esto es muy importante ya que no se convierte en un *framework* excesivamente grande en donde cabe todo con una curva de aprendizaje elevada.

Todos estos aspectos descritos apoyan a que Rails es un *framework* con el que se puede desarrollar aplicaciones Web completas con muy poco código, el desarrollador se puede centrar únicamente en la lógica de negocio y dejar que el *framework* se encargue de muchos detalles de implementación, esto crea conflicto con otros *frameworks* de desarrollo web que están pensados en cubrir cualquier caso posible, y acaban por tener una arquitectura muy grande y compleja y muchas veces da la impresión de que en lugar de desarrollar, se está luchando contra el *framework*.

La arquitectura de Rails en el presente proyecto se muestra en la figura 4, en el cual, Rails recibirá las peticiones que provengan del *reverse proxy* NGINX mediante el servidor de aplicaciones phusion passenger, realizará los procesos internos respectivos y entregará los datos solicitados a la librería React que se encarga de las vistas.

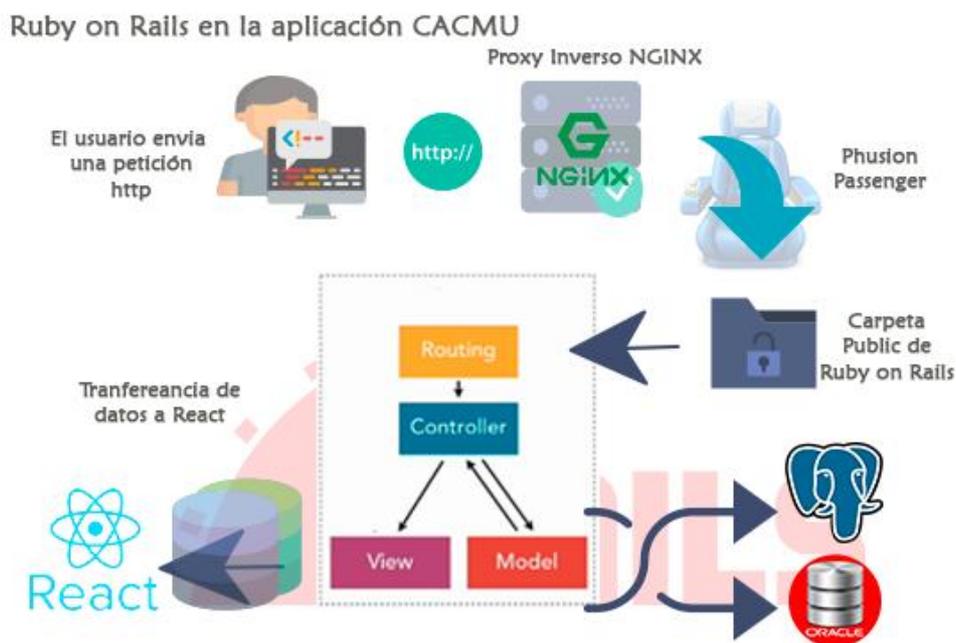


Fig. 4 Arquitectura de Rails en la aplicación

React Js

ReactJS o simplemente React es una librería Javascript desarrollada por los ingenieros de Facebook para la resolver los desafíos de creación de interfaces de usuario complejas con un conjunto de datos que cambian en el tiempo, esta es su principal área de trabajo, pero con todo el ecosistema de aplicaciones, librerías y componentes, con React encontramos un excelente aliado para hacer todo tipo de aplicaciones web, SPA (*Single Page Application*) o aplicaciones para móviles. (Gackenheimer, 2015)

React fue lanzado en el 2013, actualmente es mantenido por Facebook, Instagram, otras compañías y el aporte de la comunidad, cabe recalcar que React es una librería *open source* de código abierto con licencia BSD + patent, fue desarrollada en primera instancia para solventar las necesidades del desarrollo de la red social de Facebook, al tener la necesidad de un desarrollo ágil, pero con un mayor rendimiento que otras alternativas existentes en el mercado que maneje datos a gran escala.

React intenta ayudar a los desarrolladores a crear aplicaciones web complejas que utilizan un gran intercambio de datos. Su principio es sencillo “**declarative and composable**”. React solo se preocupará de la interfaz de usuario de la aplicación; si se piensa en el paradigma de software Modelo-Vista-Controlador (MVC) React es únicamente la V. Esto hace que se pueda usarlo junto con otras librerías JavaScript o con *frameworks* como AngularJS, Backbone o Ember. Además, React gracias a su enorme y creciente comunidad, está generando una gran biblioteca de extensiones, las cuales en muchas ocasiones no se ocupan solo de la interfaz de usuario, sino que ayudan a complementar todas las necesidades de una aplicación web. (Jimenez, 2016).

Características de React

Declarativo

React hace que sea fácil crear IU interactivas. El programador diseñará vistas simples para cada estado de la aplicación, y React actualizará y hará render de manera eficiente solo los componentes correctos cuando los datos cambien. Las vistas declarativas hacen que el código sea más predecible, más simple de entender y más sencillo de depurar. (Baquero, 2017).

Basado en Componentes

Con React se puede construir componentes altamente reutilizables que administran su propio estado y encapsulan su propio comportamiento, una vista y un estado, un componente se utiliza para resolver pequeños problemas a partir de esto unos

componentes se apoyarán en otros para resolver problemas mayores y al final la aplicación será un conjunto de componentes que trabajan entre sí. (Baquero, 2017).

Learn Once, Write Anywhere

React no realiza suposiciones sobre el resto del *stack* de desarrollo, por lo que se puede desarrollar nuevas funciones en React sin reescribir el código existente. React también puede mostrarse en el servidor utilizando NodeJs y en aplicaciones móviles usando React Native. (Jimenez, 2016)

DOM Virtual de React

Una de las principales características que lo distinguen de los demás *frameworks* y librerías es que posee un desempeño muy alto al momento de renderizado de la aplicación gracias al DOM virtual que maneja, esto no quiere decir que React no interactúe con el DOM real del navegador, pero todas las operaciones las realiza sobre el DOM virtual que es más rápido. (Jimenez, 2016)

De manera sencilla, se trata el DOM como si de una GPU se tratara, por lo que para cada objeto del DOM hay un "objeto DOM virtual" correspondiente que lo mantiene en memoria con las mismas propiedades que el DOM real pero más liviano, hacer cambios sobre el DOM real es lento pero la manipulación del DOM virtual es mucho más rápida, porque no se dibuja nada en la pantalla. (Jimenez, 2016).

Jimenez (2016) establece que para dar una idea con cada cambio en un componente de React sucede lo siguiente:

1. Se genera un nuevo árbol Virtual DOM en memoria
2. Se compara con el árbol previo antes del cambio.
3. Se deciden cuáles son los cambios mínimos a realizar
4. Se envían dichos esos cambios a la cola
5. Se procesan los cambios en el navegador.

React desde su inicio no era la librería JavaScript más usada, pero ha ido creciendo rápido y se ha convertido en la librería más popular para el desarrollo de interfaces de usuario orientado a componentes, las búsquedas web de *Frameworks & Librerías* JavaScript indican un gran aumento en la popularidad.

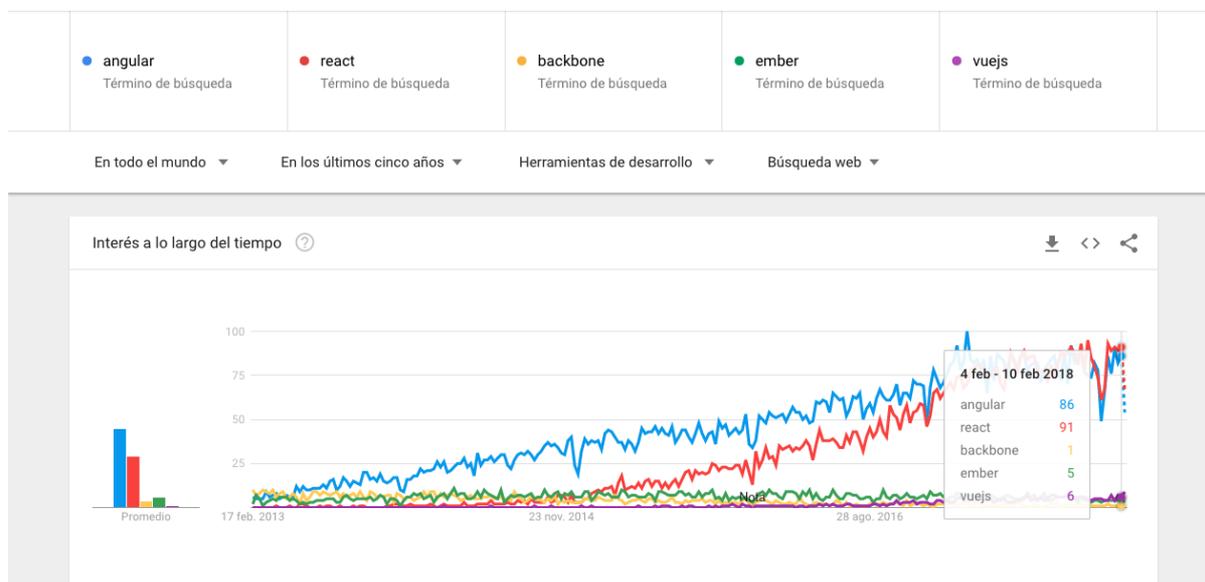


Fig. 5. Búsquedas Web por frameworks y librerías Javascript.

Fuente: *Web Searches for JavaScript Frameworks & Libraries* (Google Trends)

En base al estudio de la librería React se establece que, es una librería que se acopla perfectamente al reciente estándar y funcionalidades de JavaScript que trae ES6, lo que aporta con sencillez para realizar ciertas tareas adicionando la posibilidad de usar Promesas de manera nativa, *arrow functions* para declaración de funciones anónimas que preserven el *scope* de las funciones, sistema de manejo de módulos nativos y soporte de clases.

Hace años no había tanta lógica de negocio en el lado del *front-end* de las aplicaciones Web como lo hay ahora, React nos ofrece la facilidad de tener vistas asociadas a los datos lo que nos permite no escribir código para actualizar la página cuando estos datos varían, a su vez React está pensado para manejar grandes cantidades de datos de mejor manera que otros *frameworks* o librerías *front-end* gracias a que implementa un DOM virtual el cual tiene un mejor rendimiento que otras librerías que se basan en *data binding* que, para ligar los modelos a las vistas crea muchas conexiones entre datos y modelos, esta característica de React es importante en el desarrollo del presente proyecto ya que se busca tener vistas orientadas a componentes que estén ligadas a una cantidad considerable de datos de la cooperativa, y que, si los datos varían la vista esté en la capacidad de reflejar estos cambios.

React por si sola es una librería ligera que renderiza componentes en el DOM, pero al integrarse con su ecosistema puede ser un auténtico *framework front-end* con su propia arquitectura como lo es Angular, lo que permite al desarrollador ser más modular debido

a que conforme las necesidades de la aplicación crezcan se puede integrar funcionalidades como *routing* o la implementación de una arquitectura como Flux, esto trae como ventaja que React no está atado a nada, con un *framework* como Angular o cualquier otro, si se desea utilizar un sistema de *routing* se debe usar el que está construido en el propio *framework*, con React se puede usar cualquier librería que se desee como Vanilla, PageJs o react-router. Esto provoca que React sea tan ligero como sean las necesidades de la aplicación.

En base a las declaraciones descritas anteriormente, React es la librería que mejor satisface las necesidades del proyecto ya que Ruby on Rails manejará mayor parte de la aplicación, y React se encargará de renderizar los componentes en las vistas y de reutilizar componentes dentro de otros para expandir las funcionalidades de estos. La arquitectura de la aplicación en las vistas se muestra en la figura 6.

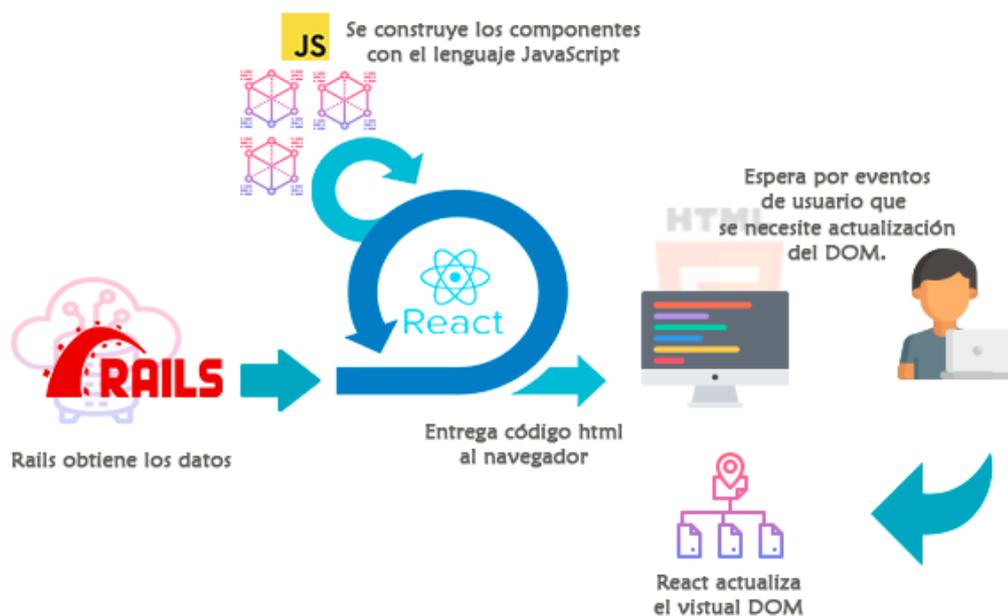


Fig. 6 Arquitectura de React en la aplicación

Sistema de gestor de base de datos PostgreSQL

“PostgreSQL es un sistema de base de datos relacional de objetos de código abierto. Tiene más de 15 años de desarrollo activo y una arquitectura comprobada que le ha valido una sólida reputación de fiabilidad, integridad de datos y corrección. Se ejecuta en todos los principales sistemas operativos, incluidos Linux, UNIX (AIX, BSD, HP-UX, macOS, Solaris) y Windows. Es totalmente compatible con ACID, tiene soporte completo para claves externas, combinaciones, vistas, disparadores y procedimientos almacenados. Incluye la mayoría de los tipos de datos SQL: 2008, incluidos INTEGER, NUMERIC,

BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL y TIMESTAMP. También es compatible con el almacenamiento de objetos grandes binarios, incluyendo imágenes, sonidos o video. Tiene interfaces de programación nativas para C / C ++, Java, .Net, Perl, Python, Ruby, Tcl, ODBC, entre otros.” (PostgreSQL, n.d.).

PostgreSQL es gratuito y libre, además de que hoy nos ofrece una gran cantidad de opciones avanzadas. De hecho, es considerado el motor de base de datos más avanzado en la actualidad. (PostgreSQL, n.d.).

Características

Dorantes (2015) afirma que una característica interesante de PostgreSQL es el control de concurrencias multiversión; o MVCC por sus siglas en inglés. Este método agrega una imagen del estado de la base de datos a cada transacción. Esto nos permite hacer transacciones eventualmente consistentes, ofreciéndonos grandes ventajas en el rendimiento.

En Postgres no se requiere usar bloqueos de lectura al realizar una transacción lo que nos brinda una mayor escalabilidad. También PostgreSQL tiene *Hot-Standby*. Este permite que los clientes hagan búsquedas (sólo de lectura) en los servidores mientras están en modo de recuperación o espera. Así podemos hacer tareas de mantenimiento o recuperación sin bloquear completamente el sistema. (Dorantes, 2015).

Dorantes, (2015) informa que postgresql aporta mucha flexibilidad a nuestros proyectos. Por ejemplo, nos permite definir funciones personalizadas por medio de varios lenguajes. Algunos son:

- PL/pgSQL
- PL/Tcl
- PL/Perl
- PL/Python
- PL/PHP
- PL/Ruby
- PL/Java

En el presente proyecto el gestor de base de datos almacenará la información de la aplicación en tablas relacionales, teniendo en cuenta que no se accederá a dicha información con código SQL directamente, sino que, se utiliza el Active Record de Ruby on Rails para mapear las

tablas de la base de datos postgresql a objetos de Ruby, en la figura 7 se muestra la arquitectura de la aplicación al nivel de la capa de datos.



Fig. 7 PostgreSQL y Active Record

1.6. Lenguaje de Marcado Haml

Haml es un lenguaje de marcado de abstracción HTML (*HTML abstraction markup language*), en otras palabras, es un lenguaje que se puede usar en lugar de HTML para crear archivos HTML. Está diseñado para hacer que el marcado sea simple, fácil de escribir, limpio, legible y adhiriéndose a los principios DRY (*Don't repeat yourself*). (Niksinski, 2013).

Haml fue creado por Hampton Catlin en mayo del año 2006, y luego mantenido y desarrollado por Nathan Weizenbaum por muchos años. La implementación oficial del marcado Haml fue desarrollada específicamente para el lenguaje Ruby con y fue lanzada con complementos para el *framework* Ruby on Rails, sin embargo, también hay implementaciones para otros lenguajes como Python, PHP, Perl, Java y más. (Niksinski, 2013).

Haml es utilizado por desarrolladores profesionales y es elogiado por su limpieza y agilidad de uso como se muestra en la imagen a continuación.

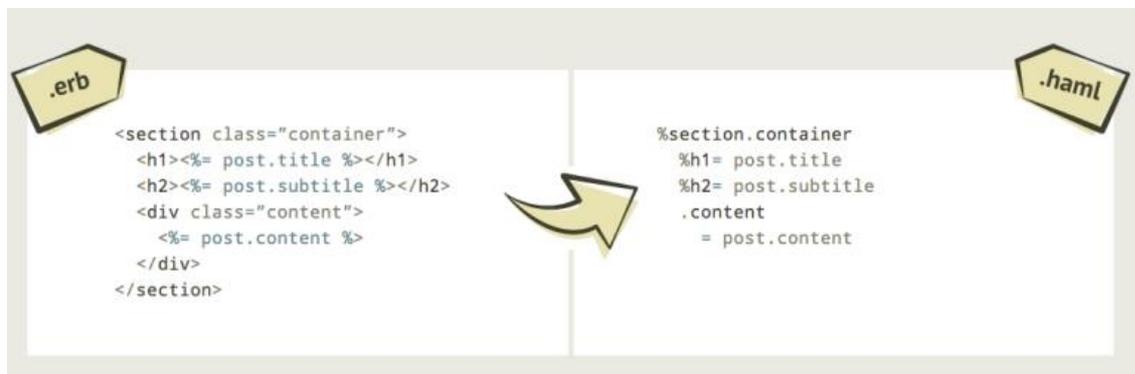


Fig. 8. Ejemplo del marcado HAML

Fuente: [(Niksinski, 2013)]

Gackenhaimer (2015) afirma que este marcado funciona como un reemplazo para los sistemas de plantillas de página en línea como PHP, ASP y ERB que es lenguaje de plantillas utilizado en la mayoría de las aplicaciones de Ruby on Rails. Haml es en sí mismo es una descripción de HTML y, por lo tanto, evita la necesidad de codificar explícitamente HTML en la plantilla con algún código para generar contenido dinámico.

1.7. Estilos CSS

Preprocesador de hojas de estilo SASS

Introducción

Un preprocesador de CSS es una herramienta que nos permite escribir pseudo-código CSS que luego será convertido a CSS real. Ese pseudo-código se conforma de variables, condiciones, bucles o funciones. Podríamos decir que tenemos un lenguaje de programación que genera CSS. (Maquetando, 2015).

El objetivo de estos preprocesadores es tener un código más sencillo de mantener y editar. Los preprocesadores incluyen características tales como variables, funciones, mixins, anidación o modularidad. (Maquetando, 2015).

Sass

Sass (acrónimo de *Syntactically Awesome StyleSheets*) es una extensión de CSS que añade características muy potentes y elegantes a este lenguaje de estilos. Sass permite el uso de variables, reglas CSS anidadas, *mixins*, importación de hojas de estilos y muchas otras características, al tiempo que mantiene la compatibilidad con CSS. Sass permite organizar mejor las hojas de estilos grandes y permite ser mucho más productivo con las hojas de estilos pequeñas. (Catlin, Weizenbaum, & Eppstein, n.d.).

Catlin et al., establece que: "Sass permite el uso de dos sintaxis diferentes para crear sus archivos. La primera sintaxis, conocida como SCSS (del inglés, *Sassy CSS*) es una extensión de la sintaxis de CSS3. Esto significa que cualquier hoja de estilos CSS3 válida también es un archivo SCSS válido."

La segunda sintaxis, conocida como "sintaxis indentada" o simplemente "sintaxis sass" permite escribir los estilos CSS de manera más concisa. En este caso, el anidamiento de selectores se indica con tabulaciones en vez de con llaves y las propiedades se separan con saltos de línea en vez de con puntos y coma. Algunos diseñadores consideran que esta segunda sintaxis es más sencilla de leer y más rápida de escribir que SCSS. (Catlin et al., n.d.).

En la figura 9 se muestra un ejemplo de la transpilación que realiza Sass al lenguaje CSS como se utilizará en el proyecto.



```
Sintaxis Sass
section {
  height: 100%;
  width: 100%;
  .card {
    height: 50px;
    width: 50px;
    .button {
      color: rgb(46, 48, 146);
    }
  }
}

Sintaxis CSS
section{
  height: 100%;
  width: 100%;
}
section .card {
  height: 50px;
  width: 50px;
}
section .card .button {
  color: rgb(46, 48, 146);
}
```

Fig. 9 Transpilación de Sass a Css

Características de Sass

Maquetando (2015) afirma que Sass incluye las siguientes características:

- 100% compatible con CSS3.
- Permite el uso de variables, anidamiento de estilos y *mixins*.
- Incluye numerosas funciones para manipular con facilidad colores y otros valores.
- Permite el uso de elementos básicos de programación como las directivas de control y las librerías.
- Genera archivos CSS bien formateados y permite configurar su formato.
- Integración con Firebug gracias a la extensión FireSass.

Material Design

Material Design es una guía integral para el diseño visual, de movimientos y de interacción en distintas plataformas y dispositivos, fue anunciada durante el Google I/O del año 2014, *Material Design* es un concepto, una filosofía, unas pautas enfocadas al diseño utilizado en Android, pero también en la web y en cualquier plataforma. El encargado de crear *Material Design* y máximo responsable de diseño en Google es el chileno Matías Duarte. (Pérez, 2014).

Fundamentos de Material Design

Pérez (2014) establece que: *Material Design* recibe su nombre por estar basado en objetos materiales, piezas colocadas en un espacio (lugar) y con un tiempo (movimiento)

determinado. Es un diseño donde la profundidad, las superficies, los bordes, las sombras y los colores juegan un papel principal como se muestra en la siguiente figura 10.

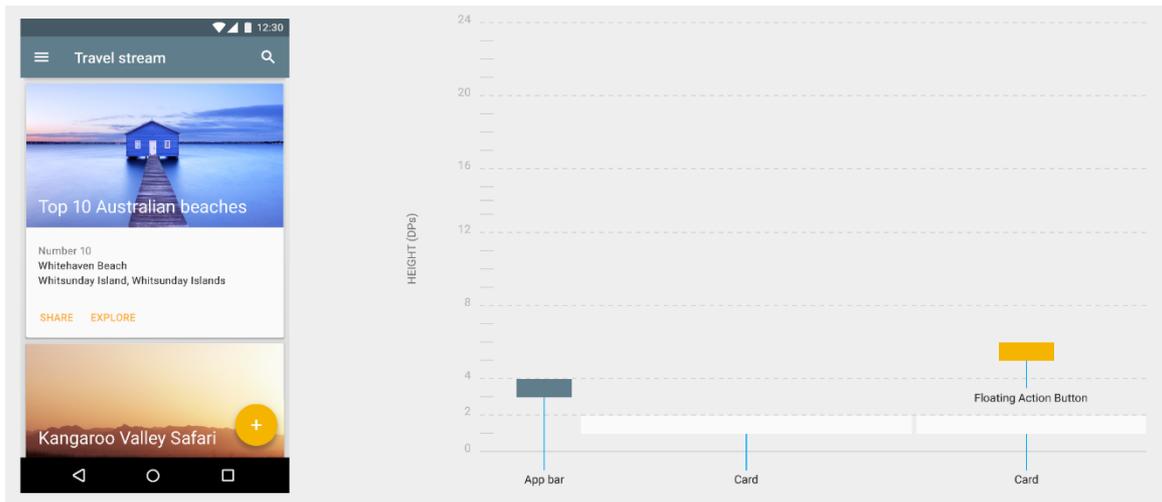


Fig. 10. Ejemplo de materiales y elevaciones de material design.

Fuente: [(Google, 2014)]

Precisamente este diseño basado en objetos es una manera de intentar aproximarse a la realidad, algo que en un mundo donde todo es táctil y virtual es difícil. *“Material Design quiere guiarse por las leyes de la física, donde las animaciones sean lógicas, los objetos se superpongan, pero no puedan atravesarse el uno al otro.”* (Pérez, 2014)

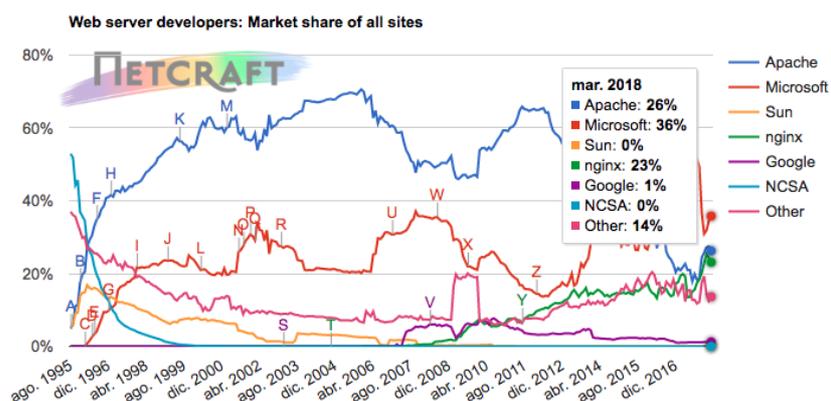
1.8. Servidor NGINX

Un servidor web es un servidor que aloja una aplicación que escucha las solicitudes HTTP. Es responsabilidad del servidor web escuchar (es decir, comprender HTTP) lo que dice el navegador y responder de manera adecuada. En ocasiones, podría ser tan simple como obtener un archivo del sistema de archivos y entregarlo al navegador web. En otras ocasiones, delega la solicitud a un controlador que realiza una lógica complicada y devuelve la respuesta procesada al servidor web, que a su vez la transfiere al cliente. Por lo general, el servidor que aloja el software del servidor web se denomina servidor web o servidor web *front-end*. (Soni, 2016).

“NGINX un servidor web y proxy inverso de código abierto ligero de alto rendimiento, que también incluye servicios de correo electrónico con acceso al Internet *Message Protocol* (IMAP) y al servidor *Post Office Protocol* (POP). Además, NGINX está listo para ser utilizado como un proxy inverso. En este modo, NGINX se utiliza para equilibrar la carga entre los servidores *back-end*, o para proporcionar almacenamiento en caché para un servidor *back-end* lento.” (Acens, 2013).

Acens (2013) afirma que hasta poco tiempo atrás, Apache era el rey indiscutible en el mundo de los servidores web, pero año tras año, la popularidad de este servidor web ha ido en aumento y empresas punteras de Internet como Facebook o WordPress lo utilizan en sus portales.

Según un estudio llevado a cabo por el portal Netcraft, a principios de marzo de 2018 la cuota de mercado Nginx ascendía al 23,0%, situándolo en tercera posición detrás de Apache y IIS de Microsoft como se muestra en la figura 11.



Developer	February 2018	Percent	March 2018	Percent	Change
Microsoft	634,359,419	34.50%	633,719,941	35.80%	1.29
Apache	504,701,560	27.45%	464,340,535	26.23%	-1.22
nginx	447,224,456	24.32%	409,124,174	23.11%	-1.22
Google	22,022,633	1.20%	21,802,670	1.23%	0.03

Fig. 11. Cuota del mercado de NGINX.

Fuente: [(Netcraft, 2018)]

Características

NGINX es un servidor web, en esta sección se presenta alguna de sus características más importantes.

Más que un servidor Web

En esencia, puede considerarse que Nginx es un servidor proxy inverso basado en eventos, esto puede resultar extraño para muchas personas, porque generalmente se dice que Nginx es un servidor web. Un proxy inverso es un tipo de servidor proxy que recupera recursos de los servidores en nombre de un cliente. En la figura 12 se ilustra lo que hace un servidor proxy. (Soni, 2016).

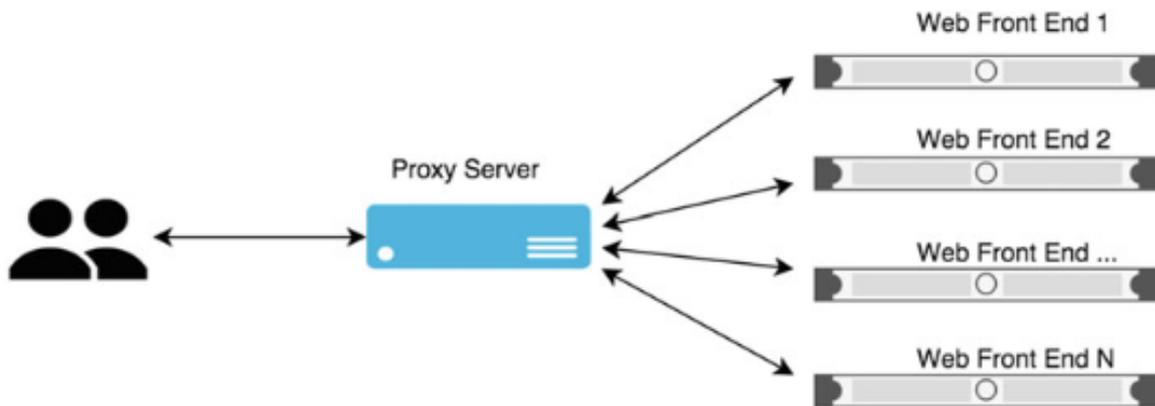


Fig. 12. NGINX como un servidor proxy.

Fuente: [(Soni, 2016)]

Servidor Web asíncrono

Nginx obtiene gran parte de su rendimiento gracias a su arquitectura asíncrona y basada en eventos, mientras que a Apache e IIS les gusta crear nuevos hilos por conexión, que son de naturaleza de bloqueo. Tanto IIS como Apache manejan los hilos usando técnicas de programación multiproceso. Nginx difiere completamente en el enfoque. No crea un hilo separado para cada solicitud. En cambio, depende de los eventos. (Soni, 2016).

Proxy inverso y balanceo de carga

Balanceo de carga, distribuye la carga entre los servidores que formen parte de la estructura, redirigiendo cada vez la petición hacia aquella máquina que tenga una menor carga. (Soni, 2016).

Bajo requerimiento de recursos y consumo

Pequeñas cosas que hacen mucho definen a Nginx, mientras que otros servidores web normalmente permiten una arquitectura simple *plug-and-play* para complementos que utilizan archivos de configuración, Nginx requiere que recompile la fuente con los módulos requeridos. Cada módulo que requiere se carga directamente dentro de un proceso Nginx. Dichos ajustes junto con diferencias arquitectónicas inteligentes aseguran que Nginx tenga una memoria y una huella de CPU muy pequeñas en el servidor y rinda un rendimiento mucho mejor que su competencia. (Soni, 2016).

Terminación SSL

Secure Sockets Layer es una necesidad para cualquier sitio web que se ocupe de datos confidenciales, y al igual que cualquier otra necesidad, hay un costo involucrado.

Cuando se trata de tráfico web, SSL también induce una sobrecarga de procesamiento adicional en el lado del servidor donde debe descifrar la solicitud cada vez. Si se elimina el SSL, se está abriendo para los ataques y si usa SSL, termina perdiendo un poco de velocidad (o un costo adicional debido a la ampliación). Como Nginx tiene la capacidad de actuar como equilibrador de carga, también puede proporcionarle un trabajo adicional. Básicamente, la idea de una terminación SSL (Figura 13) es que la solicitud llegará al equilibrador de carga en un canal seguro, pero se enviará a los otros servidores web sin SSL. De esta forma, su servidor web actúa más rápido y, finalmente, sus solicitudes también se dirigen a los clientes de forma segura. (Soni, 2016).

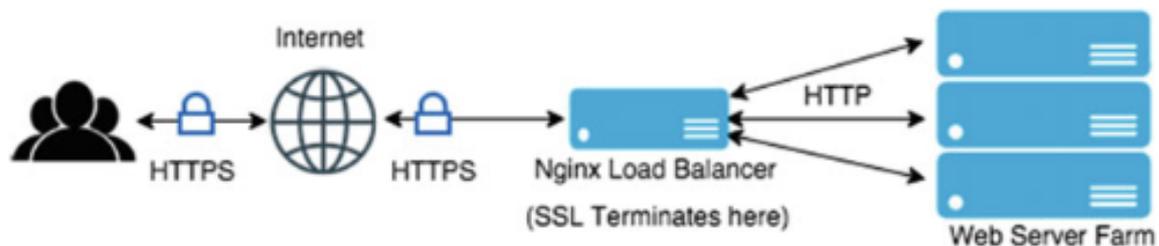


Fig. 13. NGINX como balanceador de carga y terminación SSL

Fuente: [(Soni, 2016)]

Transmisión de video HTTP usando MP4 / FLV / HDS / HLS

La entrada / salida (IO) en Nginx no se bloquea si el cliente es lento. La transmisión de video suele ser un proceso muy intensivo de IO, y Nginx hace un gran trabajo aquí. Tiene múltiples módulos que lo ayudan a proporcionar servicios de transmisión. (Soni, 2016).

Actualizaciones sin tiempo de inactividad utilizando binarios en vivo

Esta es probablemente una de las características más poderosas de Nginx. En los mundos IIS o Apache, no puede actualizar su servidor web sin bajar el servicio. Nginx genera un proceso maestro cuando se inicia el servicio. Su objetivo principal es leer y evaluar archivos de configuración. Aparte de eso, el proceso maestro inicia uno o más procesos de trabajo que hacen el trabajo real manejando las conexiones del cliente. (Soni, 2016).

Si se necesita actualizar el binario, hay pasos simples y comandos que debe emitir para que los nuevos procesos de trabajo se ejecuten en paralelo con los anteriores. Las nuevas solicitudes se enviarán a los procesos de trabajo más nuevos que tengan cargada

la última configuración. Si, por casualidad, se descubre que la actualización está causando problemas, se puede simplemente emitir otro conjunto de comandos que devolverá las solicitudes de forma elegante al proceso anterior que ya tiene la configuración de trabajo anterior cargada en él. (Soni, 2016).

Ventajas

Algunas de las principales ventajas de Nginx son:

- Acens (2013) afirma que NGINX: “Es capaz de manejar más de 10.000 conexiones simultáneas con un uso bajo de memoria.”
- Balanceo de carga, distribuye la carga entre los servidores que formen parte de la estructura, redirigiendo cada vez la petición hacia aquella máquina que tenga una menor carga. (Acens, 2013).
- Alto rendimiento: ‘Aguanta’ más y responde más rápido comparado con otros servidores web. (Jiménez, 2015).
- Es ligero, consume menos recursos para realizar la misma función que otros servidores web.
- Es multiplataforma, Nginx se puede instalar en diferentes sistemas Unix (GNU/Linux, BSD, Solaris, Mac OS X, etc.) y también en Windows.

Desventajas

Nginx presenta algunas desventajas como:

- No tiene módulos dinámicos, es decir, no se pueden instalar nuevos módulos en caliente sin tener que compilar Nginx con el módulo deseado y volver a instalarlo. (Jiménez, 2015).
- Documentación: No existe mucha información para resolver algunos tipos de problemas.
- Nginx no tiene ningún módulo por defecto para interpretar archivos PHP. Se deben gestionar por FastCGI. (Jiménez, 2015).

CAPITULO 2

DESARROLLO

2.1. Planificación

La primera fase del proceso de desarrollo de software es la planificación; en esta etapa al igual que el resto de las etapas serán analizadas una vez durante el desarrollo de cada sprint de la metodología, sin embargo, antes de comenzar con las iteraciones de los sprint es necesario realizar una planificación macro, es decir una planificación inicial.

Planificación del proyecto

Para el presente trabajo de grado se utilizó la metodología scrum para la planificación de las actividades del proyecto, para realizar las planificaciones de los *sprints*, la metodología de scrum nos provee los artefactos que ayudarán a la realización y cumplimiento de las tareas establecidas. En primera instancia se definirá los roles de la metodología, los cuales son las personas que tienen relación directa o indirecta con el proyecto, el primer artefacto que se creará es el Producto Backlog, el cual permite conocer las tareas y funcionalidades que se implementarán en el sistema categorizadas por prioridades con el método *MoSCow*, también se definirá.

Roles de la metodología Scrum

TABLA 5: Roles de la metodología Scrum para el proyecto.

Persona	Contacto	Rol
Santiago J. López C.	sjlopezc@utn.edu.ec	Coordinador / Scrum Master
Ph.D Nataly Cadena	nattycadena@yahoo.com	Gestor de Producto / Product Owner.
Santiago J. López C.	sjlopezc@utn.edu.ec	Equipo técnico / Team

Formato del product backlog

Para la elaboración del product baclog utilizaremos el formato que se muestra en la **tabla 5**, el formato nos muestra un identificador de la tarea, el módulo del sistema al cual pertenece, el número del sprint al que corresponde, una descripción de la tarea, estimación en días de la tarea y el responsable de cada una de ellas.

IDENTIFICADOR DE LA HISTORIA	MODULO	SPRINT	TAREA	ESTIMACIÓN	RESPONSABLE
sp1 – 14	CJ	1	Visualización de resultados	2	Santiago J. López C.
sp1 – 15	CJ	1	Buscador de juicios	1	Santiago J. López C.
sp1 – 16	CJ	1	Reportes de monitoreos	5	Santiago J. López C.
sp1 – 17	CJ	1	Convertir juicios de bienes a sin bienes y viceversa.	3	Santiago J. López C.
sp1 – 18	CJ	1	Gestión de etapas judiciales	2	Santiago J. López C.
sp1 – 19	CJ	1	Asignación de juicios a los asesores	2	Santiago J. López C.
sp1 – 20	CJ	1	Clasificación de procesos de los juicios	3	Santiago J. López C.
sp1 – 21	CJ	1	Privilegio de acceso a asesores y jefaturas de cobranzas	4	Santiago J. López C.
sp1 – 22	CJ	1	Eliminación de monitoreos	3	Santiago J. López C.
sp1 – 23	CJ	1	Gestión de las tablas como administrador.	6	Santiago J. López C.
sp2 – 01	CR	2	Visualización de la planificación de la cartera por agencia y asesor	8	Santiago J. López C.
sp2 – 02	CR	2	Visualización de la cartera vencida por agencia y asesor	6	Santiago J. López C.
sp2 – 03	CR	2	Visualización de la eficiencia de la cartera por agencia y asesor	6	Santiago J. López C.
sp2 – 04	CR	2	Visualización de las matrices de riesgo por agencia y asesor	8	Santiago J. López C.
sp2 – 05	CR	2	Visualización de las cosechas de créditos por agencia y asesor	6	Santiago J. López C.
sp2 – 06	CR	2	Visualización de los indicadores de créditos vigentes por agencia y asesor	4	Santiago J. López C.
sp2 – 07	CR	2	Visualización de los indicadores de créditos colocados por agencia y asesor	4	Santiago J. López C.

IDENTIFICADOR DE LA HISTORIA	MODULO	SPRINT	TAREA	ESTIMACIÓN	RESPONSABLE
sp2 – 08	CR	2	Privilegios de acceso al módulo de créditos	1	Santiago J. López C.
sp3 – 01	AG	3	Visualización de informes financieros por agencia	4	Santiago J. López C.
sp3 – 02	AG	3	Visualización de indicadores de la SEPS por agencia	7	Santiago J. López C.
sp3 – 03	AG	3	Privilegios de acceso al módulo de agencias	1	Santiago J. López C.
sp4 – 01	DS	4	Obtención del balance social de la cooperativa por agencias	5	Santiago J. López C.
sp4 – 02	DS	4	Privilegios de acceso al módulo de desempeño social	1	Santiago J. López C.
sp5 – 01	RH	5	Listado de los empleados de toda la cooperativa	1	Santiago J. López C.
sp5 – 02	RH	5	Gestión de empleados de la cooperativa	3	Santiago J. López C.
sp5 – 03	RH	5	Visualización del calendario global para mostrar las planificaciones de vacaciones	3	Santiago J. López C.
sp5 – 04	RH	5	Calculo automático de vacaciones por empleado	2	Santiago J. López C.
sp5 – 05	RH	5	Gestión de permisos pedidos por el empleado	1	Santiago J. López C.
sp5 – 06	RH	5	Ingreso de planificaciones de permisos por empleado	2	Santiago J. López C.
sp5 – 07	RH	5	Visualización del calendario de planificaciones por empleado	2	Santiago J. López C.
sp5 – 08	RH	5	Culminaciones de periodos de vacaciones de empleados	2	Santiago J. López C.

IDENTIFICADOR DE LA HISTORIA	MODULO	SPRINT	TAREA	ESTIMACIÓN	RESPONSABLE
sp5 – 09	RH	5	Visualización y guardado de historiales de permisos por empleados	3	Santiago J. López C.
sp5 – 10	RH	5	Privilegios de acceso de los usuarios de recursos humanos	1	Santiago J. López C.

Los módulos detallados en la tabla son los siguientes:

- TD: Todos los módulos.
- CJ: Módulo Créditos Judiciales
- CR: Módulo de Créditos
- AG: Módulo de Agencias.
- DS: Módulo de Desempeño Social
- RH: Módulo de Recursos Humanos

Planificación de los Sprints

Para realizar la planificación de los sprints se elaboró una tabla donde nos permitirá conocer las tareas que conlleva cada sprint, la fecha de inicio y finalización del sprint, y el estado en el que se va encontrando mientras se avanza en el sprint.

TABLA 8: Formato para redactar los sprints

PILA DEL SPRINT (SPRINT BACKLOG)				
SPRINT#				
FECHA DE INICIO:				
FECHA DE CULMINACIÓN:				
			TAREAS PENDIENTES:	
			DIAS PENDIENTES:	
PRIORIDAD DEL NEGOCIO	DESCRIPCIÓN DE LA TAREA	RESPONSABLE	DURACION DE DIAS	ESTADO

Recursos y costos asociados al desarrollo de los sprints

El presente al ser un trabajo realizado para la titulación de grado, los recursos para el desarrollo son el estudiante que elabora el proyecto y un computador que se utiliza para

desarrollarlo, no se tomó en cuenta los recursos de agua, luz, teléfono, etc. ya que son costos que no influenciaron en el desarrollo del proyecto.

TABLA 9: Recursos y costos para el proyecto

Recurso	Tipo	Inicial	Capacidad	Tasa
Desarrollador	Trabajo	D	100%	\$6.00/hora
Macbook pro	Trabajo	C	100%	\$0.50/hora

2.2. Sprint 1

Planificación

En el sprint 1 se realizará las tareas iniciales del proyecto como el control de autenticación de usuarios con credenciales de la base de datos y abarcará el desarrollo del primer módulo que se encargará de gestionar el proceso de los créditos en estado de etapa judicial, el objetivo de la herramienta es automatizar la información para el manejo eficiente y cumplimiento de los procesos establecidos para los créditos en juicio contribuyendo directamente al monitoreo del área de cobranza. La herramienta automatiza información de la base de datos del Sistema Contable VIMASIF e información externa proveniente del estado de los juicios en las consultas de la Función Judicial para el monitoreo a través de semaforización de los procesos judiciales para el manejo eficiente y oportuno de las diligencias del departamento de cobranzas de la institución, cabe recalcar que este módulo manejará únicamente créditos que estén en estado de mora.

El módulo manejará tres tipos de usuarios, los cuales son: Jefatura de Cobranzas, Jefatura de Créditos y Gerentes.

Las Jefatura de créditos tendrá la capacidad de ver un listado de todos los juicios en estado de mora que tenga la cooperativa, esta información será recopilada de la base de datos de la cooperativa, por cada crédito se tendrá la posibilidad de descartar el crédito o enviarlo a juicio asignándolo a un jefe de cobranzas. Este usuario a su vez, podrá gestionar todos los estados en el que puede estar un crédito (Activo, Cancelado, Reestructurado, Insolvencia, Terminado) la aplicación tendrá la capacidad de tratar de manera diferente cada crédito según su estado. Las jefaturas de créditos podrán guardar monitoreos mensuales, un monitoreo es un guardado del estado actual de todos los créditos en un historial mensual después de que las jefaturas hayan validado la información, podrán gestionar abogados para asignarlos a los juicios y podrán gestionar

las etapas y subetapas que puede tener un juicio según el tipo de juicio que sea (Juicios con Bienes, Juicio sin Bienes, Juicios de Insolvencia) como se muestra en las figuras 14 y 15.

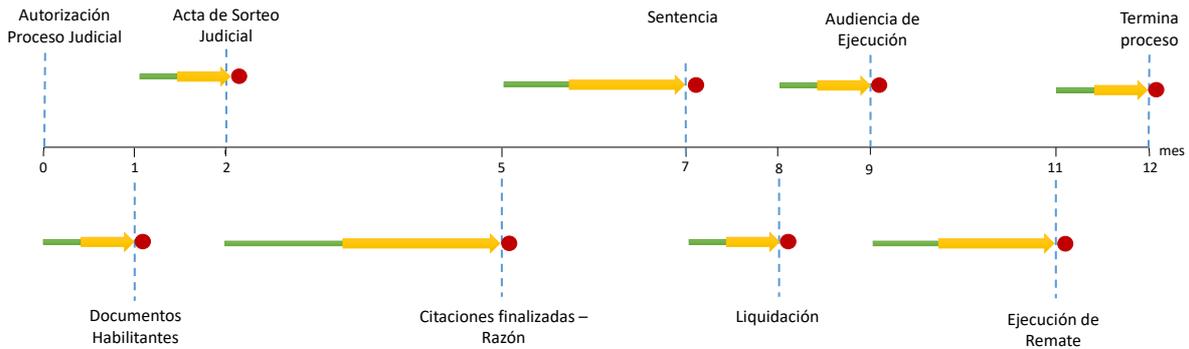


Fig. 14. Proceso Judicial para socios que poseen bienes.

Fuente: [(Cadena & López, 2018a)]

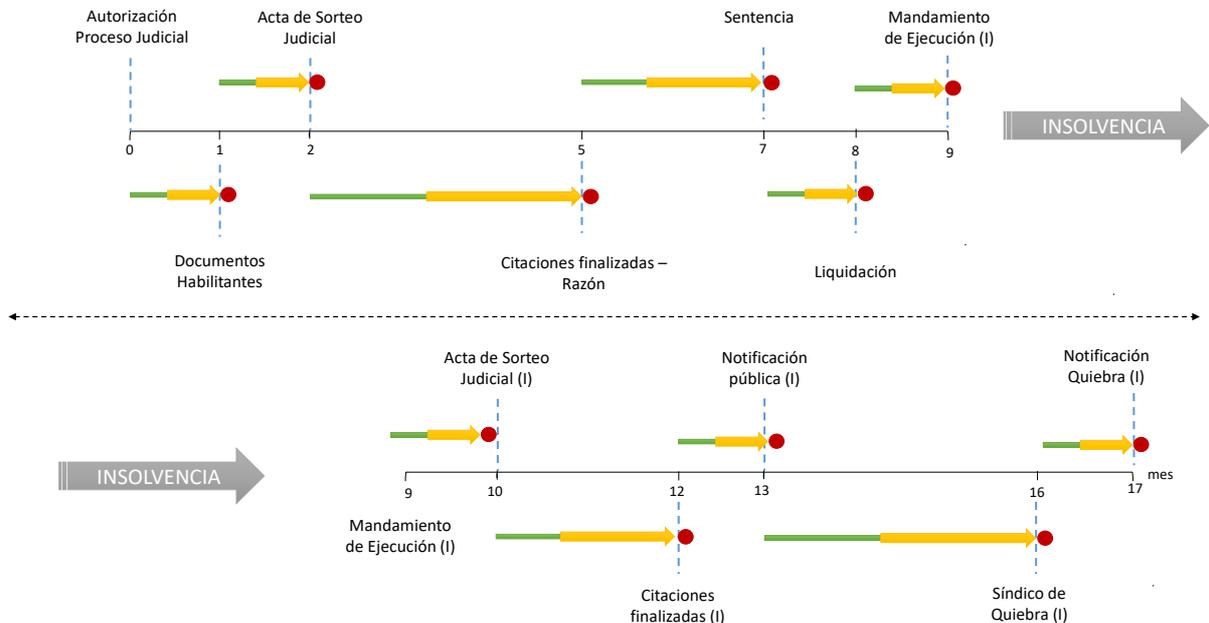


Fig. 15. Proceso Judicial para socios que no poseen bienes e insolvencia

Fuente: [(Cadena & López, 2018a)]

Las jefaturas de cobranzas podrán cambiar los juicios de una etapa a otra únicamente de los juicios asignados al usuario y agregar observaciones, al cambiar la etapa que se encuentra un crédito el algoritmo recalculará el semáforo para mostrarlo a las jefaturas y gerencias.

Las jefaturas de créditos, jefaturas de cobranzas y gerencias tendrán acceso a la evaluación de resultados en donde se mostrarán gráficas de los créditos según su semaforización (Verde, Amarillo, Rojo) junto con listados de los créditos clasificados en dichas categorías, a su vez tendrán acceso a sacar reportes de los monitoreos guardados, para conocer como estuvo un juicio determinado en un mes deseado. Los tres tipos de usuarios podrán visualizar todos los créditos clasificados según su estado (Activo, Cancelado, Reestructurado, Insolvencia, Terminado) con fines de control, y podrán ver la información del crédito recuperada de la base de datos actualizada al día.

La semaforización que manejará el aplicativo se considera como una herramienta de alerta en función del tiempo estimado de cada una de las etapas que forman parte del proceso judicial representado en la figura 16. La semaforización establece un color verde cuando se inicia la etapa hasta el 40% del tiempo que debería tomar la etapa, color amarillo si ha transcurrido la mitad del tiempo hasta un día antes de que finalice el tiempo estimado de la etapa y rojo una vez transcurrido el tiempo estimado máximo de la etapa judicial penalizando el exceso del tiempo estimado para una etapa.

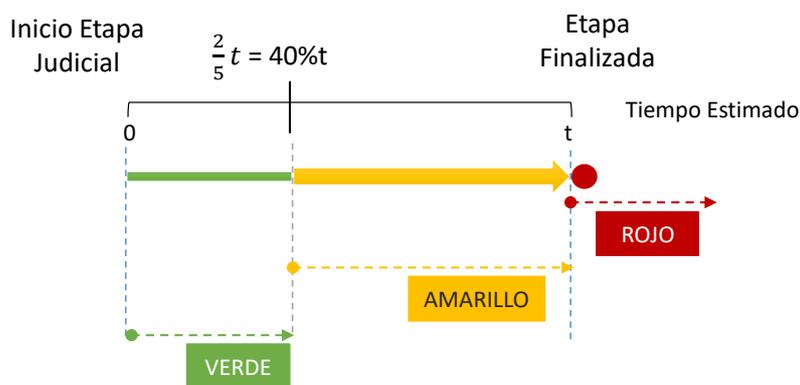


Fig. 16. Sistema de semaforización de la herramienta

Fuente: [(Cadena & López, 2018a)]

Análisis del sprint 1

Historias de usuario de sprint 1

TABLA 10: Historia de usuario 1

HISTORIA DE USUARIO	
NUMERO: sp1 - 01	USUARIO: Todos
NOMBRE HISTORIA: Autenticación de usuarios en el sistema	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 2	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un usuario autorizado para el uso del sistema necesito digitar mis credenciales únicas para poder ingresar y utilizar las funciones del aplicativo.	
COMO PROBARLO: <ul style="list-style-type: none">• Escribir la Url del sistema.• Escribir mi correo electrónico y contraseña.• Ingresar al dashboard de la aplicación exitosamente.	

TABLA 11: Historia de usuario 2

HISTORIA DE USUARIO	
NUMERO: sp1 - 02	USUARIO: Administrador
NOMBRE HISTORIA: Gestión de usuarios y cuentas.	
PRIORIDAD DE NEGOCIO: C	
PUNTOS ESTIMADOS: 1	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un usuario autorizado para el uso del sistema necesito poder cambiar la información de mi cuenta como nombres, apellidos, contraseña y correo electrónico. Como administrador del sistema necesito poder actualizar los datos de un usuario con la finalidad de dar soporte a recuperación de contraseñas.	
COMO PROBARLO: <ul style="list-style-type: none">• Ingresar al sistema.• Ingresar en la opción de mi cuenta.• Ingresar nuevos datos.• Verificar que los datos han sido actualizados.	

TABLA 12: Historia de usuario 3

HISTORIA DE USUARIO	
NUMERO: sp1 - 03	USUARIO: Administrador
NOMBRE HISTORIA: Gestión de usuarios y cuentas.	
PRIORIDAD DE NEGOCIO: C	
PUNTOS ESTIMADOS: 1	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
<p>DESCRIPCIÓN: Como un usuario autorizado para el uso del sistema necesito poder cambiar la información de mi cuenta como nombres, apellidos, contraseña y correo electrónico. Como administrador del sistema necesito poder actualizar los datos de un usuario con la finalidad de dar soporte a recuperación de contraseñas.</p>	
<p>COMO PROBARLO:</p> <ul style="list-style-type: none"> • Ingresar al sistema. • Ingresar en la opción de mi cuenta. • Ingresar nuevos datos. • Verificar que los datos han sido actualizados. 	

TABLA 13: Historia de usuario 4

HISTORIA DE USUARIO	
NUMERO: sp1 - 04	USUARIO: Jefatura de Créditos y Cobranzas
NOMBRE HISTORIA: Autorizar o rechazar juicios	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 2	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
<p>DESCRIPCIÓN: Como un asesor de créditos necesito ingresar al sistema los juicios que cumplan con los requisitos para enviar a juicios o rechazarlos, con la finalidad tener un ingreso automático de los juicios al sistema y que sean controlados por el asesor de cobranzas. Como un asesor de cobranzas necesito visualizar los juicios autorizados a mi usuario para ingresarlos e inicial proceso judicial.</p>	
<p>COMO PROBARLO:</p> <ul style="list-style-type: none"> • Ingresar al sistema • En el menú ingresos ver el listado de juicios pendientes en estado de mora • Ingresar juicio y asignarlo a un jefe de cobranzas • Rechazar un juicio y verlo en la sección de juicios no autorizados • Un jefe de cobranzas puede entrar a la pestaña de nuevos juicios y visualizar los juicios asignados para poder ingresarlos en el sistema. 	

TABLA 14: Historia de usuario 5

HISTORIA DE USUARIO	
NUMERO: sp1 - 05	USUARIO: Jefatura de Créditos
NOMBRE HISTORIA: Visualización de créditos judiciales	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 3	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor de créditos necesito, tener un listado de los créditos Activos, Terminados, Cancelados, Reestructurados, Reingresos o Abandonados y divididos en categoría, con la finalidad de medir el desempeño del departamento de cobranzas.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar al sistema • Ingresar al módulo de cobranzas judiciales • Se debe visualizar los créditos en estado de mora en tres tablas; Créditos con Bienes, Créditos sin Bienes, Créditos de Insolvencia. • En el menú de Juicios se despliegan los nombres de los créditos al ingresar a uno de ellos se debe visualizar únicamente los créditos dentro de dicha categoría. 	

TABLA 15: Historia de usuario 6

HISTORIA DE USUARIO	
NUMERO: sp1 - 06	USUARIO: Todos
NOMBRE HISTORIA: Semaforización y etapas estimadas	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor judicial, asesor de créditos y gerente, necesitamos visualizar un semáforo en colores (Verde, Amarillo, Rojo) del estado del juicio, así como su etapa estimada, con la finalidad de identificar de manera fácil los procesos judiciales de créditos que están retrasados.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar al sistema • Ingresar al módulo de cobranzas judiciales • Visualizar en las tablas de créditos las columnas de semáforo y etapa estimada del crédito. 	

TABLA 16: Historia de usuario 7

HISTORIA DE USUARIO	
NUMERO: sp1 - 07	USUARIO: Jefatura de Créditos
NOMBRE HISTORIA: Gestión de estado de juicios	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 2	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un jefe de créditos necesito, modificar los estados de los juicios (Activo, Terminado, Cancelado, Abandonado, Reestructurado, Insolvencia), con la finalidad de controlar el desempeño del departamento de cobranzas.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar al sistema. • Ingresar al módulo de cobranzas judiciales. • Escoger un crédito específico y entrar a los detalles del juicio. • Cambiar el estado del juicio utilizando los botones con los nombres de los estados. 	

TABLA 17: Historia de usuario 8

HISTORIA DE USUARIO	
NUMERO: sp1 - 08	USUARIO: Todos
NOMBRE HISTORIA: Detalles de los juicios	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 3	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor de créditos, asesor judicial y gerente, necesitamos visualizar los datos del crédito, datos del socio, saldos, datos judiciales, datos de los garantes de un juicio, con la finalidad de tener acceso rápido a la información detallada del juicio y no tener que acceder a la base de datos Oracle para consultarlos.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar al módulo de cobranzas judiciales • En la lista de juicios presionar el botón para ver los detalles del juicio. 	

TABLA 18: Historia de usuario 9

HISTORIA DE USUARIO	
NUMERO: sp1 - 09	USUARIO: Todos
NOMBRE HISTORIA: Gráfica de procesos de juicios	
PRIORIDAD DE NEGOCIO: C	
PUNTOS ESTIMADOS: 1	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor de créditos, asesor de créditos y gerentes, necesitamos visualizar una gráfica de avance de cada juicio, con la finalidad de conocer visualmente las etapas restantes del juicio.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar al módulo de cobranzas judiciales • En la lista de juicios presionar el botón para ver los detalles del juicio. 	

TABLA 19: Historia de usuario 10

HISTORIA DE USUARIO	
NUMERO: sp1 - 10	USUARIO: Jefatura de Créditos
NOMBRE HISTORIA: Gestión de abogados	
PRIORIDAD DE NEGOCIO: S	
PUNTOS ESTIMADOS: 1	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor de créditos, necesito poder gestionar los abogados de la aplicación, con la finalidad de tener un listado de abogados con sus datos para poder asignarlos a los juicios.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar al módulo de cobranzas judiciales. • Ingresar a la pestaña Gestionar y luego en Jueces. • Visualizar el listado de abogados. • Hacer click en cualquier campo que se quiera modificar para mostrar la opción de modificación. • Ingresar un abogado nuevo mediante el formulario de ingreso. 	

TABLA 20: Historia de usuario 11

HISTORIA DE USUARIO	
NUMERO: sp1 - 11	USUARIO: Jefatura de Créditos
NOMBRE HISTORIA: Gestión de actividades y sub actividades de juicios	
PRIORIDAD DE NEGOCIO: S	
PUNTOS ESTIMADOS: 2	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como una jefatura de créditos necesito, ingresar subactividades a cualquier etapa de los juicios, con la finalidad de que el asesor de cobranzas pueda asignar sub actividades a los juicios.	
COMO PROBARLO: <ul style="list-style-type: none"> Ingresar al módulo de cobranzas judiciales. Ingresar a la pestaña Gestionar y luego en Etapas. Seleccionar la etapa y hacer click en el botón ingresar proceso. Se visualiza el formulario de ingreso de sub actividades. 	

TABLA 21: Historia de usuario 12

HISTORIA DE USUARIO	
NUMERO: sp1 - 12	USUARIO: Jefatura de Créditos
NOMBRE HISTORIA: Monitoreo	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor de créditos, necesito guardar un monitoreo de créditos judiciales mensual a inicios de cada mes, con la finalidad de guardar el historial de semaforización para reportes.	
COMO PROBARLO: <ul style="list-style-type: none"> Ingresar al módulo de cobranzas judiciales. Ingresar a la pestaña de Monitoreo. Utilizar el botón de guardar monitoreo y confirmar la acción. 	

TABLA 22: Historia de usuario 13

HISTORIA DE USUARIO	
NUMERO: sp1 - 13	USUARIO: Jefatura de Créditos
NOMBRE HISTORIA: Pre visualización del monitoreo	
PRIORIDAD DE NEGOCIO: C	
PUNTOS ESTIMADOS: 2	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor de créditos, necesito pre visualizar los datos que se van a guardar en el monitoreo, con la finalidad de realizar correcciones a los juicios y validar la información antes de ser guardada.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar al módulo de cobranzas judiciales. • Ingresar a la pestaña de Monitoreo. • Visualizar la tabla en donde se encuentran los juicios junto con los semáforos que se van a grabar en el monitoreo del mes. 	

TABLA 23: Historia de usuario 14

HISTORIA DE USUARIO	
NUMERO: sp1 - 14	USUARIO: Todos
NOMBRE HISTORIA: Evaluación de resultados	
PRIORIDAD DE NEGOCIO: S	
PUNTOS ESTIMADOS: 2	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor de créditos, asesor de cobranzas y gerencias, necesitamos poder ver gráficas de resumen de los semáforos de los créditos y tablas categorizadas por semáforos, con la finalidad de poder evaluar el rendimiento del departamento de cobranzas.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar al módulo de cobranzas judiciales. • Ingresar a la pestaña Resultados. • Se visualizarán las tablas y gráficas de créditos judiciales. 	

TABLA 24: Historia de usuario 15

HISTORIA DE USUARIO	
NUMERO: sp1 - 15	USUARIO: Todos
NOMBRE HISTORIA: Buscador de juicios	
PRIORIDAD DE NEGOCIO: W	
PUNTOS ESTIMADOS: 1	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor de créditos, asesor de cobranzas, gerencias y asesore, necesitamos poder buscar juicios por su: id de crédito, id de socio, cédula o nombres del socio, con la finalidad de tener acceso rápido a los juicios.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar al módulo de cobranzas judiciales. • En el menú en el formulario de buscar ingresar cualquiera de la siguiente información de un juicio: id de crédito, id de socio, cédula o nombres del socio. • En los resultados mostrará todos los juicios que cumplan con las condiciones ingresadas. 	

TABLA 25: Historia de usuario 16

HISTORIA DE USUARIO	
NUMERO: sp1 - 16	USUARIO: Todos
NOMBRE HISTORIA: Reporte de semaforización	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor de créditos, asesor de cobranzas, gerencias y asesores judiciales, necesitamos poder sacar reportes de los historiales guardados, así como porcentajes de semaforización por mes de reporte, con la finalidad de evaluar los avances de los juicios en el tiempo.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar al módulo de cobranzas judiciales. • Desplegar el formulario de reporte de monitoreo mediante el botón. • Ingresar el rango de fechas del reporte deseado • Visualizar el reporte de monitoreo. 	

TABLA 26: Historia de usuario 17

HISTORIA DE USUARIO	
NUMERO: sp1 - 17	USUARIO: Jefatura de Créditos
NOMBRE HISTORIA: Convertir juicios de bienes a sin bienes y viceversa.	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 3	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor de créditos, necesito poder reingresar juicios, reestructurar juicios y convertir un juicio de bienes a sin bienes y viceversa, con la finalidad de tener un completo control de los juicios para eventualidades que sucedan con los clientes.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar al módulo de cobranzas judiciales. • Ingresar a los detalles del juicio. • Convertir el juicio de bienes a sin bienes y viceversa mediante el botón con dicho nombre. 	

TABLA 27: Historia de usuario 18

HISTORIA DE USUARIO	
NUMERO: sp1 - 18	USUARIO: Jefatura de cobranzas
NOMBRE HISTORIA: Gestión de etapas judiciales	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor judicial necesito actualizar las etapas de cada juicio, con la finalidad de calcular los semáforos y tener un control del proceso judicial de cada juicio por parte de cada departamento.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar al módulo de cobranzas judiciales. • Hacer click en la columna de etapa actual de un determinado juicio para desplegar las etapas disponibles. • Seleccionar una y visualizar el cálculo del semáforo. 	

TABLA 28: Historia de usuario 19

HISTORIA DE USUARIO	
NUMERO: sp1 - 19	USUARIO: Jefatura de Créditos
NOMBRE HISTORIA: Asignación de juicios a los asesores	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 4	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un jefe de créditos necesito poder ingresar los nuevos juicios y asígnalos a un asesor de cobranzas con la finalidad que los asesores de cobranzas puedan visualizar y gestionar únicamente los juicios que se encuentran asignados a él.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar al módulo de cobranzas judiciales. • Ingresar a la pestaña de Ingresos. • Del listado escoger el crédito a ingresar mediante el botón Autorizar y seleccionar de la lista el asesor al que se le asignará el juicio. 	

TABLA 29: Historia de usuario 20

HISTORIA DE USUARIO	
NUMERO: sp1 - 20	USUARIO: Jefatura de cobranzas
NOMBRE HISTORIA: Clasificación del proceso de los juicios.	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor judicial necesito poder seleccionar si un nuevo proceso judicial asignado a mi usuario es un proceso judicial con bienes o sin bienes, con la finalidad de establecer las diferentes etapas que debe cumplir el juicio.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar al módulo de cobranzas judiciales. • Ingresar a la pestaña nuevos juicios. • Ingresar un juicio asignado al usuario. • En el formulario seleccionar el tipo de juicio (Bienes o sin Bienes) según sea el caso. 	

TABLA 30: Historia de usuario 22

HISTORIA DE USUARIO	
NUMERO: sp1 - 22	USUARIO: Gerentes
NOMBRE HISTORIA: Eliminación de monitoreos.	
PRIORIDAD DE NEGOCIO: S	
PUNTOS ESTIMADOS: 3	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como gerente de la cooperativa, necesito poder eliminar monitores mensuales que se hayan guardado, con la finalidad de eliminar datos guardados con errores para realizar correcciones.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar al módulo de cobranzas judiciales. • En el panel de Eliminación de Monitoreo escoger el monitoreo a eliminar. 	

TABLA 31: Historia de usuario 23

HISTORIA DE USUARIO	
NUMERO: sp1 - 23	USUARIO: Administrador
NOMBRE HISTORIA: Gestión de las tablas como administrador.	
PRIORIDAD DE NEGOCIO: C	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 1
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un administrador del sistema, necesito tener acceso a las tablas implicadas en el módulo de cobranzas judiciales para corrección de errores o manejo de la información.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar sistema • Hacer click en el panel de administración. • Ver el listado de las tablas de la plataforma para poder gestionarlas. 	

Diseño y codificación del sprint

Diagrama de la base de datos

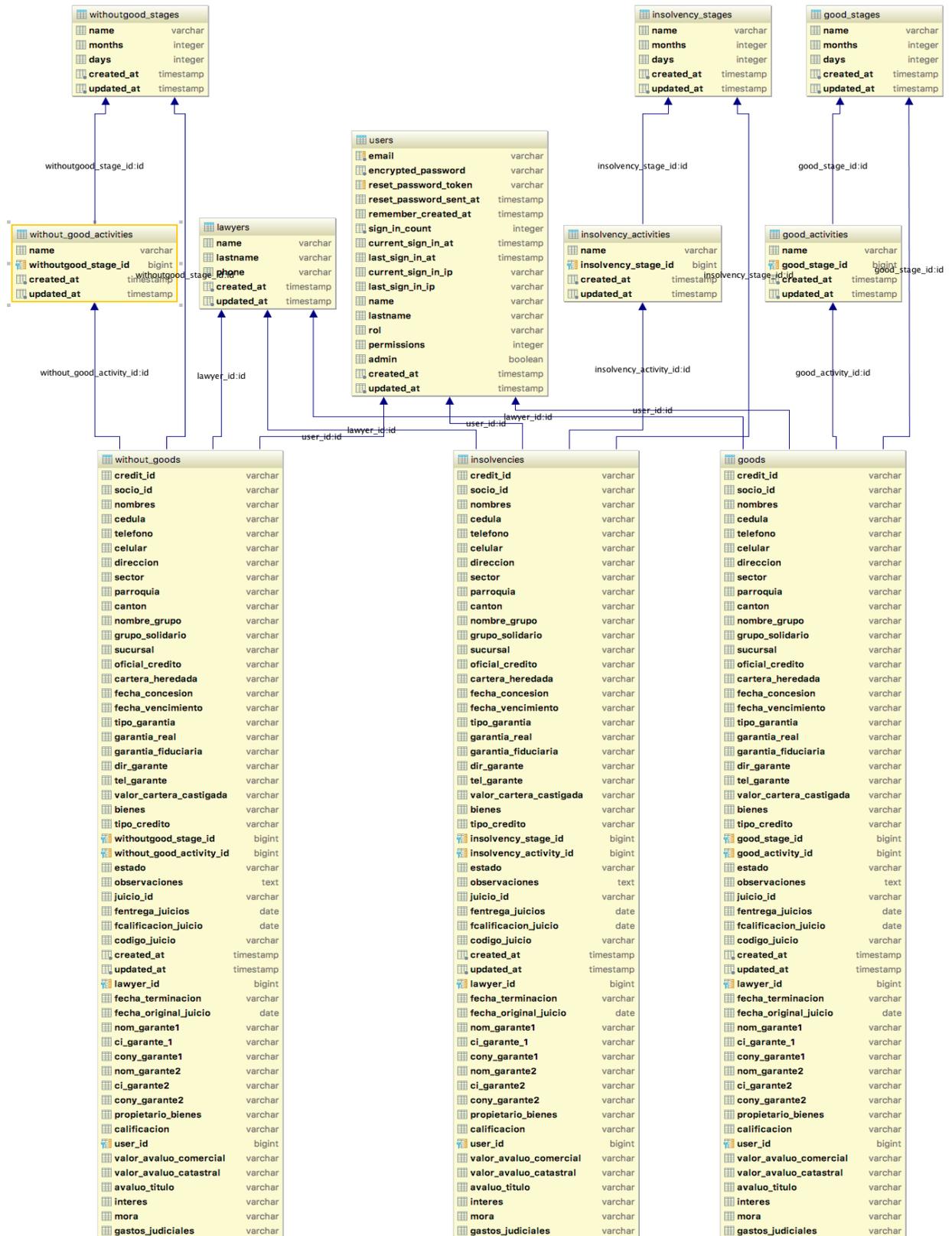


Fig. 17. Diagrama de la base de datos del módulo de cobranzas judiciales

Dependencias del módulo

Listado de gemas de Ruby y dependencias necesarias para el desarrollo del sprint.

- ❖ gem 'rails' version: 5.1.3
- ❖ gem 'sqlite3'
- ❖ gem 'puma', version: 3.7
- ❖ gem 'sass-rails', version: 5.0
- ❖ gem 'webpacker'
- ❖ gem 'webpacker-react', version: 0.2.0
- ❖ gem 'coffee-rails', version: 4.2
- ❖ gem 'turbolinks', version: 5
- ❖ gem 'jbuilder', version: 2.5
- ❖ gem 'devise'
- ❖ gem 'haml-rails'
- ❖ gem 'annotate'
- ❖ gem 'best_in_place', version: 3.1
- ❖ gem 'jquery-rails', version: 4.1
- ❖ gem 'pg'
- ❖ gem 'filterrific'
- ❖ gem "chartkick"
- ❖ gem 'searchkick'
- ❖ gem 'rails_admin', version: 1.2
- ❖ gem 'activerecord-oracle_enhanced-adapter', version: 1.8.1
- ❖ gem 'ruby-oci8'
- ❖ gem 'whenever'
- ❖ gem 'therubyracer'
- ❖ gem 'execjs'
- ❖ gem 'capistrano', version: 3.7
- ❖ gem 'capistrano-rails', version: 1.2
- ❖ gem 'capistrano-passenger', version: 0.2.0
- ❖ gem 'capistrano-rbenv', version: 2.1
- ❖ gem 'capistrano-rails-console'
- ❖ gem 'react-rails'

Producto entregable del sprint

Cumplido el desarrollo y realizado las pruebas de funcionamiento correspondientes se entrega el módulo de cobranzas judiciales terminado y funcionando en el servidor de la cooperativa, a continuación, se detalla el sprint backlog final y las pantallas finales del módulo de cobranzas judiciales.

Sprint backlog final

TABLA 32: Sprint Backlog final del spint 1

PILA DEL SPRINT (SPRINT BACKLOG)					
SPRINT					
FECHA DE INICIO: 27 noviembre 2017					
FECHA DE CULMINACIÓN: 27 enero 2018					
				TAREAS PENDIENTES:	0
				DIAS PENDIENTES:	0
PRIORIDAD DEL NEGOCIO	DESCRIPCIÓN DE LA TAREA	RESPONSABLE	DURACIÓN DE DIAS	ESTADO	
ALTA	Autenticación de Usuarios en el Sistema	Santiago López	2	Terminado	
MEDIA	Gestión de Usuarios y Cuentas.	Santiago López	1	Terminado	
ALTA	Visualización de un Dashboard de los módulos del Sistema	Santiago López	1	Terminado	
ALTA	Autorizar o Rechazar Juicios	Santiago López	1	Terminado	
ALTA	Visualización de los créditos judiciales	Santiago López	3	Terminado	
ALTA	Semaforización y Etapas Estimadas	Santiago López	5	Terminado	
ALTA	Gestión del Estado de Juicios	Santiago López	2	Terminado	
MEDIA	Detalles de los Juicios	Santiago López	3	Terminado	
BAJA	Gráfica de procesos de Juicios	Santiago López	1	Terminado	
MEDIA	Gestión de los abogados	Santiago López	1	Terminado	
MEDIA	Gestión de las subactividades de Juicios	Santiago López	2	Terminado	
ALTA	Monitoreos	Santiago López	6	Terminado	
BAJA	Previsualización del monitoreo	Santiago López	2	Terminado	
ALTA	Visualización de los resultados	Santiago López	2	Terminado	
BAJA	Buscador de Juicios	Santiago López	1	Terminado	
ALTA	Reportes de los monitoreos	Santiago López	5	Terminado	
ALTA	Convertir Juicios de bienes a sin bienes y viceversa.	Santiago López	3	Terminado	
ALTA	Gestión de las etapas judiciales	Santiago López	2	Terminado	
ALTA	Asignación de Juicios a los asesores	Santiago López	2	Terminado	
ALTA	Clasificación de los procesos de los Juicios	Santiago López	3	Terminado	
MEDIA	Privilegio de acceso a los asesores y de facturas de cobranzas	Santiago López	4	Terminado	
ALTA	Eliminación del monitoreo	Santiago López	3	Terminado	
MEDIA	Gestión de las tablas como administrador.	Santiago López	6	Terminado	

Capturas de pantalla del módulo terminado



Fig. 18. Login del aplicativo



Fig. 19. Dashboard de la aplicación

Usuario - Jefaturas de Cobranzas

Historial de semaforización

Juicios con Bienes

Nombre	Id Socio	Id Juicio	Cédula	Etapa Actual	Proceso	Estado	Semaforo	Acciones
[Redacted]	12729		[Redacted]	Acta de sorteo judicial	-	Reingreso	amarillo	Ver
[Redacted]	46784		[Redacted]	Documentos habilitantes	-	Activo	rojo	Ver
[Redacted]	13157	10333-2017-01639	[Redacted]	Sentencia	-	Activo	verde	Ver
[Redacted]	16007	10333-2015-01910	[Redacted]	Acta de sorteo judicial	-	Reingreso	rojo	Ver
[Redacted]	49221	10333-2016-01405	[Redacted]	Sentencia	-	Activo	verde	Ver
[Redacted]	51798	10333-2016-01995/10311-2016-00209G	[Redacted]	Acta de sorteo judicial	-	Activo	rojo	Ver
[Redacted]	52779	10333-2016-01599	[Redacted]	Acta de sorteo judicial	-	Reingreso	rojo	Ver
[Redacted]	44293	10333-2015-01449	[Redacted]	Acta de sorteo judicial	-	Activo	rojo	Ver
[Redacted]	49330	10333-2015-0367	[Redacted]	Acta de sorteo judicial	-	Activo	rojo	Ver

Fig. 20. Listado de juicios asignados al usuario

Historial de semaforización

Juicios con Bienes

Nombre	Id Socio	Id Juicio	Cédula	Etapa Actual	Proceso	Estado	Semaforo	Acciones
[Redacted]	12729		[Redacted]	Acta de sorteo judicial	-	Reingreso	amarillo	Ver
[Redacted]	46784		[Redacted]	Documentos habilitantes	-	Activo	rojo	Ver
[Redacted]	13157	10333-2017-01639	[Redacted]	Sentencia	-	Activo	verde	Ver
[Redacted]	16007	10333-2015-01910	[Redacted]	Acta de sorteo judicial	-	Reingreso	rojo	Ver
[Redacted]	49221	10333-2016-01405	[Redacted]	Sentencia	-	Activo	verde	Ver
[Redacted]	51798	10333-2016-01995/10311-2016-00209G	[Redacted]	Acta de sorteo judicial	-	Activo	rojo	Ver

- Autorización proceso judicial
- Documentos habilitantes
- ✓ Acta de sorteo judicial
- Citaciones finalizadas - Razón
- Sentencia
- Liquidación
- Audiencia de ejecución
- Ejecución de remate
- Termina proceso

Fig. 21. Gestión de estado de juicios



Fig. 22. Datos de un crédito judicial



Fig. 23. Evaluación de resultados

Inicio													
Dashboard / Home / Nuevos Juicios		Buscar		Buscar		Juicios		Nuevos Juicios		Resultados		Cuenta	
<h1>Juicios pendientes</h1>													
Id Credito	Id Socio	Nombres	Cedula	Nombre grupo	Grupo solidario	Oficial credito	Bienes	Tipo credito	Ver	Ingresar			
67843	4534525	Jose	1789387564						Ver	Ingresar			
5684	58372	Tatiana	10040673658	Nombre Grupo		Nelson	T,C,v	Solidario	Ver	Ingresar			

Fig. 24. Juicios Pendientes de ingreso

Inicio													
Dashboard / Home / Nuevos Juicios		Buscar		Buscar		Juicios		Nuevos Juicios		Resultados		Cuenta	
<h2>Datos del Credito</h2>													
Credit			Nombre grupo			Tiene bienes ?							
<input type="text" value="5684"/>			<input type="text" value="Nombre Grupo"/>			<input type="button" value="Si tiene bienes"/>							
Socio			Grupo solidario			Abogado							
<input type="text" value="58372"/>			<input type="text"/>			<input type="button" value="Please select"/>							
Seleccione los bienes						Poseedor de los bienes							
<input type="checkbox"/> Casa <input type="checkbox"/> Terreno <input type="checkbox"/> Vehículo						<input type="checkbox"/> Socio <input type="checkbox"/> Garante 1 <input type="checkbox"/> Garante 2							
<h2>Datos de la Socia(o)</h2>													
Nombres						Direccion							
<input type="text" value="Tatiana"/>						<input type="text" value="calle 345"/>							
Cedula						Sector							
<input type="text" value="10040673658"/>						<input type="text" value="E"/>							
Telefono						Pasaporta							
<input type="text"/>						<input type="text"/>							

Fig. 25. Ingreso de nuevos juicios

Inicio **Buscar** **Juicios** **Nuevos Juicios** **Resultados** **Cuenta**

Datos Extras

Oficial credito <input type="text" value="Nelson"/>	Tipo garantia <input type="text" value="Solidaria"/>	Dir garante <input type="text" value="Calle 34"/>
Cartera heredada <input type="text" value="Romel"/>	Garantia real <input type="text" value="Garantia Real"/>	Tel garante <input type="text"/>
Fecha concesion <input type="text" value="H 02-06-2016"/>	Garantia fiduciaria <input type="text" value="Garantia fiduciaria"/>	Valor cartera castigada <input type="text"/>
Fecha vencimiento <input type="text" value="02-06-2016"/>	Tipo credito <input type="text" value="Solidario"/>	

Nombres garante 1 <input type="text"/>	Nombres garante 2 <input type="text"/>
Cedula garante 1 <input type="text"/>	Cedula garante 2 <input type="text"/>
Cony garante1 <input type="text"/>	Cony garante2 <input type="text"/>

Fig. 26. Ingreso de nuevos juicios

Inicio **Buscar** **Juicios** **Nuevos Juicios** **Resultados** **Cuenta**

Dashboard Home

Historial de semaforización

Fecha de inicio

Fecha de fin

Visualizar

Juicios con Bienes

Nombre	Id Socio	Id Juicio	Cédula	Etapa Actual	Proceso	Estado	Semaforo	Acciones
[Redacted]	12729		[Redacted]	Acta de sorteo judicial	-	Reingreso	amarillo	Ver
[Redacted]	46784		[Redacted]	Documentos habilitantes	-	Activo	rojo	Ver
[Redacted]	13157	10333-2017-01639	[Redacted]	Sentencia	-	Activo	verde	Ver
[Redacted]	16007	10333-2015-	[Redacted]	Acta de sorteo	-	Reingreso	rojo	Ver

Fig. 27. Formulario de consulta de monitoreos

Inicio **Juicios** **Nuevos Juicios**² Resultados Cuenta

Dashboard / Home / Historial de semaforos

Reporte de juicios

Abogados

Agencias

Asesores

Asesor C.

Ver	Cedula	Id Crédito	Id Socio	Estado	12-2017	01-2018
<input type="button" value="Ver"/>	402094494	31962	49193	Activo	rojo	-

Fig. 28. Reporte de monitoreo de juicios

Inicio **Juicios** **Nuevos Juicios**² Resultados Cuenta

Ver	Cedula	Id Crédito	Id Socio	Estado	12-2017	01-2018
<input type="button" value="Ver"/>	██████████	31962	49193	Activo	rojo	-
<input type="button" value="Ver"/>	██████████	29518	49223	Activo	rojo	-
<input type="button" value="Ver"/>	██████████	38926	11159	Activo	amarillo	-
<input type="button" value="Ver"/>	██████████	36022	52373	Reingreso	amarillo	-
<input type="button" value="Ver"/>	██████████	33621	53867	Activo	amarillo	-
<input type="button" value="Ver"/>	██████████	31470	52708	Activo	amarillo	-
<input type="button" value="Ver"/>	██████████	34135	52404	Reingreso	amarillo	-
<input type="button" value="Ver"/>	██████████	28657	15142	Activo	terminado	-
<input type="button" value="Ver"/>	██████████	27480	48759	Activo	terminado	-
<input type="button" value="Ver"/>	██████████	21221	43529	Activo	amarillo	-
<input type="button" value="Ver"/>	██████████	28396	49192	Activo	rojo	-
<input type="button" value="Ver"/>	██████████	23181	11024	Activo	amarillo	-
<input type="button" value="Ver"/>	██████████	30820	52224	Activo	amarillo	-

Fig. 29. Reporte de monitoreo de juicios



Fig. 30. Reporte de monitoreo de juicios

Usuario: Supervisión y Validación – Jefatura de Crédito

Inicio | Buscar | **Buscar** | Juicios | Gestionar | Monitoreo | Ingresos | Resultados | Cuenta

Dashboard / Home

Historial de semaforización

Juicios con Bienes

Nombre	Id Crédito	Id Socio	Cédula	Etapa Actual	Etapa Estimada	Estado	Semaforo	Ver			
[Blurred]	24961	13432	[Blurred]	Acta de sorteo judicial	Citaciones finalizadas - Razón	Reestructurado	rojo	Ver	Terminar	Reingresar	
[Blurred]	32866	46784	[Blurred]	Citaciones finalizadas - Razón	Citaciones finalizadas - Razón	Activo	verde	Ver	Terminar	Reingresar	
[Blurred]	31048	12729	[Blurred]	Acta de sorteo judicial	Citaciones finalizadas - Razón	Reestructurado	rojo	Ver	Terminar	Reingresar	
[Blurred]	6346	12671	[Blurred]	Citaciones finalizadas - Razón	Citaciones finalizadas - Razón	Activo	verde	Ver	Terminar	Reingresar	
[Blurred]	32152	45776	[Blurred]	Acta de sorteo judicial	Citaciones finalizadas - Razón	Reestructurado	rojo	Ver	Terminar	Reingresar	
[Blurred]	6985	10257	[Blurred]	Acta de sorteo judicial	Citaciones finalizadas - Razón	Reingreso	rojo	Ver	Terminar		
[Blurred]	21221	43529	[Blurred]	Citaciones finalizadas - Razón	Citaciones finalizadas - Razón	Activo	amarillo	Ver	Terminar	Reingresar	
[Blurred]	25688	40449	[Blurred]	Citaciones finalizadas - Razón	Citaciones finalizadas - Razón	Activo	amarillo	Ver	Terminar	Reingresar	
[Blurred]	25323	13157	[Blurred]	Citaciones finalizadas - Razón	Citaciones finalizadas - Razón	Activo	amarillo	Ver	Terminar	Reingresar	

Fig. 31. Pantalla principal Jefatura de Crédito

Inicio **Juicios** **Monitoreo** **Ingresos** **Resultados** **Cuenta**

Bienvenid@ Santy Lopez

Instrucciones

Estados de los juicios con bienes

Nombre Etapa	Mes	Ingresar proceso
Autorización proceso judicial	0	<input type="button" value="Ingresar proceso"/>
Documentos habilitantes	1	<input type="button" value="Ingresar proceso"/>
Acta de sorteo judicial	2	<input type="button" value="Ingresar proceso"/>
Sentencia	7	<input type="button" value="Ingresar proceso"/>
Liquidación	8	<input type="button" value="Ingresar proceso"/>
Audiencia de ejecución	9	<input type="button" value="Ingresar proceso"/>
Ejecución de remate	11	<input type="button" value="Ingresar proceso"/>
Termina proceso	12	<input type="button" value="Ingresar proceso"/>
Citaciones finalizadas - Razón	5	<input type="button" value="Ingresar proceso"/>

Estados de los juicios sin bienes

Nombre Etapa	Mes	Ingresar proceso
Autorización proceso judicial	0	<input type="button" value="Ingresar proceso"/>

192.100.1.53/good_activities/new?good_stage=4

Fig. 32. Gestión de Actividades y Etapas

Inicio **Juicios** **Monitoreo** **Ingresos** **Resultados** **Cuenta**

Ingresar Proceso

Nombre

Actividades actuales

Avaluo pericial
Publicación en el portal de la Judicatura el remate
Remisión de Posturas
Audiencia de Remate
Adjudicación del BIEN Y ENTREGA DE DINERO

Fig. 33. Gestión de Actividades y Etapas



Fig. 34. Guardar monitoreos mensuales y previsualización de monitoreos



Fig. 35. Evaluación de resultados usuario de Jefatura de Créditos

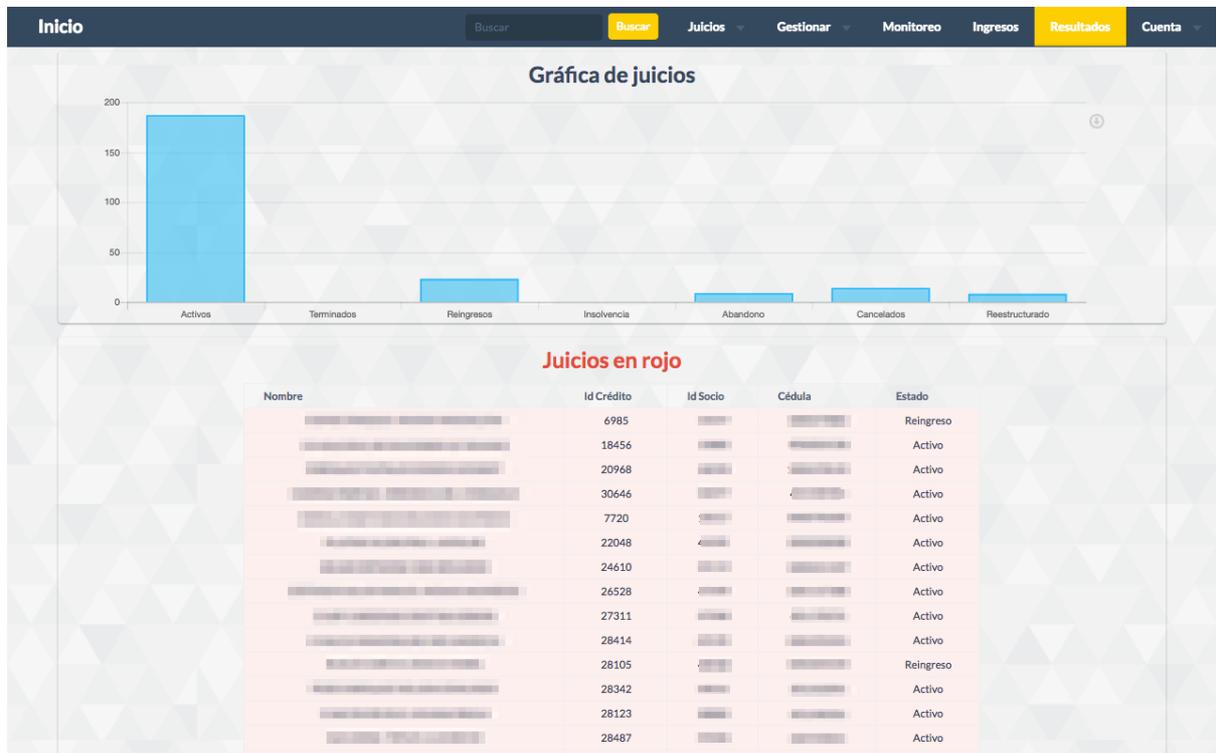


Fig. 36. Evaluación de resultados usuario de Jefatura de Créditos

Juicios en amarillo

Nombre	Id Crédito	Id Socio	Cédula	Estado
[Redacted]	21388	[Redacted]	[Redacted]	Activo
[Redacted]	21221	[Redacted]	[Redacted]	Activo
[Redacted]	6986	[Redacted]	[Redacted]	Activo
[Redacted]	24662	[Redacted]	[Redacted]	Activo
[Redacted]	26886	[Redacted]	[Redacted]	Activo
[Redacted]	24685	[Redacted]	[Redacted]	Activo
[Redacted]	25688	[Redacted]	[Redacted]	Activo
[Redacted]	25323	[Redacted]	[Redacted]	Activo
[Redacted]	21969	[Redacted]	[Redacted]	Activo
[Redacted]	28501	[Redacted]	[Redacted]	Activo
[Redacted]	28335	[Redacted]	[Redacted]	Reingreso
[Redacted]	28338	[Redacted]	[Redacted]	Reingreso
[Redacted]	28484	[Redacted]	[Redacted]	Reingreso
[Redacted]	28494	[Redacted]	[Redacted]	Reingreso
[Redacted]	29255	[Redacted]	[Redacted]	Activo
[Redacted]	30451	[Redacted]	[Redacted]	Activo
[Redacted]	29725	[Redacted]	[Redacted]	Reingreso
[Redacted]	29025	[Redacted]	[Redacted]	Activo
[Redacted]	29065	[Redacted]	[Redacted]	Reingreso
[Redacted]	30761	[Redacted]	[Redacted]	Activo
[Redacted]	30927	[Redacted]	[Redacted]	Activo
[Redacted]	30652	[Redacted]	[Redacted]	Activo
[Redacted]	30689	[Redacted]	[Redacted]	Activo
[Redacted]	33621	[Redacted]	[Redacted]	Activo
[Redacted]	32011	[Redacted]	[Redacted]	Activo
[Redacted]	32541	[Redacted]	[Redacted]	Activo
[Redacted]	29220	[Redacted]	[Redacted]	Activo

Fig. 37. Evaluación de resultados usuario de Jefatura de Créditos

Inicio Juicios Gestionar Monitoreo **Ingresos** Resultados Cuenta

Ingreso de nuevos juicios

Id Credito	Id Socio	Nombres	Saldo Total	Monto Real	Oficina	Dias de Mora	Tipo de credito	Autorizar	Descartar
7845	898769	Martin	8493.0	7483.0		180		<input type="button" value="Autorizar"/>	<input type="button" value="Descartar"/>
95837	3545	Jose Luis R.	12300.0	30000.0		105	Productivo	<input type="button" value="Autorizar"/>	<input type="button" value="Descartar"/>
5765346	93765	Jaqueline Na	1200.0	4000.0		100	Productivo	<input type="button" value="Autorizar"/>	<input type="button" value="Descartar"/>

Créditos castigados

Id Credito	Id Socio	Nombres	Saldo Total	Monto Real	Oficina	Dias de Mora	Tipo de credito	Autorizar	Descartar
6453	46453	Josue	1.0	34543.0		110		<input type="button" value="Autorizar"/>	<input type="button" value="Descartar"/>

Juicios descartados

Fig. 38. Gestión de nuevos juicios

Inicio Juicios Gestionar Monitoreo **Ingresos** Resultados Cuenta

Asignación del asesor

Selecciona el asesor que será responsable de este juicio.

▼

-
-
-

Id Credito	Id Socio	Nombres	Saldo Total	Monto Real	Oficina	Dias de Mora	Tipo de credito	Autorizar	Descartar
17367	40768	NAVARRETE DIAZ LILIANA						<input type="button" value="Autorizar"/>	<input type="button" value="Descartar"/>
708	10623	OSBERNOS MARIANGELA ALVARO						<input type="button" value="Autorizar"/>	<input type="button" value="Descartar"/>
17449	46077	MUNIZ PAREDES MARIANELA MARIBEL	2284.01	3048.33	Muerto	174	MICROCREDITO	<input type="button" value="Autorizar"/>	<input type="button" value="Descartar"/>
18144	12001	YARRI ESCOBAR ALBA ROODI	410.06	111.00	Economía Solidaria	180	CONSUMO PRIORITARIO	<input type="button" value="Autorizar"/>	<input type="button" value="Descartar"/>
18111	41120	CLASOCTA POMASQUI CAROLITA SUSANA	643.59	7020	Muerto	328	MICROCREDITO	<input type="button" value="Autorizar"/>	<input type="button" value="Descartar"/>
17852	40774	ANBRAMCO ANBRAMCO ESTANISLAO JUAN	549.45	7080	Muerto	140	MICROCREDITO	<input type="button" value="Autorizar"/>	<input type="button" value="Descartar"/>
12464	43072	VIVERO ALDO OLIVERA AMERICA	542.8	3540	Muerto	343	CONSUMO PRIORITARIO	<input type="button" value="Autorizar"/>	<input type="button" value="Descartar"/>
20348	42421	QUEVEDA FERNANDO WILLIAN EDUARDO	1498.20	10710	Muerto	708	MICROCREDITO	<input type="button" value="Autorizar"/>	<input type="button" value="Descartar"/>
17934	40788	TABEZ TANDUCHI LUIS FERNANDO	4678.78	10335	Muerto	418	CONSUMO PRIORITARIO	<input type="button" value="Autorizar"/>	<input type="button" value="Descartar"/>
20884	43854	NAVARRETE VILALBA MARCELA DE JESUS	300.89	12130	Muerto	138	MICROCREDITO	<input type="button" value="Autorizar"/>	<input type="button" value="Descartar"/>
20792	42044	GALINDO LANDETA CHRISTIAN DE WILSON	348.74	7260	Muerto	174	CONSUMO PRIORITARIO	<input type="button" value="Autorizar"/>	<input type="button" value="Descartar"/>

Fig. 39. Autorización y asignación de nuevos juicios

Usuario: Monitoreo Gerencia y Sub-gerencia de Negocios

The screenshot shows a web application interface with a dark blue header. The header contains the text 'Inicio' on the left, a search bar with 'Buscar' and a yellow 'Buscar' button, and navigation links 'Juicios', 'Resultados', and 'Cuenta' on the right. Below the header, there is a breadcrumb 'Dashboard > Home'. The main content area has a light gray background with a geometric pattern. It features a large heading 'Bienvenid@ Santy Lopez'. Below this, there are two tabs: 'Reporte de monitoreos' (active) and 'Lista de monitoreos'. The 'Reporte de monitoreos' tab contains a form with two date selection fields: 'Fecha de inicio' and 'Fecha de fin', each with a 'Calendario' button and an input field. A yellow 'Visualizar' button is positioned below the date fields.

Fig. 40. Reporte de monitoreos del usuario de gerencias

The screenshot shows the same web application interface as Fig. 40, but with the 'Lista de monitoreos' tab selected. The main content area displays the heading 'Lista de meses en los que se ha realizado los monitoreos'. Below this heading, there is a table with one row containing the text '12-2017' and a yellow 'Eliminar monitoreo' button. The browser's address bar at the bottom shows 'localhost:3000/home_creditos#tab2'.

Fig. 41. Eliminación de monitoreo

Validación y revisión.

PLAN DE PRUEBA Y REPORTE DE EVALUACIÓN

Entidad: Cooperativa de Ahorro y Crédito Mujeres Unidas

Fecha: 09 de enero 2018

Mediante el presente informe se documenta la realización completa del plan de pruebas y evaluación del sistema para verificar y validar la funcionalidad correcta del sistema en base a los requerimientos solicitados por la cooperativa.

IDENTIFICA DOR DE LA HISTORIA	TAREA	ESTADO
sp12-01	Autenticación de Usuarios en el Sistema	Aprobado
sp12-02	Gestión de usuarios y cuentas.	Aprobado
sp12-03	Visualización de un dashboard de módulos del Sistema	Aprobado
sp12-04	Autorizar o rechazar juicios	Aprobado
sp12-05	Visualización de créditos judiciales	Aprobado
sp12-06	Semaforización de etapas estimadas	Aprobado
sp12-07	Gestión de estado de juicios	Aprobado
sp12-08	Detalles de los juicios	Aprobado
sp12-09	Gráfica de procesos de juicios	Aprobado
sp12-10	Gestión de abogados	Aprobado
sp12-11	Gestión de subactividades de juicios	Aprobado
sp12-12	Monitoreo	Aprobado
sp12-13	Previsualización del monitoreo	Aprobado
sp12-14	Visualización de resultados	Aprobado
sp12-15	Buscador de juicios	Aprobado
sp12-16	Reportes de monitoreos	Aprobado
sp12-17	Convertir juicios de bienes a bienes y viceversa.	Aprobado
sp12-18	Gestión de etapas judiciales	Aprobado
sp12-19	Asignación de juicios a los asesores	Aprobado
sp12-20	Clasificación de procesos de los juicios	Aprobado
sp12-21	Privilegio de acceso a asesores y facturas de cobranzas	Aprobado
sp12-22	Eliminación de monitoreos	Aprobado
sp12-23	Gestión de las tablas como administrador.	Aprobado

2.3. Sprint 2

Planificación

En el sprint 2 se realizará el desarrollo del módulo de créditos, la herramienta de Gestión de la Cartera de Crédito es parte del Sistema Web para la Gestión y Monitoreo de Áreas Estratégicas de la Cooperativa de Ahorro y Crédito Mujeres Unidas CACMU Ltda. – Tantanakushka Warmikunapak y contiene información sobre las operaciones de créditos que se encuentran dentro de las bases de datos interna de la cooperativa. El objetivo de la herramienta es automatizar la información de la cartera de la cooperativa, para el manejo y administración eficiente facilitando la toma de decisiones y planificación efectiva de la misma. La herramienta automatiza información de la base de datos del Sistema Contable VIMACCOOP a través de herramientas de crédito, riesgo, productividad y reportería para el manejo eficiente y oportuno de las diligencias del departamento de crédito, riesgos y gerencia de la institución. (Cadena & López, 2018b).

El módulo incluye una serie de herramientas que se detallarán a continuación, los cuales permiten a las áreas de negocio y riesgo obtener información detallada por agencias y por asesores para toma de decisiones tanto para la administración central como para cada una de las agencias. Adicionalmente, se incluye como parámetros los días en mora para los diversos análisis de cada una de las áreas.

Planificación de la cartera

La planificación de la cartera, contiene información sobre la cartera de créditos vigente distribuida por semanas y por días del mes actual, para la obtención de este reporte se necesitará como datos de entrada al formulario un rango de días mora, la elección de una agencia y un asesor, el objetivo principal de esta herramienta es conocer en qué semana y en qué días se concentran la mayor parte del total de los valores planificados por recuperar versus el valor total recuperado, la planificación del valor total de la cartera por recuperar estará distribuida en porcentajes por cada semana del mes como muestra la figura 42.

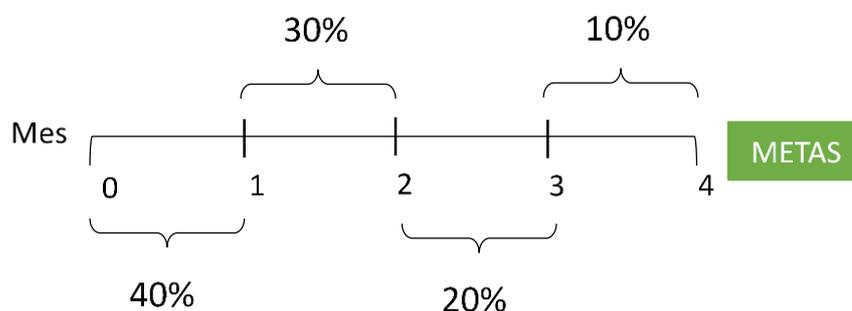


Fig. 42. Porcentajes de colocación por semana

Fuente: (Cadena & López, 2018b)

A su vez, se podrá visualizar los valores de las cuotas, cantidad, provisiones, y valores recuperados por cada semana del mes junto con la opción de ver los créditos pertenecientes a cada semana. Las semanas del mes estarán calculadas de la siguiente manera:

- Primera Semana: Del 1 al 7 del mes actual.
- Segunda Semana: 8 al 14 del mes actual.
- Tercera Semana: 15 al 21 del mes actual.
- Cuarta Semana: 22 al fin del mes actual.

Cartera Vencida

Para la obtención de la cartera vencida se necesitará como datos de entrada la fecha al que se quiere consultar la cartera vencida, un rango de días mora y la selección de si se desea un reporte por agencia o por asesor. La consulta despliega la cartera vigente distribuida según la norma para la gestión en las cooperativas de ahorro y crédito de la Resolución No. 129-2015-F de la Junta de Política y Regulación Monetaria y Financiera (Cadena & López, 2018b), con una serie de variables crediticias y socioeconómicas que son: montó de créditos, capital activo, capital que no devenga intereses, capital vencido, cartera afectada, saldo cartera y porcentaje en mora de la cartera vencida de la cooperativa distribuida por agencias o asesores junto con la opción de visualizar el detalle de los créditos pertenecientes a cada categoría.

Eficiencia de la cartera

El objetivo de esta herramienta es dar un seguimiento a todos los créditos que se encuentran en mora incluso si no presentan un riesgo para la institución. Para la obtención de la eficiencia de cartera se necesitará como datos de entrada un rango de días mora en números y escoger si se desea un reporte por agencia, asesor o grupo de créditos mediante. El sistema entregará un reporte diferente en base a la selección del

usuario en donde podrá visualizar la sumatoria de número de créditos morosos, saldo capital pendiente, número créditos, saldo de la cartera, porcentaje de créditos morosos, y saldo en mora, asesores junto con la opción de visualizar el detalle de los créditos pertenecientes a cada categoría.

Matrices de transición

Para la obtención de las matrices de transición se necesitará un formulario con la fecha de inicio y fin del reporte, la elección de una agencia y un asesor. El sistema mostrará dos matrices de transición (matriz de saldos, matriz de cantidades) en base a la calificación de los créditos que calcula en función de los días de morosidad y el segmento del crédito al que pertenece como se presenta en la TABLA 33. Esta clasificación muestra la variación de los créditos según la calificación que haya tenido desde la fecha de inicio de reporte hasta la fecha de finalización del mismo. También se implementará un formulario para seleccionar dos calificaciones y consultar el detalle de todos los créditos que hayan traicionado entre dichas calificaciones.

TABLA 33: Matriz de transición

		Calificación después de la transición									
		Riesgo Normal			Riesgo Potencial		Riesgo Deficiente		Dudoso Recaudo	Pérdida	
		A-1	A-2	A-3	B-1	B-2	C-1	C-2	D	E	
Calificación Inicial	Riesgo Normal	A-1	X _{A1A1}	X _{A1A2}	X _{A1A3}	X _{A1B1}	X _{A1B2}	X _{A1C1}	X _{A1C2}	X _{A1D}	X _{A1E}
		A-2	X _{A2A1}	X _{A2A2}	X _{A2A3}	X _{A2B1}	X _{A2B2}	X _{A2C1}	X _{A2C2}	X _{A2D}	X _{A2E}
		A-3	X _{A3A1}	X _{A3A2}	X _{A3A3}	X _{A3B1}	X _{A3B2}	X _{A3C1}	X _{A3C2}	X _{A3D}	X _{A3E}
	Riesgo Potencial	B-1	X _{B1A1}	X _{B1A2}	X _{B1A3}	X _{B1B1}	X _{B1B2}	X _{B1C1}	X _{B1C2}	X _{B1D}	X _{B1E}
		B-2	X _{B2A1}	X _{B2A2}	X _{B2A3}	X _{B2B1}	X _{B2B2}	X _{B2C1}	X _{B2C2}	X _{B2D}	X _{B2E}
	Riesgo Deficiente	C-1	X _{C1A1}	X _{C1A2}	X _{C1A3}	X _{C1B1}	X _{C1B2}	X _{C1C1}	X _{C1C2}	X _{C1D}	X _{C1E}
		C-2	X _{C2A1}	X _{C2A2}	X _{C2A3}	X _{C2B1}	X _{C2B2}	X _{C2C1}	X _{C2C2}	X _{C2D}	X _{C2E}
	Dudoso Recaudo	D	X _{DA1}	X _{DA2}	X _{DA3}	X _{DB1}	X _{DB2}	X _{DC1}	X _{DC2}	X _{DD}	X _{DE}
	Pérdida	E	X _{EA1}	X _{EA2}	X _{EA3}	X _{EB1}	X _{EB2}	X _{EC1}	X _{EC2}	X _{ED}	X _{EE}

Fuente: (Cadena & López, 2018b)

Cosechas

La consulta de cosechas incluye una clasificación de toda la cartera que se encuentra afectada según la fecha de concesión del crédito(Naty), en la herramienta de cosechas

se necesitará como datos de entrada en el formulario una fecha de reporte, la selección de una agencia y un asesor, la aplicación mostrará un reporte de la cartera afectada desde los inicios de la cooperativa en base a la fecha de ingreso, clasificando a los créditos por años y meses junto con la opción de ver los detalles de los créditos de un determinado periodo de tiempo.

Indicadores de créditos vigentes

Para los indicadores de créditos vigentes se necesitará un formulario con la fecha de reporte, un rango de días, la selección de una agencia y un asesor, el sistema mostrará los créditos actualmente vigentes de la cooperativa divididos en grupos de indicadores los cuales son:

- Género
- Sector
- Sector calculado
- Tipo de créditos
- Origen de recursos
- Metodología
- Nivel de instrucción
- Estado civil
- Rango de edades
- Rango de ingresos

Indicadores de créditos colocados

Para los indicadores de créditos colocados se necesitará un formulario con la fecha de inicio y fin de colocación, un rango de días en números, la selección de una agencia y un asesor, el sistema mostrará todos los créditos colocados divididos en los mismos indicadores de los créditos anteriores.

Cada una de las tablas, formularios y reportes de este módulo serán desarrollados como componentes de *React*, que nos servirán para reutilizar componentes en diferentes vistas y de esta manera disminuir la redundancia de código dentro del módulo y aumentar la velocidad de manejo de datos y actualizaciones de la interfaz delegando a *React* el manejo de las vistas del módulo, se tendrá también la posibilidad de exportar las tablas de los reportes a hojas de Excel.

Análisis del sprint 2

Historias de usuario de sprint 2

TABLA 34: Historia de usuario 24

HISTORIA DE USUARIO	
NUMERO: sp2 - 01	USUARIO: Asesor de créditos, Jefaturas de Agencias y Gerentes
NOMBRE HISTORIA: Visualización de la planificación de la cartera por agencia y asesor.	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 2
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor de créditos, jefaturas de agencia y gerente, necesitamos obtener un reporte de la cartera que esta por vencer en el mes actual, distribuido por mes, semas y días junto con sus totales y metas de colocación, con la finalidad de tomar decisiones para administrar la cartera de la cooperativa.	
COMO PROBARLO: <ul style="list-style-type: none">• Ingresar sistema• Ingresar al módulo de créditos• En el menú lateral seleccionar Eficiencia de cartera• Ingresar los datos deseados• Visualizar los reportes	

TABLA 35: Historia de usuario 25

HISTORIA DE USUARIO	
NUMERO: sp2 - 02	USUARIO: Asesor de créditos, Jefaturas de Agencias y Gerentes
NOMBRE HISTORIA: Visualización de la cartera vencida por agencia y asesor.	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 2
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor de créditos, jefaturas de agencia y gerente, necesitamos poder obtener un reporte de los créditos vencidos en el mes actual agrupados por agencia o asesor, junto con sus totales, con la finalidad de tomar decisiones para administrar la cartera de la cooperativa y dar mayor importancia en las planificaciones a determinados créditos.	
COMO PROBARLO: <ul style="list-style-type: none">• Ingresar sistema• Ingresar al módulo de créditos• En el menú lateral seleccionar Cartera Vencida• Ingresar los datos deseados para el reporte• Visualizar la cartera vencida.	

TABLA 36: Historia de usuario 26

HISTORIA DE USUARIO	
NUMERO: sp2 - 03	USUARIO: Asesor de créditos, Jefaturas de Agencias y Gerentes
NOMBRE HISTORIA: Visualización de la eficiencia de la cartera por agencia y asesor	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 2
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como un asesor de créditos, jefaturas de agencia y gerente, necesitamos poder obtener un reporte de los créditos colocados en el mes actual clasificados por agencia o asesor, junto con sus totales, con la finalidad de tomar decisiones para administrar la cartera de la cooperativa.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar sistema • Ingresar al módulo de créditos • En el menú lateral seleccionar Eficiencia de Cartera • Ingresar los datos deseados para el reporte • Visualizar la eficiencia de la cartera 	

TABLA 37: Historia de usuario 27

HISTORIA DE USUARIO	
NUMERO: sp2 - 04	USUARIO: Asesor de créditos, Jefaturas de Agencias y Gerentes
NOMBRE HISTORIA: Visualización de las matrices de riesgo por agencia y asesor	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 2
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor de créditos, jefaturas de agencia y gerente, necesitamos poder visualizar las matrices de transición de los créditos extraídos en determinadas fechas por agencia o asesor, clasificados por las calificaciones de los créditos (A1, A2, A3, B1, B2, D, E), junto con un reporte de los créditos pertenecientes a cada transición de la matriz, con la finalidad de tomar decisiones para administrar la cartera de la cooperativa.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar sistema • Ingresar al módulo de créditos • En el menú lateral seleccionar Matrices de riesgo • Ingresar los datos deseados para el reporte 	

- Visualizar la matriz de transición de saldos, matriz de transición de cantidades y un formulario para extraer los créditos que pertenecen a una transición.

TABLA 38: Historia de usuario 28

HISTORIA DE USUARIO	
NUMERO: sp2 - 05	USUARIO: Asesor de créditos, Jefaturas de Agencias y Gerentes
NOMBRE HISTORIA: Visualización de las cosechas de créditos por agencia y asesor.	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 2
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor de créditos, jefaturas de agencia y gerente, necesitamos poder visualizar las cosechas de los créditos entre fechas ingresadas por agencia o asesor, con la finalidad de tomar decisiones para administrar la cartera de la cooperativa.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar sistema • Ingresar al módulo de créditos • En el menú lateral seleccionar Cosechas • Ingresar los datos deseados para el reporte • Visualizar las cosechas y la tabla de detalles de los créditos. 	

TABLA 39: Historia de usuario 29

HISTORIA DE USUARIO	
NUMERO: sp2 - 06	USUARIO: Asesor de créditos, Jefaturas de Agencias y Gerentes
NOMBRE HISTORIA: Visualización de los indicadores de créditos vigentes por agencia y asesor	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 2
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor de créditos, jefaturas de agencia y gerente, necesitamos poder visualizar un reporte de los indicadores de créditos vigentes de la cooperativa en una fecha ingresada, por agencia y asesor, los créditos estarán clasificados por: Género, Sector, Sector calculado, Tipo de créditos, Origen de recursos, Metodología, Nivel de instrucción, Estado civil, Rango de edades, Rango de ingresos, con la finalidad de obtener indicadores de operación de la cooperativa.	

COMO PROBARLO:

- Ingresar sistema
- Ingresar al módulo de créditos
- En el menú lateral seleccionar Indicadores C. Vigentes
- Ingresar los datos deseados para el reporte
- Visualizar los indicadores.

TABLA 40: Historia de usuario 30

HISTORIA DE USUARIO	
NUMERO: sp2 - 07	USUARIO: Asesor de créditos, Jefaturas de Agencias y Gerentes
NOMBRE HISTORIA: Visualización de los indicadores de créditos colocados por agencia y asesor.	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 2
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor de créditos, jefaturas de agencia y gerente, necesitamos poder visualizar un reporte de los indicadores de créditos colocados de la cooperativa en una fecha ingresada, por agencia y asesor, los créditos estarán clasificados por: Género, Sector, Sector calculado, Tipo de créditos, Origen de recursos, Metodología, Nivel de instrucción, Estado civil, Rango de edades, Rango de ingresos, con la finalidad de obtener indicadores de operación de la cooperativa.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar sistema • Ingresar al módulo de créditos • En el menú lateral seleccionar Indicadores C. Colocados • Ingresar los datos deseados para el reporte • Visualizar los indicadores. 	

Diseño y codificación del sprint**Diseño de las consultas a la base de datos**

En la TABLA 41, se detallan los parámetros y las características para cada consulta a la base de datos Oracle de cada herramienta.

TABLA 41: Base de datos de la herramienta

Consulta	Variables
Planificación de la cartera - cartera por vencer	<p>Cruces: agencia y asesor</p> <p>Período: mes actual</p> <p>Parámetros: días mora</p> <p>Análisis: saldos cartera, número de créditos, provisión de la cartera, valor recuperado, meta de colocación</p> <p>Informativos: identificación de la operación, identificación del crédito, características del crédito, información socioeconómica socia/o, fechas de pago próxima cuota, fecha</p>

Consulta	Variables
	de pago realizado, criterios de validación del valor de la cuota (VC) y pago realizado (PR)
Cartera vencida	<p>Cruces: agencia o asesor</p> <p>Período: ingreso fecha de reporte</p> <p>Parámetros: días mora</p> <p>Análisis: número de créditos, monto de créditos, capital activo, capital no devenga intereses, capital vencido, cartera afectada, saldo cartera, porcentaje de morosidad</p> <p>Informativas: identificación de la operación, identificación del crédito, características del crédito, información socioeconómica socio/a</p>
Eficiencia de la cartera	<p>Cruces: agencia, asesor o grupo de crédito (e.g. microcrédito, consumo, etc.)</p> <p>Período: día actual</p> <p>Parámetros: días mora</p> <p>Análisis: número de créditos morosos, saldo capital pendiente, número de créditos totales, saldo total, % créditos mora (número), % saldo en mora</p> <p>Informativas: identificación de la operación, identificación del crédito, características del crédito, información socioeconómica socio/a</p>
Matrices de transición	<p>Cruces: agencia y asesor</p> <p>Período: ingreso fecha inicial y final del reporte</p> <p>Parámetros: ninguno</p> <p>Análisis: cartera vigente tabulada por clasificación de crédito, número de créditos vigentes tabulados por clasificación de crédito, % créditos se mantienen, empeoran y mejoran.</p> <p>Informativas: identificación de la operación, identificación del crédito, características del crédito, información socioeconómica socio/a</p>
Cosechas	<p>Cruces: agencia y asesor</p> <p>Período: ingreso único del reporte</p> <p>Parámetros: ninguno</p> <p>Análisis: cartera afectada clasificada por mes y año de concesión</p> <p>Informativas: identificación de la operación, identificación del crédito, características del crédito, información socioeconómica socio/a</p>

Fuente: (Cadena & López, 2018b)

Dependencias del módulo

Para la programación del módulo se utilizará las siguientes librerías de javascript necesarias para el desarrollo del sprint.

- material-ui
- formsy-react
- momento js
- react-bootstrap-table
- reqwest
- WebpackerReact

Producto entregable del sprint

Cumplido el desarrollo y realizado las pruebas de funcionamiento correspondientes se entrega el módulo de créditos terminado y funcionando en el servidor interno de la cooperativa, a continuación, se detalla el sprint backlog resultante del final del sprint y las pantallas finales del módulo créditos.

Sprint backlog

TABLA 42: Sprint Backlog final del spint 2

PILA DEL SPRINT (SPRINT BACKLOG)				
SPRINT				
FECHA DE INICIO: 10 Enero 2018				
FECHA DE CULMINACIÓN: 22 Febrero 2018				
TAREAS PENDIENTES:				0
DIAS PENDIENTES:				0
PRIORIDAD DEL NEGOCIO	DESCRIPCIÓN DE LA TAREA	RESPONSABLE	DURACION DE DIAS	ESTADO
ALTA	Visualización de la planificación de la Cartera por Agencia y Asesor	Santiago López	8	Terminado
ALTA	Visualización de la Cartera vencida por Agencia y Asesor	Santiago López	6	Terminado
ALTA	Visualización de la eficiencia de la Cartera por Agencia y Asesor	Santiago López	6	Terminado
ALTA	Visualización de las matrices de riesgo por Agencia y Asesor	Santiago López	8	Terminado
ALTA	Visualización de las cosechas de créditos por Agencia y Asesor	Santiago López	6	Terminado
ALTA	Visualización de los indicadores de créditos vigentes por Agencia y Asesor	Santiago López	4	Terminado
ALTA	Visualización de los indicadores de créditos colocados por Agencia y Asesor	Santiago López	4	Terminado
ALTA	Privilegios de acceso al módulo de créditos	Santiago López	1	Terminado

Capturas de pantalla del módulo terminado.

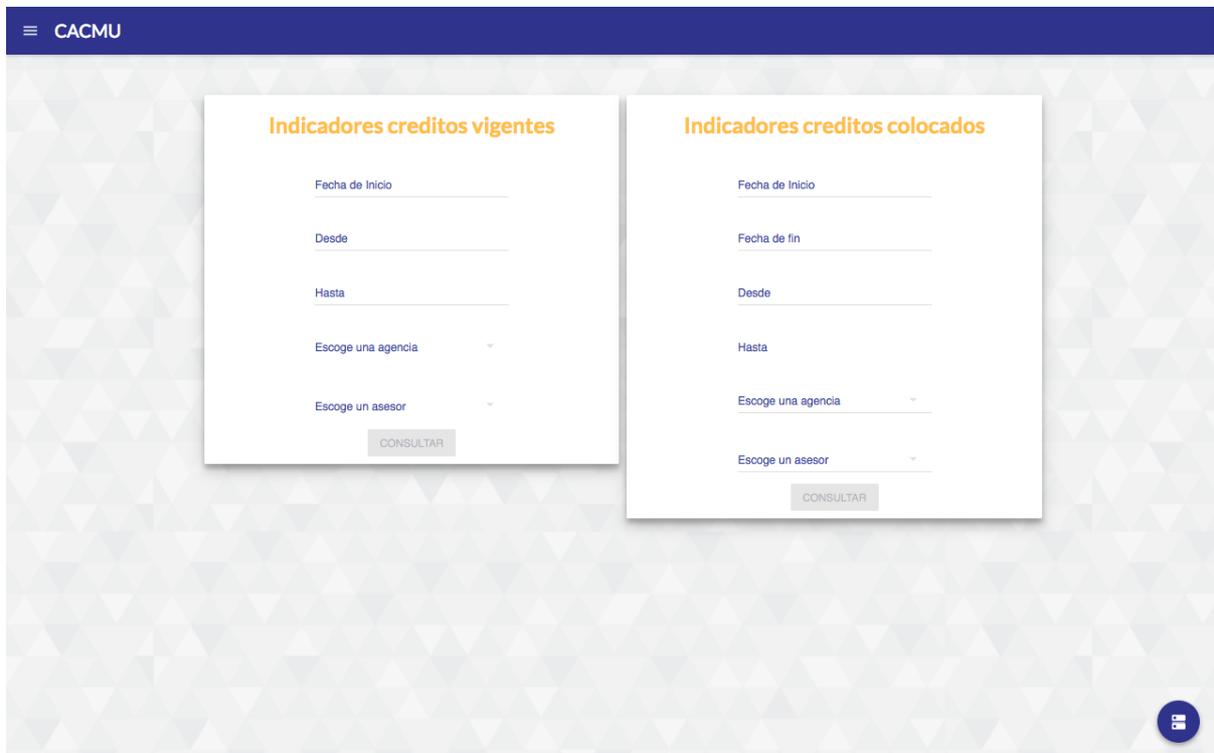


Fig. 43. Pantalla principal módulo de créditos

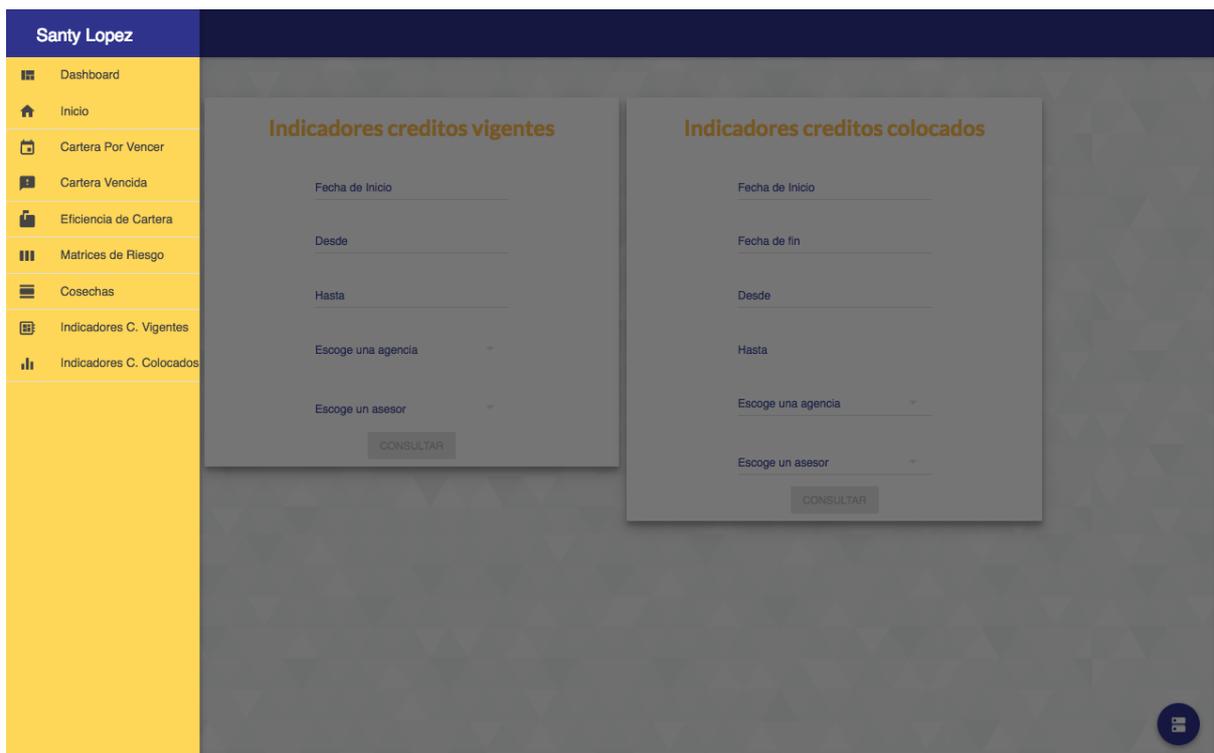


Fig. 44. Menú lateral módulo de créditos

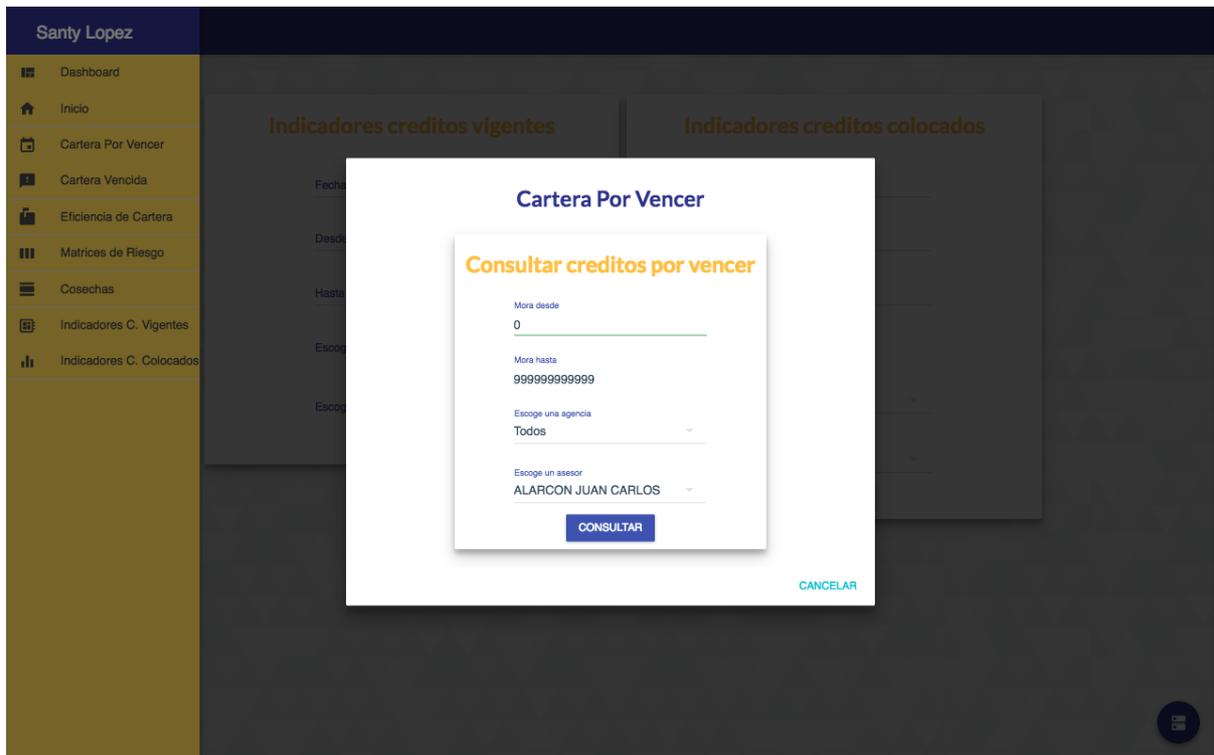


Fig. 45. Formulario de consulta de la planificación de la cartera

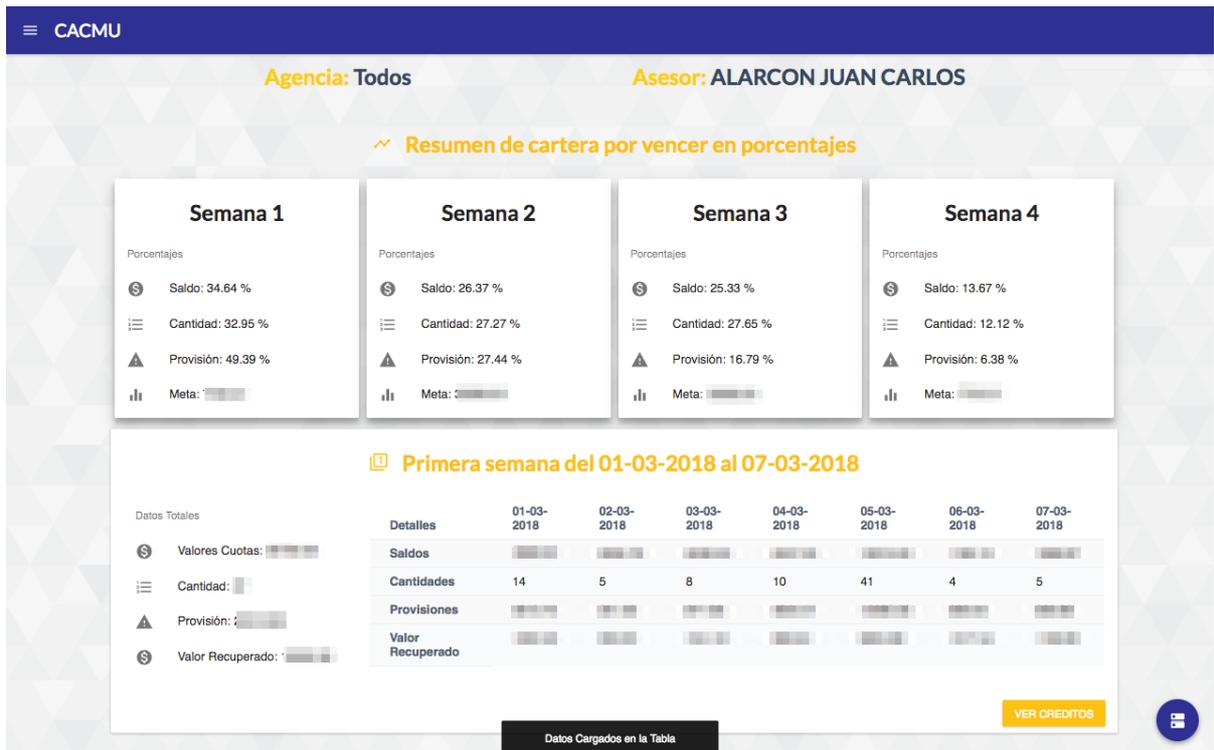


Fig. 46. Planificación de cartera

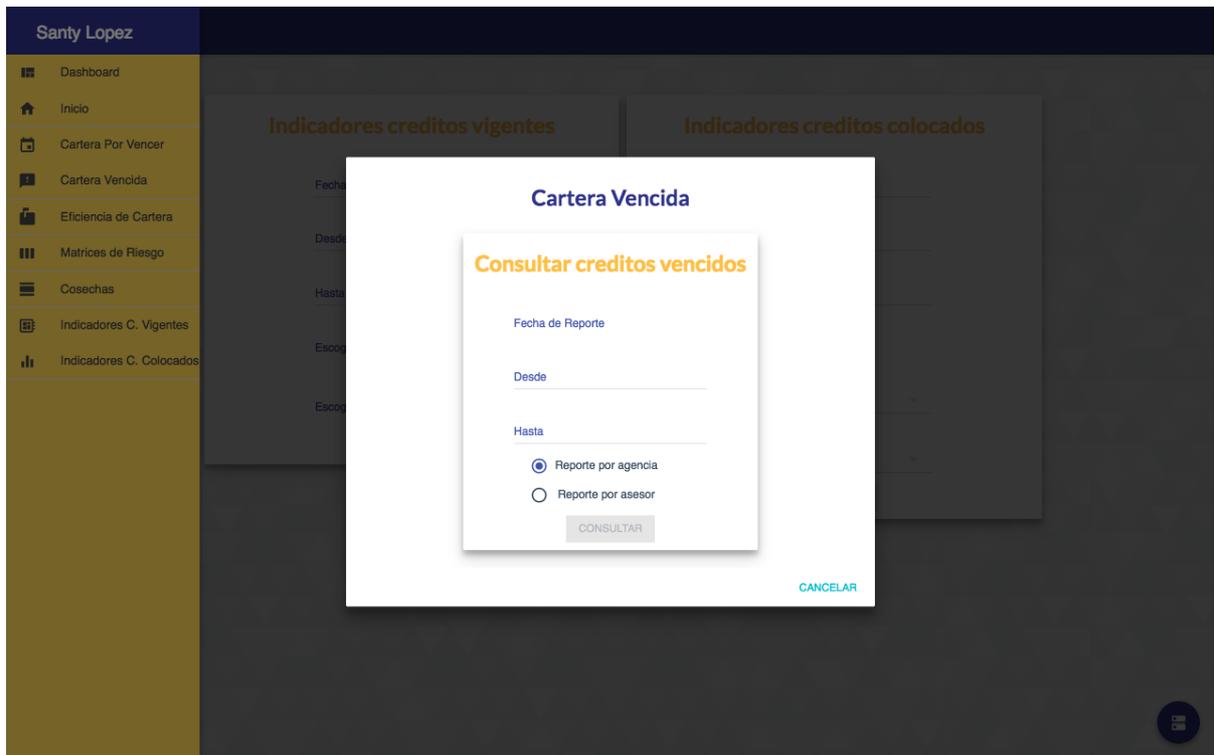


Fig. 47. Formulario de consulta de la cartera vencida

Sucursales	# Creditos	Monto Creditos	Cap Activo	Cap No Dev...	Cap Vencido	Cartera Afec...	Saldo Cartera	% Mora
Cuenca del Lago San Pablo	273							24.15%
Cuenca del Rio Mira	316							17.01%
Economia Solidaria	358							2.48%
Frontera Norte	281							12.70%
La Merced	186							6.42%
Matriz	1750							7.67%
Servimovil	395							8.02%
Valle Fertil	329							14.03%
Total	3888							8.61%

Fig. 48. Créditos vencidos por agencia

Detalles de los créditos

Enter Soci	Enter Nombre...	Enter Codi	Tip ▼▲	Gen... ▼▲	Edad ▼▲	Enter Créd	Periodicid... ▼▲	Fecha Ingre... ▼▲	F
46612			C	F	58	29339	ME	04-03-2013	2
56829			C	F	24	41295	ME	01-06-2016	2
51786			C	M	39	42259	ME	07-07-2014	3
60429			C	M	20	42753	ME	11-12-2017	2
46426			C	M	58	29335	ME	21-01-2013	2
49416			C	M	45	31254	ME	23-12-2013	0
53789			P	F	48	32948	ME	10-04-2015	2
44148			C	F	31	37326	ME	09-07-2012	0
15597			C	F	32	31596	ME	19-08-2010	0
53015			C	F	29	33173	ME	23-12-2014	0
43960			C	F	38	41474	ME	02-07-2012	1
57968			C	M	62	42240	ME	29-11-2016	1
56794			C	M	35	41508	ME	30-05-2016	2
46427			C	F	31	41343	BM	21-01-2013	2
15314			C	F	51	39700	TR	22-07-2010	2
53520			C	M	22	40599	ME	10-03-2015	2
54786			C	F	35	42221	ME	27-07-2015	1
60373			C	M	26	42665	ME	29-11-2017	0
15414			C	F	45	43307	ME	05-08-2010	0
42383			C	F	36	41669	BM	08-03-2012	3
56807			C	F	49	41576	ME	30-05-2016	2
58912			C	M	36	42568	ME	10-05-2017	0
46700			C	M	35	25220	ME	21-03-2013	3
42607			C	M	29	29109	ME	22-03-2012	1
54960			C	M	50	34367	ME	18-08-2015	2

1 2 3 4 5 > >>

Fig. 49. Detalles de los créditos vencidos

Santy Lopez

- Dashboard
- Inicio
- Cartera Por Vencer
- Cartera Vencida
- Eficiencia de Cartera
- Matrices de Riesgo
- Cosechas
- Indicadores C. Vigentes
- Indicadores C. Colocados

Indicadores creditos vigentes

Indicadores creditos colocados

Eficiencia de la Cartera

Consultar eficiencia de cartera

Desde _____

Hasta _____

Reporte por agencia
 Reporte por asesor
 Reporte por grupo credito

CONSULTAR

CANCELAR

Fig. 50. Formulario de consulta de la eficiencia de cartera



Fig. 51. Eficiencia de cartera por agencia

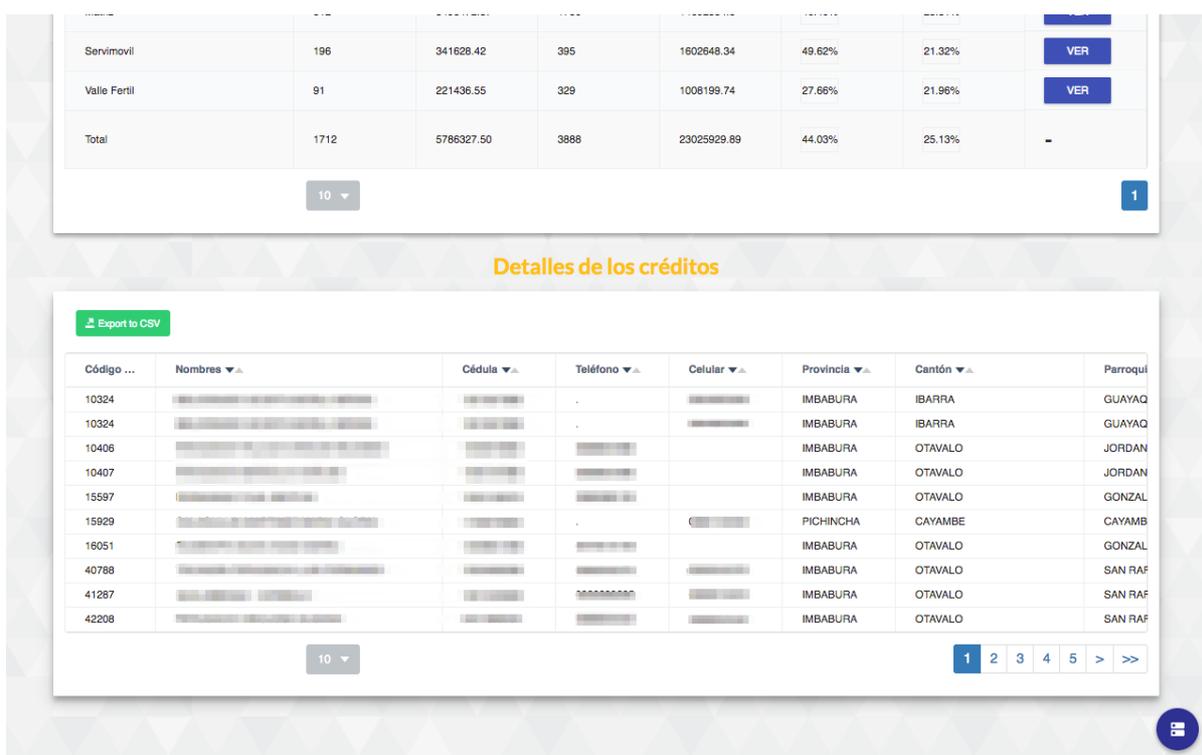


Fig. 52. Detalles de los créditos de la eficiencia de cartera

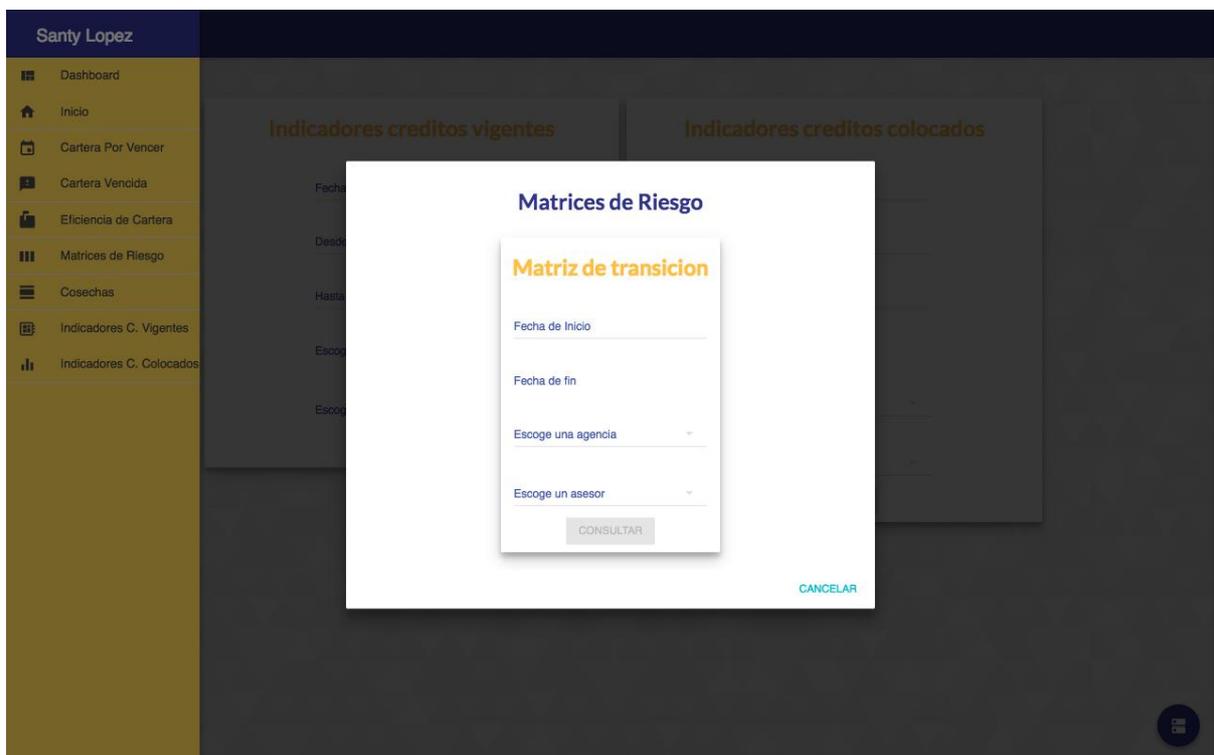


Fig. 53. Formulario de consulta de las matrices de transición.

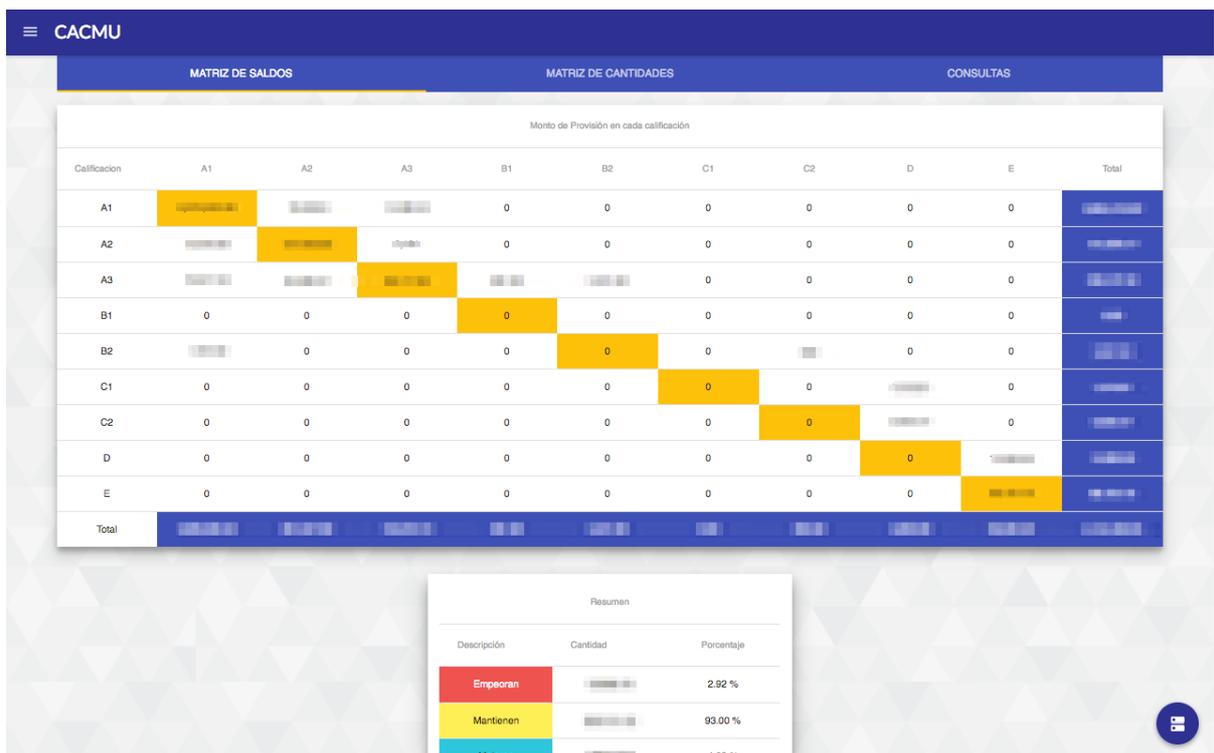


Fig. 54. Matriz de transición de saldos

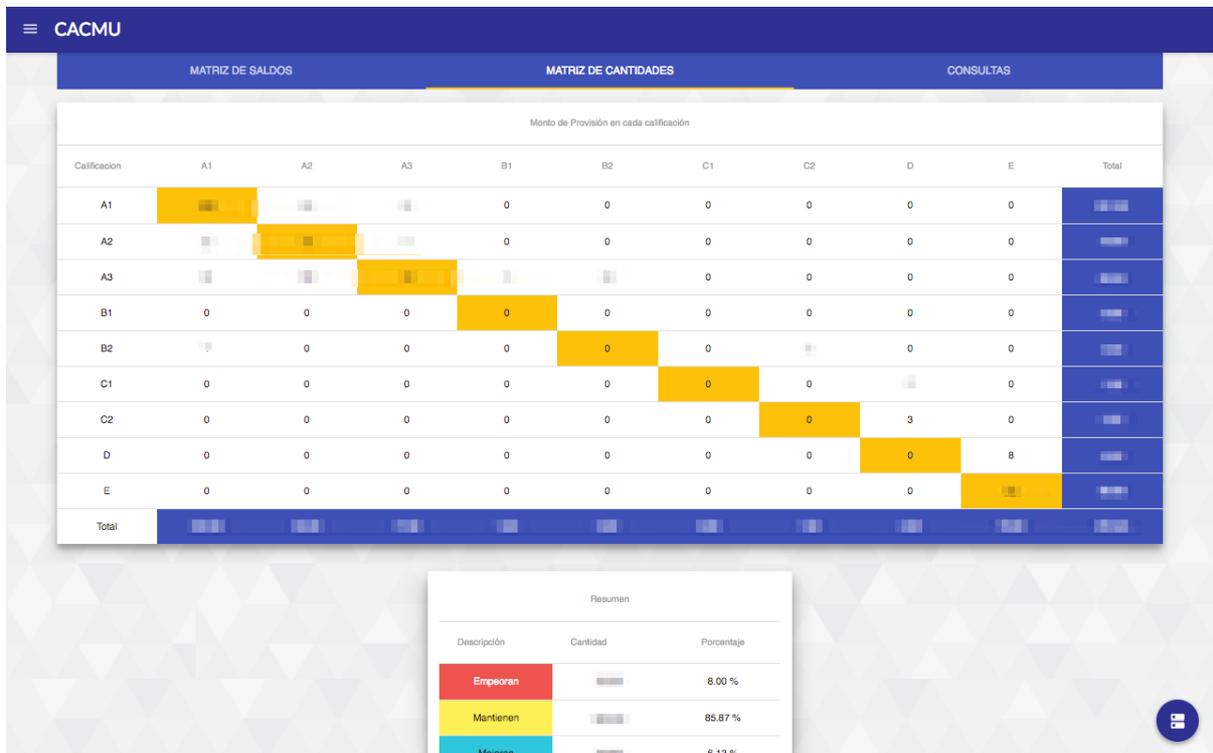


Fig. 55. Matriz de transición de cantidades

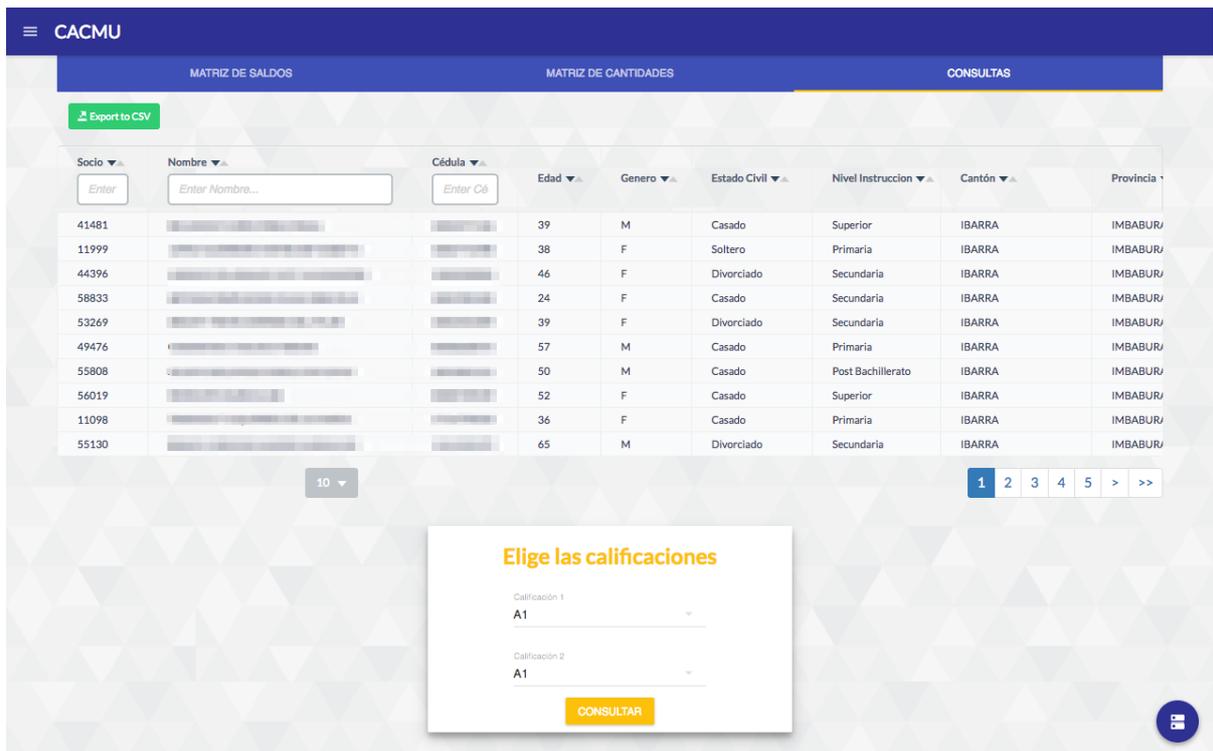


Fig. 56. Consulta de detalles de la matriz de transición

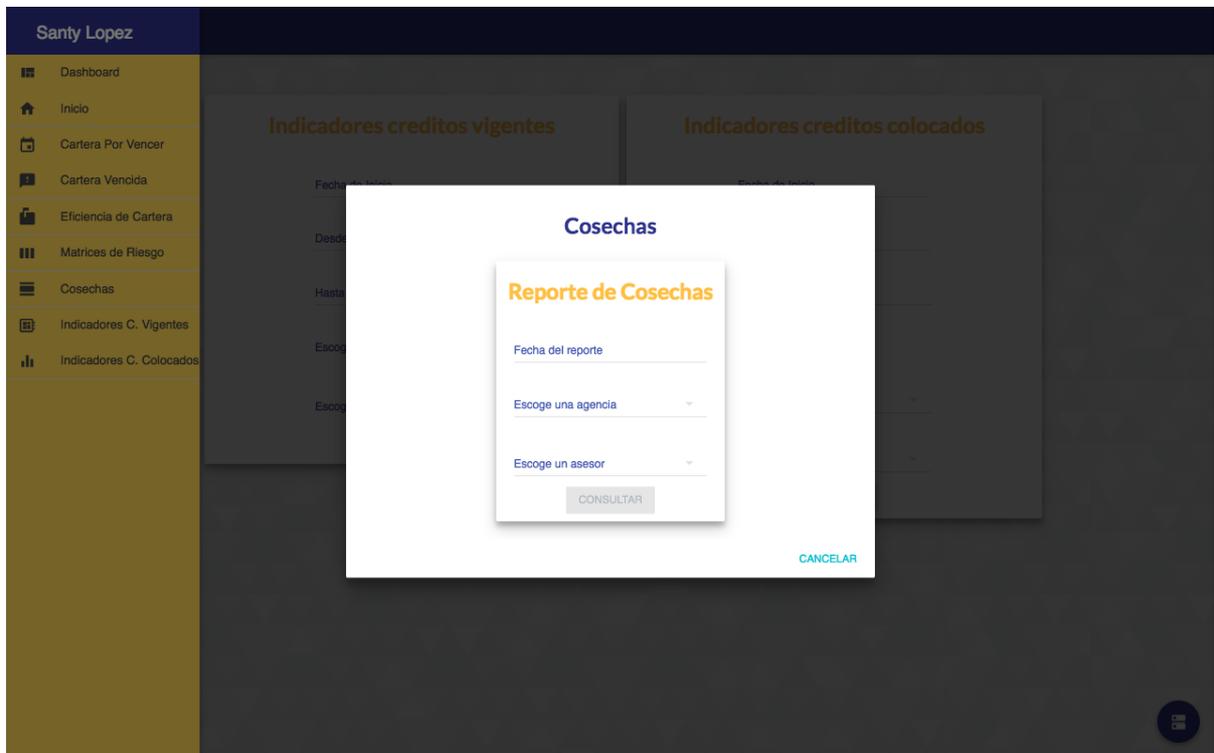


Fig. 57. Formulario de consulta de cosechas

☰ CACMU
Agencia: Todos
Asesor: Todos

Cosechas

Reporte De Mora					
Reporte Mora	Año	Mes	Cantidad	Suma Cartera	Ver
Reporte año 2005					
	2005	Mayo	1	1.00	VER
	TOTAL		1.00	1.00	
Reporte año 2008					
	2008	Octubre	1	1.00	VER
	2008	Diciembre	4	3.91	VER
	TOTAL		5.00	4.91	
Reporte año 2009					
	2009	Octubre	1	1.00	VER
	2009	Noviembre	4	3.91	VER

Fig. 58. Cosechas de la cooperativa CACMU

Enero	VER
Marzo	VER
Septiembre	VER
TOTAL	

Export to CSV

Socio ▼▲	Nombre ▼▲	Cedula ▼▲	Edad ▼▲	Genero ▼▲	credito ▼▲
			37	M	18023
			48	F	18211
			60	F	18141
			53	F	18150
			34	F	18149
			31	F	18144
			31	M	18143
			49	M	17855
			37	M	18145

10 1

Fig. 59. Detalles de las cosechas

≡ CACMU

Indicadores creditos vigentes

Fecha de Inicio

Desde

Hasta

Escoge una agencia

Escoge un asesor

Indicadores creditos colocados

Fecha de Inicio

Fecha de fin

Desde

Hasta

Escoge una agencia

Escoge un asesor

Fig. 60. Formularios de consulta de los indicadores de créditos

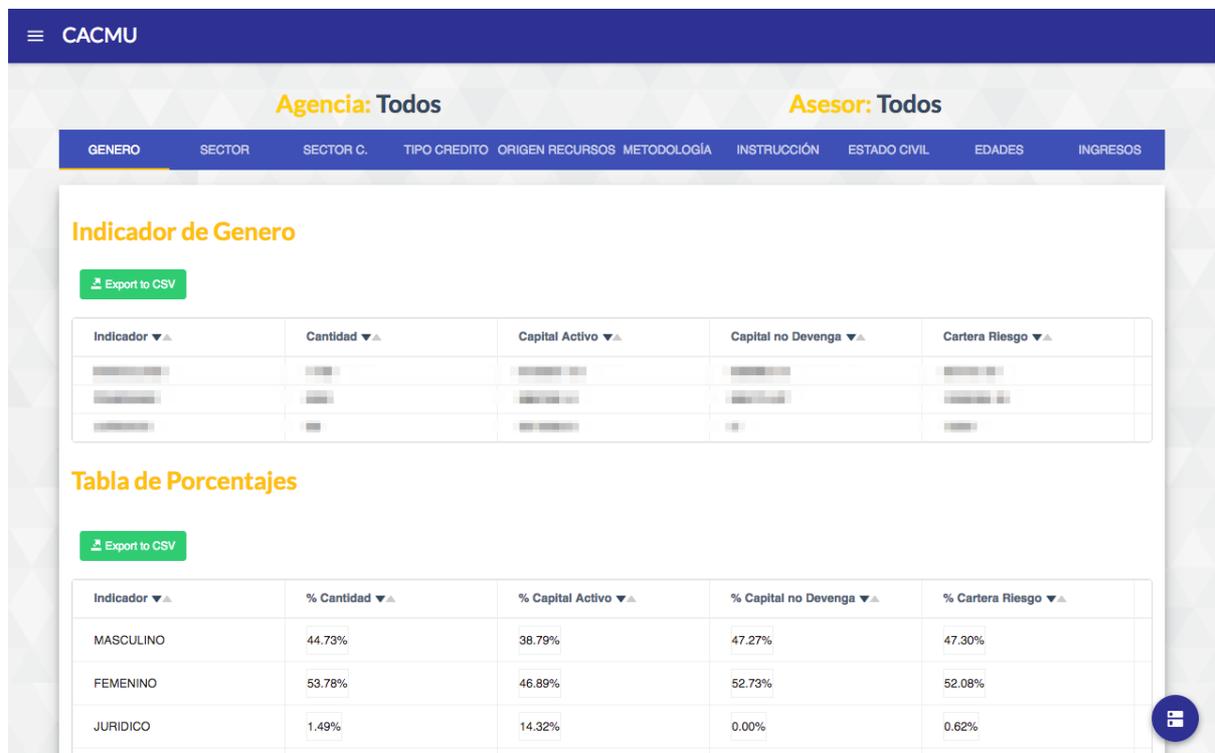


Fig. 61. Indicadores de género de créditos vigentes

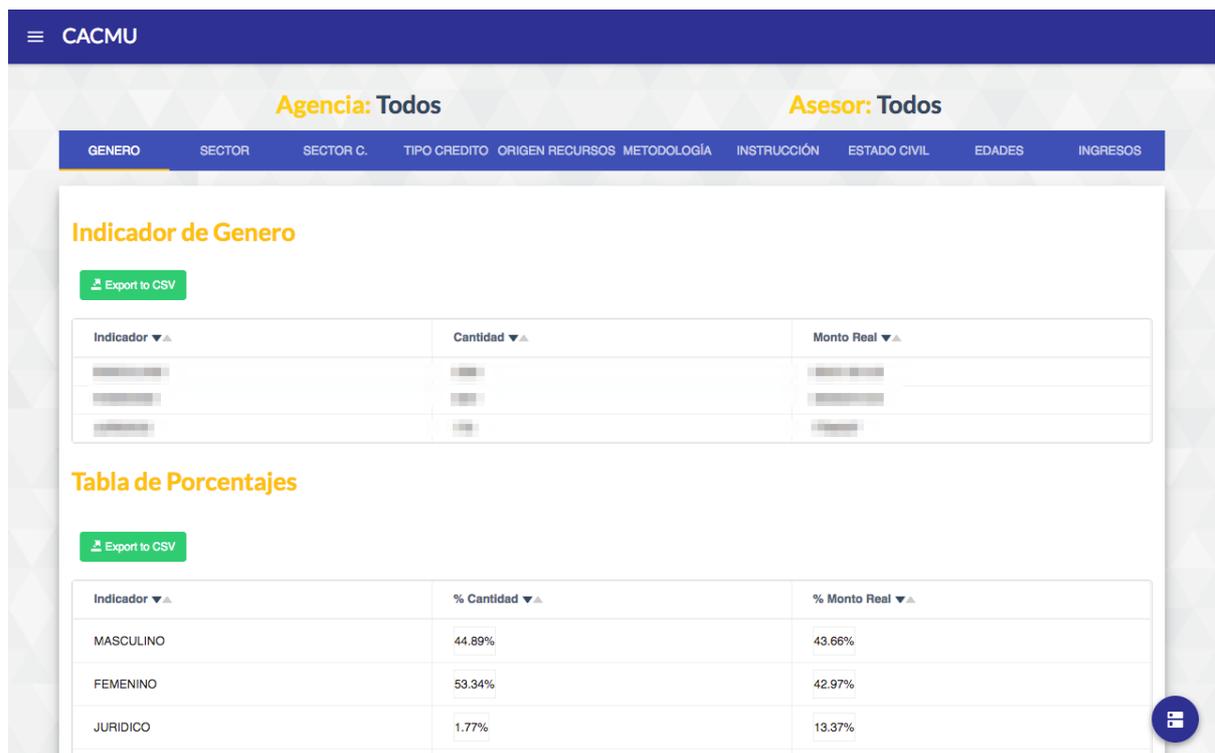


Fig. 62. Indicador de género de créditos colocados

Validación y revisión.

PLAN DE PRUEBA Y REPORTE DE EVALUACIÓN

Entidad: Cooperativa de Ahorro y Crédito Mujeres Unidas

Fecha: 26 de febrero 2018

Mediante el presente informe se documenta la realización completa del plan de pruebas y evaluación del sistema para verificar y validar la funcionalidad correcta del sistema en base a los requerimientos solicitados por la cooperativa.

IDENTIFICADOR DE LA HISTORIA	TAREA	ESTADO
sp24-01	Visualización de la planificación de la Cartera por Agencia y Asesor	Aprobado
sp24-02	Visualización de la Cartera vencida por Agencia y Asesor	Aprobado
sp24-03	Visualización de la eficiencia de la Cartera por Agencia y Asesor	Aprobado
sp24-04	Visualización de las matrices de riesgo por Agencia y Asesor	Aprobado
sp24-05	Visualización de las cosechas de créditos por Agencia y Asesor	Aprobado
sp24-06	Visualización de los indicadores de créditos vigentes por Agencia y Asesor	Aprobado
sp24-07	Visualización de los indicadores de créditos colocados por Agencia y Asesor	Aprobado
sp24-08	Privilegios de acceso al módulo de créditos	Aprobado

2.4. Sprint 3

Planificación

En el sprint 3 se realizará el desarrollo del módulo de agencias de la cooperativa CACMU, en este módulo se implementará dos herramientas listadas a continuación:

- ❖ Indicadores Financieros
- ❖ Indicadores de la SEPS

Para la herramienta de los indicadores financieros como datos de entrada se necesitará escoger una agencia de la que se requiera obtener los indicadores financieros, el sistema entregará un reporte de diciembre del año anterior a diciembre del año actual de las siguientes cuentas de la cooperativa: Activos, caja bancos, cartera bruta, cartera bruta de microcrédito, cartera total en riesgo, reserva por préstamos incobrables, pasivos, obligaciones con el pueblo, depósitos a la vista, dpf más cesantía, patrimonio, certificados de aportación, ingresos totales, intereses y descuentos ganados, operaciones inter financieras, intereses en inversiones, intereses y descuentos de cartera de crédito, ingresos por servicios, otros ingresos, gastos totales, gastos financieros, intereses causados, gastos de provisión, gastos operacionales, gastos de personal. Por cada indicador se tendrá la opción de mostrar una gráfica de evolución de las cuentas en el tiempo y la opción para descargar la tabla a hojas de Excel, la fecha a la que se debe obtener los indicadores debe ser al último día de cada mes, si el mes de consulta es el mes actual se debe consultar al día actual menos uno.

En la herramienta de indicadores de la Superintendencia de Economía Popular y Solidaria (SEPS) se necesitará como datos de entrada la selección de una agencia de la que se desee obtener los indicadores de la SEPS, el sistema consultará las cuentas desde la base de datos y realizará los cálculos necesarios para cada indicador, el reporte será de diciembre del año anterior a diciembre del año actual para cada indicador mostrado a continuación: Solvencia normativa, apalancamiento, liquidez, morosidad ampliada, cobertura de provisiones, relación de productividad, ROA, eficiencia institucional en colocación, grado de absorción de margen financiero neto, tasa activa general, tasa pasiva general, ROE. Por cada indicador se tendrá la opción de mostrar una gráfica de evolución de las cuentas en el tiempo y la opción para descargar la tabla a hojas de Excel, la fecha a la que se debe obtener los indicadores debe ser al último día de cada mes, si el mes de consulta es el mes actual se debe consultar al día actual menos uno.

Análisis del sprint 3

Historias de usuario de sprint 3

TABLA 43: Historia de usuario 31

HISTORIA DE USUARIO	
NUMERO: sp3 - 01	USUARIO: Asesor de créditos, Jefaturas de Agencias y Gerentes
NOMBRE HISTORIA: Visualización de informes financieros por agencia	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 3
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor de créditos, jefaturas de agencia y gerentes, necesitamos visualizar y descargar un reporte de los indicadores financieros de cada uno de las agencias o un consolidado de toda la cooperativa y una gráfica de evolución de cada cuenta en el tiempo, con la finalidad de controlar el estado económico - financiero de la cooperativa.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar sistema. • Ingresar al módulo de agencias. • En informes de cuentas por agencias seleccionar una agencia o la opción de todos. • Ver el reporte de cuentas financieras de diciembre a diciembre junto con sus gráficas. • Descargar la tabla de indicadores financieros. 	

TABLA 44: Historia de usuario 32

HISTORIA DE USUARIO	
NUMERO: sp3 - 02	USUARIO: Asesor de créditos, Jefaturas de Agencias y Gerentes
NOMBRE HISTORIA: Visualización de indicadores de la SEPS por agencia.	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 3
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor de créditos, jefaturas de agencia y gerentes, necesitamos visualizar y descargar un reporte de los indicadores establecidos por la SEPS de cada uno de las agencias o un consolidado de toda la cooperativa y una gráfica de evolución de cada indicador en el tiempo, con la finalidad de visualizar y controlar los indicadores de control establecidos por la entidad pública.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar sistema. • Ingresar al módulo de agencias. • En indicadores de la SEPS seleccionar una agencia o la opción de todos. • Ver el reporte de indicadores de la SEPS de diciembre a diciembre junto con sus gráficas. • Descargar la tabla de indicadores financieros. 	

TABLA 45: Historia de usuario 33

HISTORIA DE USUARIO	
NUMERO: sp3 - 03	USUARIO: Asesor de créditos, Jefaturas de Agencias y Gerentes
NOMBRE HISTORIA: Visualización de indicadores de la SEPS por agencia.	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 3
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor de créditos, jefaturas de agencia y gerentes, necesitamos visualizar y descargar un reporte de los indicadores establecidos por la SEPS de cada uno de las agencias o un consolidado de toda la cooperativa y una gráfica de evolución de cada indicador en el tiempo, con la finalidad de visualizar y controlar los indicadores de control establecidos por la entidad pública.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar sistema. • Ingresar al módulo de agencias. • En indicadores de la SEPS seleccionar una agencia o la opción de todos. • Ver el reporte de indicadores de la SEPS de diciembre a diciembre junto con sus gráficas. • Descargar la tabla de indicadores financieros. 	

Diseño y codificación del sprint

Dependencias del módulo

Para la programación del módulo se utilizará las siguientes librerías de javascript necesarias para el desarrollo del sprint.

- material-ui
- react-html-table-to-excel
- formsy-react
- momento js
- react-bootstrap-table
- reqwest
- WebpackerReact

Producto entregable del sprint

Una vez cumplido el desarrollo y realizado las pruebas de funcionamiento correspondientes se entrega el módulo de agencias terminado y funcionando en el servidor interno de la cooperativa, a continuación, se detalla el sprint backlog resultante del final del sprint y las pantallas finales del módulo de agencias.

Sprint backlog

TABLA 46: Sprint Backlog final del sprint 3

PILA DEL SPRINT (SPRINT BACKLOG)					
SPRINT					
FECHA DE INICIO: 26 febrero 2018					
FECHA DE CULMINACIÓN: 10 marzo 2018					
				TAREAS PENDIENTES:	0
				DIAS PENDIENTES:	0
PRIORIDAD DEL NEGOCIO	DESCRIPCIÓN DE LA TAREA	RESPONSABLE	DURACIÓN DE DIAS	ESTADO	
ALTA	Visualización de informes financieros por agencia	Santiago López	4	Terminado	
ALTA	Visualización de indicadores de la SEPS por agencia	Santiago López	7	Terminado	
ALTA	Privilegios de acceso al módulo de agencias	Santiago López	1	Terminado	

Capturas de pantalla del módulo terminado



Fig. 63. Pantalla principal módulo de agencias

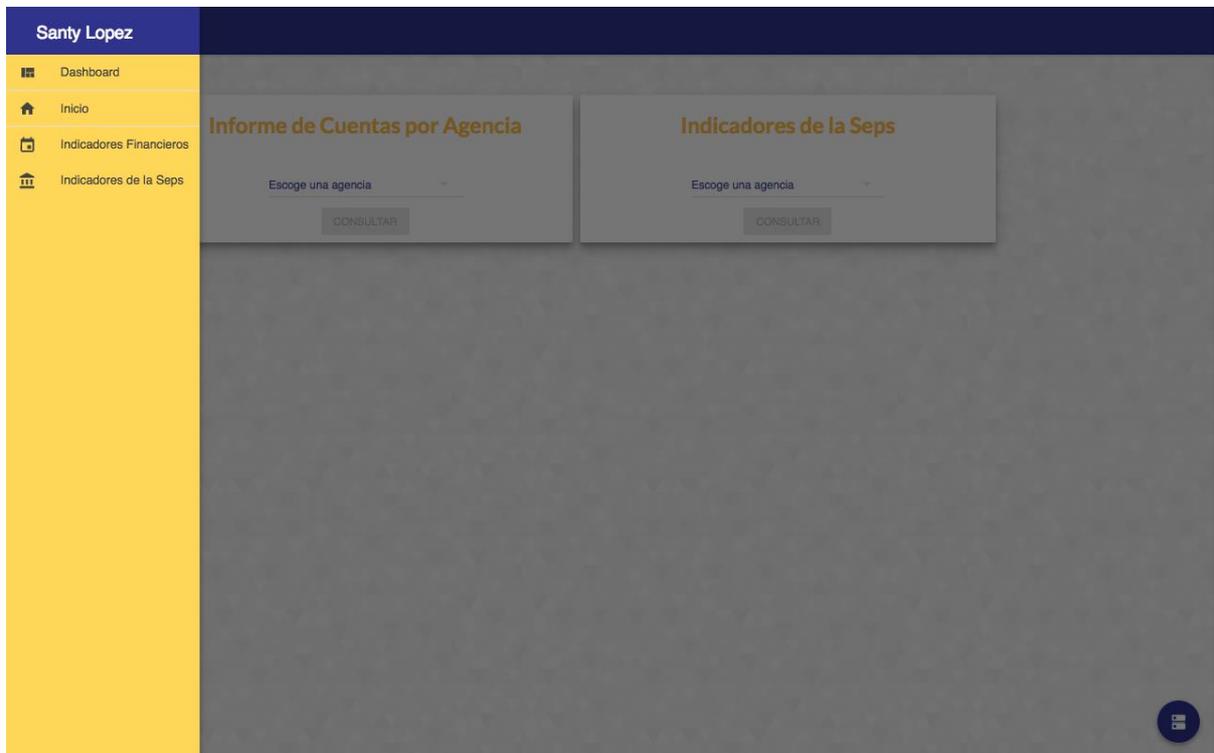


Fig. 64. Menú lateral módulo de agencias

The image shows a web application interface for 'CACMU'. The main content area is titled 'Indicadores Financieros' and includes a 'Descargar excel' button. Below this is a table with 11 columns representing months from December to September and 11 rows representing financial categories. All values in the table are 0.

Cuenta	Diciembre	Enero	Febrero	Marzo	Abril	Mayo	Junio	Julio	Agosto	Septiembre
Activos	0	0	0	0	0	0	0	0	0	0
Fondos	0	0	0	0	0	0	0	0	0	0
Caja Bancos	0	0	0	0	0	0	0	0	0	0
Cartera Bruta	0	0	0	0	0	0	0	0	0	0
Cartera Bruta - microcrédito	0	0	0	0	0	0	0	0	0	0
Cartera en riesgo total	0	0	0	0	0	0	0	0	0	0
Reserva por préstamos incobrables	0	0	0	0	0	0	0	0	0	0
Pasivos	0	0	0	0	0	0	0	0	0	0
Obligaciones con el público	0	0	0	0	0	0	0	0	0	0
Depósitos a la vista	0	0	0	0	0	0	0	0	0	0
DPF + Cesantía	0	0	0	0	0	0	0	0	0	0
Patrimonio	0	0	0	0	0	0	0	0	0	0

Fig. 65. Reporte de indicadores por agencia

INDICADORES SEPS

Descargar excel

Indicadores	dic-2018	ene-2018	feb-2018	mar-2018	abr-2018
Solvencia normativa	18.02%	17.92%	18.11%	18.63%	18.09%
Apalancamiento	6.57%	6.68%	6.58%	6.14%	6.25%
Liquidez	26.27%	27.16%	26.79%	23.85%	23.64%
Morosidad Ampliada	3.01%	2.96%	2.66%	2.40%	5.19%
Cobertura de Provisiones	105.45%	109.00%	118.29%	112.23%	51.73%
Relación de Productividad	89.93%	90.41%	92.41%	92.68%	91.36%
ROA	0.00%	0.01%	0.01%	-0.02%	-0.02%
Eficiencia Institucional en Colocación	0.00%	0.00%	1.00%	1.00%	1.58%
Grado de Absorción de Margen Financiero Neto	0.00%	99.73%	104.58%	133.27%	130.76%
Tasa Activa General	0.00%	1.37%	2.49%	3.52%	3.95%
Tasa Pasiva General	0.00%	0.73%	1.38%	2.15%	2.55%
Roe	18.02%	17.92%	18.11%	18.63%	18.09%

Fig. 68. Gráfica del indicador de solvencia normativa

Validación y revisión.

PLAN DE PRUEBA Y REPORTE DE EVALUACIÓN

Entidad: Cooperativa de Ahorro y Crédito Mujeres Unidas

Fecha: 14 de marzo 2018

Mediante el presente informe se documenta la realización completa del plan de pruebas y evaluación del sistema para verificar y validar la funcionalidad correcta del sistema en base a los requerimientos solicitados por la cooperativa.

IDENTIFICADOR DE LA HISTORIA	TAREA	ESTADO
sp3 – 01	Visualización de informes financieros por agencia	Aprobado
sp3 – 02	Visualización de indicadores de la SEPS por agencia	Aprobado
sp3 – 03	Privilegios de acceso al módulo de agencias	Aprobado

Descripción	Número de Pruebas	Porcentaje
Pruebas Aceptadas	39	97,50%
Pruebas Rechazadas	1	2.5%
Total	40	100%
Pruebas Corregidas	1	2.5%

2.5. Sprint 4

Planificación

En el sprint 4 se realizará el desarrollo del módulo de desempeño social de la cooperativa CACMU, en este módulo se implementará la herramienta de balance social.

Para la obtención del balance social se necesitará como datos de entrada del formulario el ingreso de dos fechas, y la selección de una agencia, en el reporte se visualizará un reporte de las cuentas sociales de la cooperativa y el cálculo de la tasa de variación entre los meses ingresados. El reporte se podrá descargar automáticamente en una hoja de Excel.

Para el desarrollo del módulo; el formulario y la tabla del balance social serán desarrollados como componentes de React para facilitar la reutilización de código en la aplicación.

Análisis del sprint 4

Historias de usuario de sprint 4

TABLA 47: Historia de usuario 34

HISTORIA DE USUARIO	
NUMERO: sp4 - 01	USUARIO: Asesor de créditos, Jefaturas de Agencias y Gerentes
NOMBRE HISTORIA: Obtención del balance social de la cooperativa por agencias	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 4
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor de créditos, jefaturas de agencia y gerentes, necesitamos visualizar y descargar un reporte del balance social de la cooperativa y de cada una de las agencias de la misma, con el cálculo de la tasa de variación entre los dos meses del reporte, con la finalidad de poner en el valor el aporte social de la institución.	
COMO PROBARLO: <ul style="list-style-type: none">• Ingresar sistema.• Ingresar al módulo de desempeño social.• Ingresar las dos fechas para el reporte, escoger una agencia o la opción de todas y consultar.• Visualizar el reporte y su tasa de variación.• Descargar el reporte en la opción de Excel.	

TABLA 48: Historia de usuario 35

HISTORIA DE USUARIO	
NUMERO: sp4 - 02	USUARIO: Asesor de créditos, Jefaturas de Agencias y Gerentes
NOMBRE HISTORIA: Privilegios de acceso al módulo de desempeño social	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 4
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor de créditos, jefaturas de agencia y gerentes, necesitamos poder ingresar al módulo de desempeño social, con la finalidad de utilizar la herramienta del balance social.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar sistema. • Ingresar al módulo de desempeño social. 	

Diseño y codificación del sprint

Dependencias del módulo

Para la programación del módulo se utilizará las siguientes librerías de javascript necesarias para el desarrollo del sprint.

- material-ui
- react-html-table-to-excel
- formsy-react
- react-bootstrap-table
- WebpackerReact

Producto entregable del sprint

Una vez cumplido el desarrollo y realizado las pruebas de funcionamiento correspondientes junto con el departamento de informática de la cooperativa, se entrega el módulo de desempeño social terminado y funcionando en el servidor interno de la cooperativa, a continuación, se detalla el sprint backlog resultante del final del sprint y las pantallas finales del módulo de desempeño social.

Sprint backlog final

TABLA 49: Sprint Backlog final del sprint 4

PILA DEL SPRINT (SPRINT BACKLOG)				
SPRINT				
FECHA DE INICIO: 15 marzo 2018				
FECHA DE CULMINACIÓN: 21 marzo 2018				
TAREAS PENDIENTES:				0
DIAS PENDIENTES:				0
PRIORIDAD DEL NEGOCIO	DESCRIPCIÓN DE LA TAREA	RESPONSABLE	DURACIÓN DE DIAS	ESTADO
ALTA	Obtención del balance social de la cooperativa por agencias	Santiago López	5	Terminado
ALTA	Privilegios de acceso al módulo de desempeño social	Santiago López	1	Terminado

Capturas de pantalla del módulo terminado

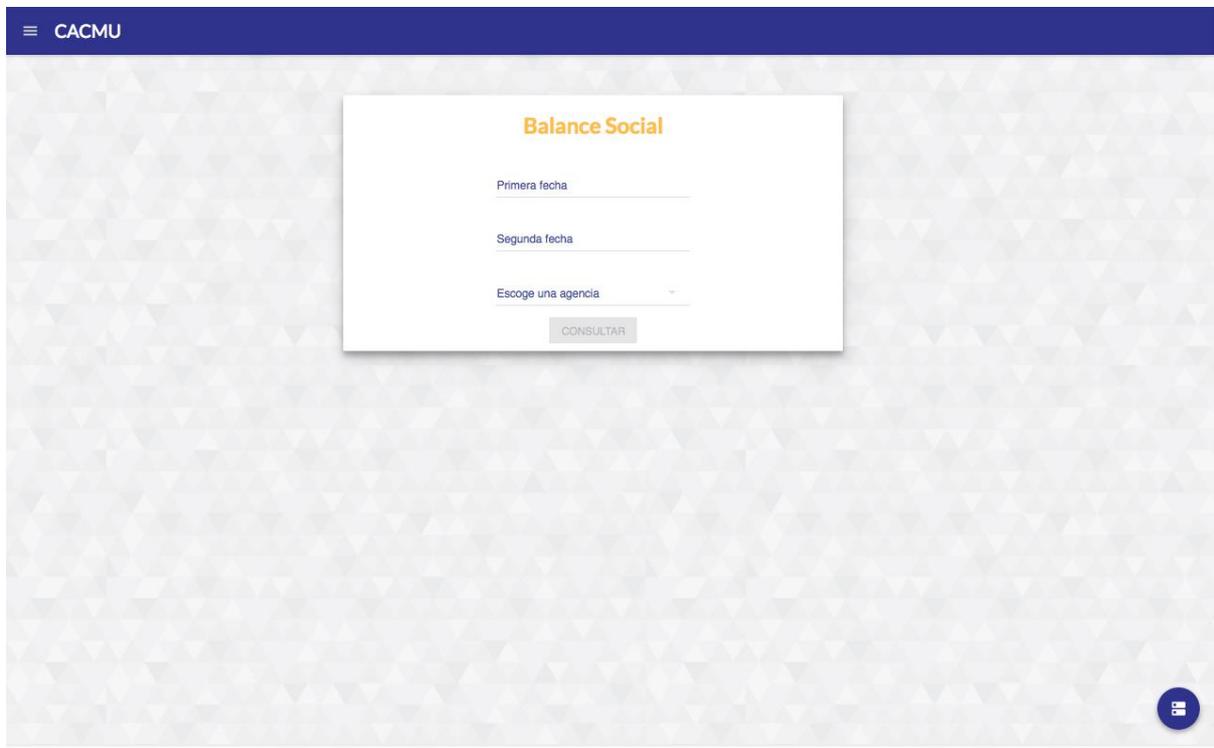


Fig. 69. Pantalla principal módulo de desempeño social

☰ CACMU

Tabla del Balance Social

[Descargar excel](#)

Cuenta	February	% February	March	% March	Diferencia	Tasa de variación
Intereses por créditos otorgados (ingresos ganados)						103.31
Intereses por ahorros & inversiones						104.75
Ingresos por venta de activos						0
Ingresos no operacionales						80.38
Captación de ahorros						0.2
Captación DPF						3.33
Financiamiento local						-2.55
Financiamiento internacional						0
Total valor económico directo creado						0
Pago interes a socios por ahorros & inversiones neto						107.6
Pago a acreedores locales (k+i)						35.44
Pago a proveedores						0
Pago a Estado						219.94
Pago talento humano						109.5
Operación del negocio/gastos de operación						141.51
Inversión en la comunidad						49.41
Total valor económico distribuido						0
Total(Total Directo / Total Distribuido)						

Fig. 70. Reporte de balance social

Santy Lopez

- Dashboard
- Inicio
- Balance Social

Balance Social

Primera fecha

Segunda fecha

Escoge una agencia

[CONSULTAR](#)

Fig. 71. Menú lateral módulo de desempeño social

Validación y revisión.

PLAN DE PRUEBA Y REPORTE DE EVALUACIÓN

Entidad: Cooperativa de Ahorro y Crédito Mujeres Unidas

Fecha: 23 de marzo 2018

Mediante el presente informe se documenta la realización completa del plan de pruebas y evaluación del sistema para verificar y validar la funcionalidad correcta del sistema en base a los requerimientos solicitados por la cooperativa.

IDENTIFICADOR DE LA HISTORIA	TAREA	ESTADO
sp4 – 01	Obtención del balance social de la cooperativa por agencias	Aprobado
sp4 – 02	Privilegios de acceso al módulo de desempeño social	Aprobado

Descripción	Número de Pruebas	Porcentaje
Pruebas Aceptadas	15	100%
Pruebas Rechazadas	0	
Total	15	100%

2.6. Sprint 5

Planificación

En el último sprint se realizará el desarrollo del módulo de recursos humanos de la cooperativa CACMU, en este módulo se implementará la herramienta de control de permisos y vacaciones del personal de la cooperativa.

El módulo debe tener un listado ordenado del personal de la cooperativa y los detalles de cada uno, el sistema deberá calcular los días de vacaciones que por ley debe tener cada empleado en base a la fecha de ingreso a la cooperativa y un cálculo de días de vacación restantes en base a los permisos que ya haya pedido un empleado tomando en cuenta los días de vacaciones acumulados. También se debe tener la posibilidad de tener una planificación de vacaciones por cada empleado con la finalidad de que los asesores y jefes de recursos humanos puedan planificar con anterioridad la salida a vacaciones de cada empleado.

Los asesores de recursos humanos podrán ingresar nuevos empleados (nombres completos, código del empleado, nombre de la agencia, cargo y fecha de ingreso), editar la información de empleados existentes, ingresar permisos, visualizar el historial de permisos de un empleado e ingresar las planificaciones de salidas de vacaciones del empleado.

La jefatura de recursos humanos podrá ingresar y editar empleados, editar o eliminar los permisos ingresados para cada empleado, visualizar historiales de permisos e ingresar y visualizar planificaciones.

Los asesores y jefaturas de recursos humanos deberán tener la posibilidad de culminar el periodo de un empleado cuando haya cumplido el año, con lo cual el aplicativo guardará automáticamente los permisos en un historial que podrá ser consultado después y eliminará los permisos pedidos por el empleado en dicho periodo.

Análisis del sprint 5

Historias de usuario de sprint 5

TABLA 50: Historia de usuario 36

HISTORIA DE USUARIO	
NUMERO: sp5 - 01	USUARIO: Asesor de recursos humanos, Jefatura de recursos humanos
NOMBRE HISTORIA: Listado de los empleados de toda la cooperativa	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 3	ITERACION ASIGNADA: 5
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor de recursos humanos y jefatura de recursos humanos, necesitamos poder visualizar un listado de los empleados de la cooperativa ordenados por su nombre y agencia, con la finalidad de tener fácil acceso a la plantilla de toda la cooperativa.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar sistema. • Ingresar al módulo de recursos humanos. • Ingresar al apartado de manejo de permisos y vacaciones. • Visualizar el listado de empleados. 	

TABLA 51: Historia de usuario 37

HISTORIA DE USUARIO	
NUMERO: sp5 - 02	USUARIO: Asesor de recursos humanos, Jefatura de recursos humanos
NOMBRE HISTORIA: Gestión de empleados de la cooperativa.	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 3	ITERACION ASIGNADA: 5
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor de recursos humanos, necesito poder ingresar los datos de nuevos empleados (nombres completos, código del empleado, nombre de la agencia, cargo y fecha de ingreso) o editar los datos de empleados, con la finalidad de gestionar la plantilla laboral de la cooperativa.	
Como jefe de recursos humanos, necesito poder ingresar los datos de nuevos empleados (nombres completos, código del empleado, nombre de la agencia, cargo y fecha de ingreso) con la finalidad de gestionar la plantilla laboral de la cooperativa.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar sistema. • Ingresar al módulo de recursos humanos. • Utilizar el botón de ingreso de un empleado para visualizar el formulario de ingreso. 	

- En los detalles del empleado usar la opción de editar empleado para actualizar su información.

TABLA 52: Historia de usuario 38

HISTORIA DE USUARIO	
NUMERO: sp5 - 03	USUARIO: Asesor de recursos humanos, Jefatura de recursos humanos
NOMBRE HISTORIA: Visualización del calendario global para mostrar las planificaciones de vacaciones.	
PRIORIDAD DE NEGOCIO: C	
PUNTOS ESTIMADOS: 2	ITERACION ASIGNADA: 5
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor y jefe de recursos humanos, necesitamos visualizar un calendario en donde se muestren las planificaciones de permisos de todos los empleados, con la finalidad de gestionar de una mejor manera las salidas a vacaciones de los empleados de la cooperativa.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar sistema. • Ingresar al módulo de recursos humanos. • Utilizar el botón de calendario para visualizar las planificaciones ingresadas de todos los empleados 	

TABLA 53: Historia de usuario 39

HISTORIA DE USUARIO	
NUMERO: sp5 - 04	USUARIO: Asesor de recursos humanos, Jefatura de recursos humanos
NOMBRE HISTORIA: Calculo automático de vacaciones por empleado.	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 5	ITERACION ASIGNADA: 5
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor y jefe de recursos humanos, necesitamos ver cuantos días de vacaciones por derecho tiene el empleado en base a las normativas del estado y cuantos días de vacaciones restantes tiene un empleado conforme a los permisos y salidas que ha tomado en el periodo y días acumulados, con la finalidad de llevar un control eficiente de las vacaciones de cada empleado.	
COMO PROBARLO: <ul style="list-style-type: none"> • Ingresar sistema. • Ingresar al módulo de recursos humanos. 	

- Ingresar a al módulo de manejo de permisos y vacaciones e ingresar a los detalles de un empleado, visualizar los campos: días vigentes, días acumulados, días restantes, horas restantes, días ocupados, horas ocupadas.

TABLA 54: Historia de usuario 40

HISTORIA DE USUARIO	
NUMERO: sp5 - 05	USUARIO: Asesor de recursos humanos, Jefatura de recursos humanos
NOMBRE HISTORIA: Gestión de permisos pedidos por el empleado.	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 3	ITERACION ASIGNADA: 5
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
<p>DESCRIPCIÓN: Como asesor un asesor de recursos humanos, necesito poder ingresar los datos de un permiso (fecha del permiso, descripción del permiso, número de horas) de un empleado, con la finalidad de llevar un control de las salidas de los empleados y calcular los días de vacaciones consumidos por el empleado.</p> <p>Como un jefe de recursos humanos, necesito editar y eliminar los permisos ingresados de un empleado, con la finalidad de monitoreo y corrección de errores.</p>	
<p>COMO PROBARLO:</p> <ul style="list-style-type: none"> • Ingresar sistema. • Ingresar al módulo de recursos humanos. • Ingresar a al módulo de manejo de permisos y vacaciones e ingresar a los detalles de un empleado, utilizar el botón de Ingresar permisos para mostrar el formulario de ingreso y observar la actualización del cálculo de días restantes. • Usar los botones de editar y eliminar permisos para gestionarlos. 	

TABLA 55: Historia de usuario 41

HISTORIA DE USUARIO	
NUMERO: sp5 - 06	USUARIO: Asesor de recursos humanos, Jefatura de recursos humanos.
NOMBRE HISTORIA: Ingreso de planificaciones de permisos por empleado.	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 3	ITERACION ASIGNADA: 5
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
<p>DESCRIPCIÓN: Como asesor y jefe de recursos humanos, necesitamos ingresar las planificaciones de vacaciones (fecha de inicio, fecha de finalización) de cada empleado, con la finalidad de realizar una planificación temprana de las vacaciones que debe tomar un empleado al año.</p>	

COMO PROBARLO:

- Ingresar sistema.
- Ingresar al módulo de recursos humanos.
- Ingresar a al módulo de manejo de permisos y vacaciones e ingresar a los detalles de un empleado, utilizar el botón de planificaciones.
- Usar el botón de ingreso de planificaciones para mostrar el formulario de ingreso.
- Verificar la creación de la planificación en el calendario.

TABLA 56: Historia de usuario 42

HISTORIA DE USUARIO	
NUMERO: sp5 - 07	USUARIO: Asesor de recursos humanos, Jefatura de recursos humanos.
NOMBRE HISTORIA: Visualización del calendario de planificaciones por empleado.	
PRIORIDAD DE NEGOCIO: C	
PUNTOS ESTIMADOS: 4	ITERACION ASIGNADA: 5
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor y jefe de recursos humanos, necesitamos visualizar un calendario en el cual se muestren las planificaciones a vacaciones de un determinado empleado, con la finalidad de gestionar de una mejor manera las salidas a vacaciones de dicho empleado.	
COMO PROBARLO:	
<ul style="list-style-type: none"> • Ingresar sistema. • Ingresar al módulo de recursos humanos. • Ingresar a al módulo de manejo de permisos y vacaciones e ingresar a los detalles de un empleado, utilizar el botón de planificaciones. • Visualizar el calendario con las planificaciones ya ingresadas. 	

TABLA 57: Historia de usuario 43

HISTORIA DE USUARIO	
NUMERO: sp5 - 08	USUARIO: Asesor de recursos humanos, Jefatura de recursos humanos.
NOMBRE HISTORIA: Culminaciones de periodos de vacaciones de empleados.	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 4	ITERACION ASIGNADA: 5
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor y jefe de recursos humanos, necesitamos poder culminar el periodo de vacaciones de un empleado, con la finalidad de pasar los días restantes de vacaciones de un empleado a días acumulados y volver a realizar el cálculo de días de vacaciones que por derecho deberá tener el empleado en el nuevo periodo.	

COMO PROBARLO:

- Ingresar sistema.
- Ingresar al módulo de recursos humanos.
- Ingresar a al módulo de manejo de permisos y vacaciones e ingresar a los detalles de un empleado.
- Utilizar el botón de Culminar periodo, visualizar la actualización de días acumulados y días vigentes.

TABLA 58: Historia de usuario 44

HISTORIA DE USUARIO	
NUMERO: sp5 - 09	USUARIO: Asesor de recursos humanos, Jefatura de recursos humanos.
NOMBRE HISTORIA: Visualización y guardado de historiales de permisos por empleados.	
PRIORIDAD DE NEGOCIO: S	
PUNTOS ESTIMADOS: 3	ITERACION ASIGNADA: 5
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	
DESCRIPCIÓN: Como asesor y jefe de recursos humanos, necesitamos visualizar un historial de permisos que haya pedido el empleado y actualizar los permisos cada vez que se culmine el periodo del empleado, con la finalidad de tener historiales de respaldo de permisos para quejas o reclamos.	
COMO PROBARLO:	
<ul style="list-style-type: none"> • Ingresar sistema. • Ingresar al módulo de recursos humanos. • Ingresar a al módulo de manejo de permisos y vacaciones e ingresar a los detalles de un empleado. • Ver los historiales de permisos del empleado utilizando el botón de historial de permisos. 	

TABLA 59: Historia de usuario 45

HISTORIA DE USUARIO	
NUMERO: sp5 - 10	USUARIO: Asesor de recursos humanos, Jefatura de recursos humanos.
NOMBRE HISTORIA: Privilegios de acceso de los usuarios de recursos humanos.	
PRIORIDAD DE NEGOCIO: M	
PUNTOS ESTIMADOS: 1	ITERACION ASIGNADA: 5
PROGRAMADOR RESPONSIBLE: Santiago J. López C.	

DESCRIPCIÓN: Como asesor y jefe de recursos humanos, necesitamos que únicamente nosotros podamos ingresar al módulo de recursos humanos, con la finalidad de proteger los datos privados de permisos y vacaciones hacia el resto de los usuarios.

COMO PROBARLO:

- Ingresar sistema.
- Al ingresar otro tipo de usuario a este módulo se mostrará un mensaje de no autorización de ingreso al módulo y se redireccionará al dashboard de la aplicación.

Diseño y Codificación del sprint

Dependencias del módulo

Para la programación del módulo se utilizará las siguientes librerías de javascript necesarias para el desarrollo del sprint.

- material-ui
- react-big-calendar-like-google
- momentjs
- reqwest
- flat-ui
- formsy-react
- WebpackerReact

Producto entregable del sprint

Una vez cumplido el desarrollo y realizado las pruebas de funcionamiento correspondientes junto con el departamento de informática de la cooperativa, se entrega el módulo de desempeño social terminado y funcionando en el servidor interno de la cooperativa, a continuación, se detalla el sprint backlog resultante del final del sprint y las pantallas finales del módulo de recursos humanos.

Sprint backlog final

TABLA 60: Sprint Backlog final del sprint 5

PRIORIDAD DEL NEGOCIO	DESCRIPCIÓN DE LA TAREA	RESPONSABLE	DURACIÓN DE DIAS	ESTADO
ALTA	Listado de los empleados de toda la cooperativa	Santiago López	1	Terminado
ALTA	Gestión de empleados de la cooperativa	Santiago López	3	Terminado
BAJA	Visualización del calendario global para mostrar las planificaciones de vacaciones	Santiago López	3	Terminado
ALTA	Calculo automático de vacaciones por empleado	Santiago López	2	Terminado
ALTA	Gestión de permisos pedidos por empleado	Santiago López	1	Terminado
ALTA	Ingreso de planificaciones de permisos por empleado	Santiago López	2	Terminado
BAJA	Visualización del calendario de planificaciones por empleado	Santiago López	2	Terminado
ALTA	Culminaciones de periodos de vacaciones de empleados	Santiago López	2	Terminado
MEDIA	Visualización y guardado de historiales de permisos por empleados	Santiago López	3	Terminado
ALTA	Privilegios de acceso de los usuarios de recursos humanos	Santiago López	1	Terminado

Capturas de pantalla del módulo terminado

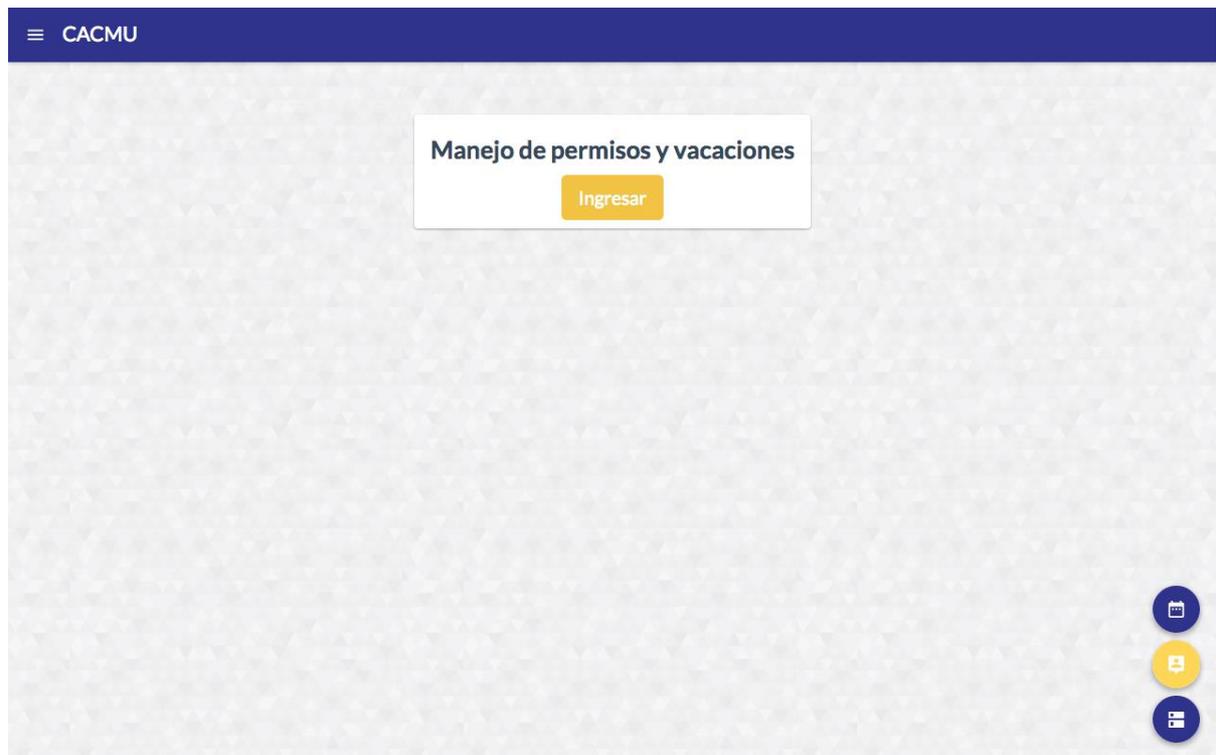


Fig. 72. Pantalla principal módulo de recursos humanos

☰ CACMU

Listado de los empleados registrados

Nómina laboral		
	Amanda Maricela Calagullin Lechin Asesor de Negocios Jr -	
	Jessica Delgado Asistente de Informacion - Frontera Norte	
	Chloe Guevara Gerente - Lita	
	Franz Fisher Asistente Administrativo - Matriz	
	Ja Chavez Subgerente - Matriz	
	Janet Chavez Subgerente - Matriz	
	Janett Gerente - Matriz	
	Lessly Estrada Jefa - Matriz	
	Lessly Madeleyne Asesora - Matriz	
	Madeleyne Jefa - Matriz	
	Narcisa Gomez Carranco Jefa - Matriz	

Fig. 73. Listado de empleado de la cooperativa

☰ CACMU

Ingresar un nuevo empleado

Nombre completo

Código del empleado

Agencia

Cargo

Fecha de ingreso

Calendario

Fecha de calculo de vacaciones

Calendario

Fig. 74. Formulario de ingreso de un nuevo empleado

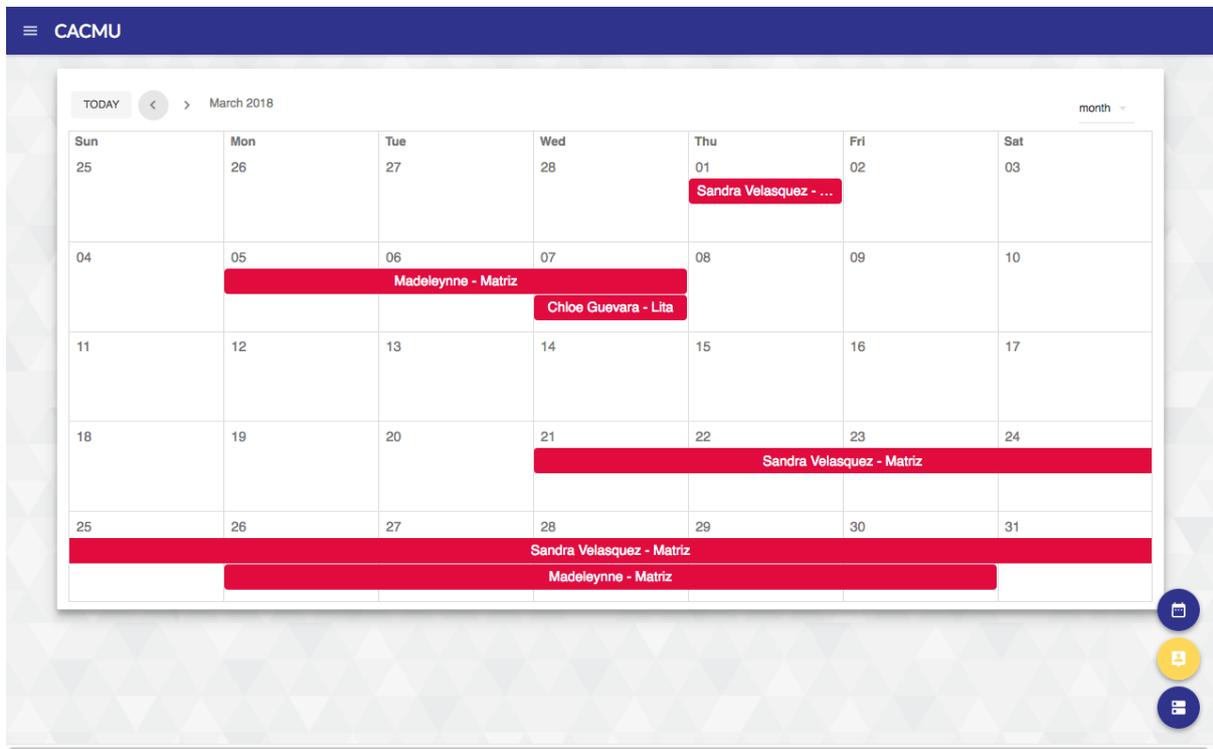


Fig. 75. Calendario global de planificación de vacaciones

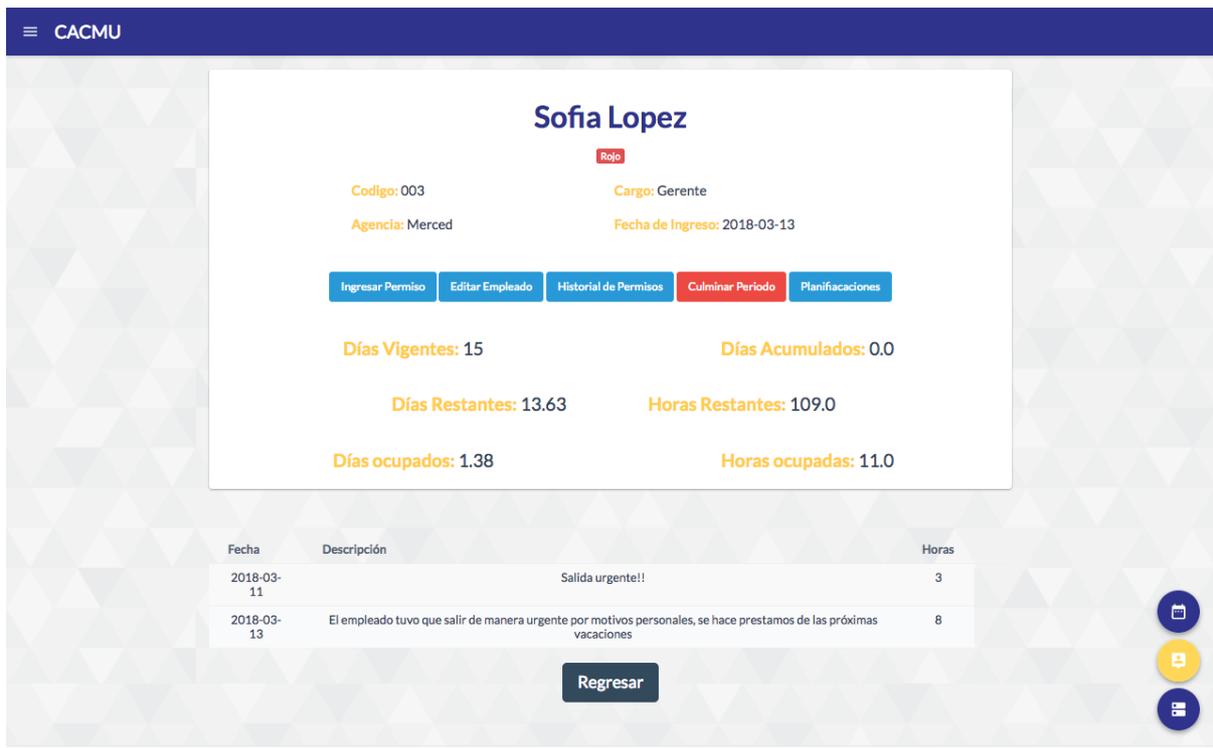


Fig. 76. Detalles de un empleado

☰ CACMU

Editar empleado

Nombre completo

Código del empleado

Agencia

Cargo

Fecha de ingreso

Fecha de calculo de vacaciones



Fig. 77. Formulario de edición de un empleado

☰ CACMU

Ingresar permiso para Sofia Lopez

Fecha de Permiso

Descripción

Número de Horas



Fig. 78. Formulario de ingreso de permisos de un empleado

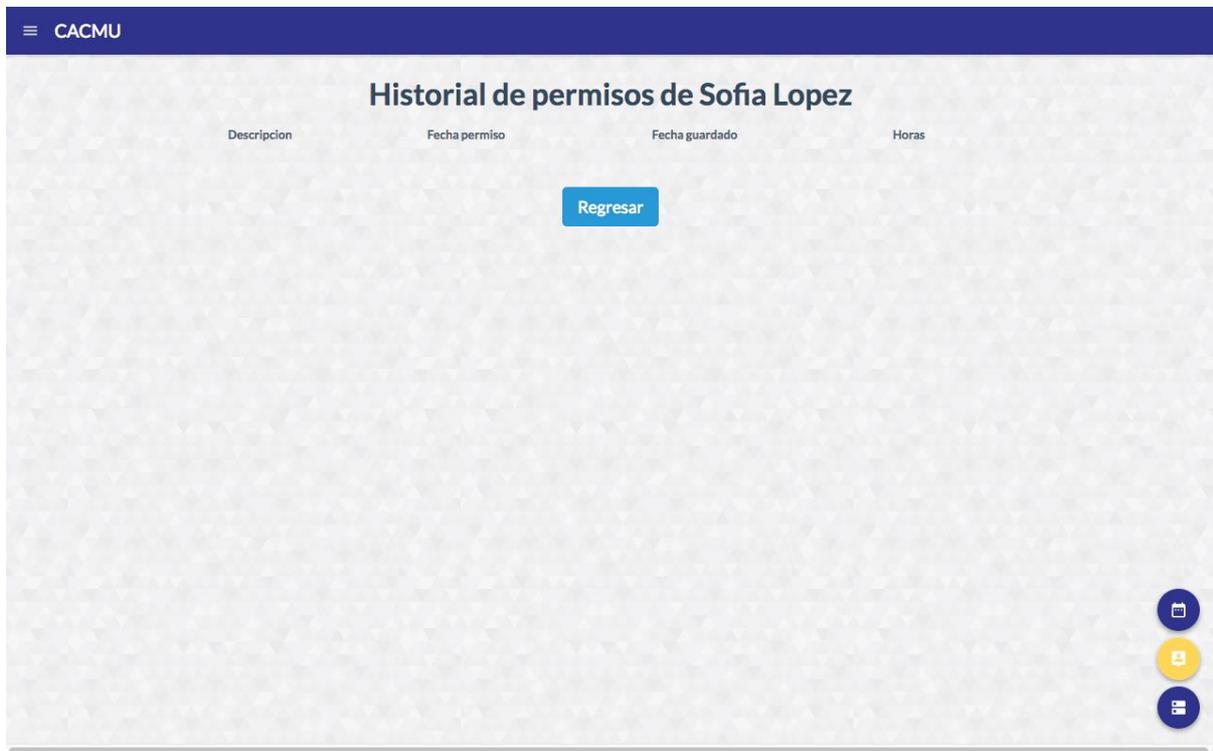


Fig. 79. Historial de permisos de un empleado

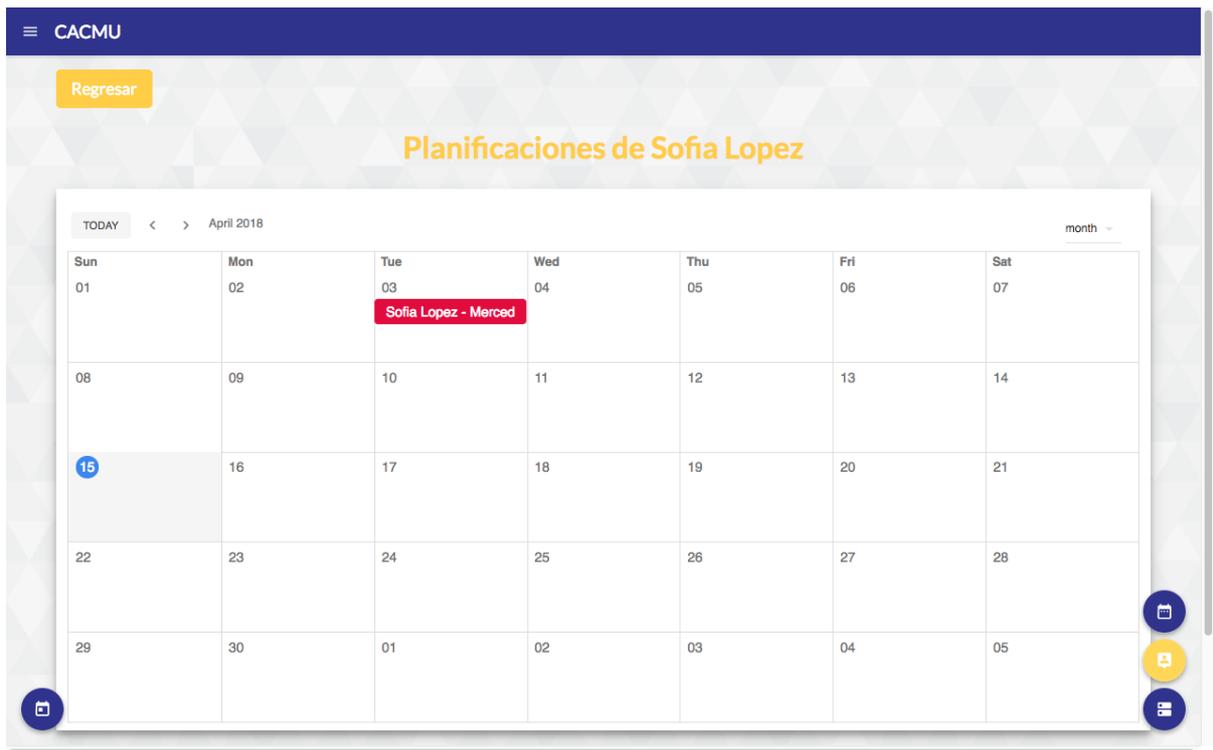


Fig. 80. Calendario de planificación de vacaciones de un empleado

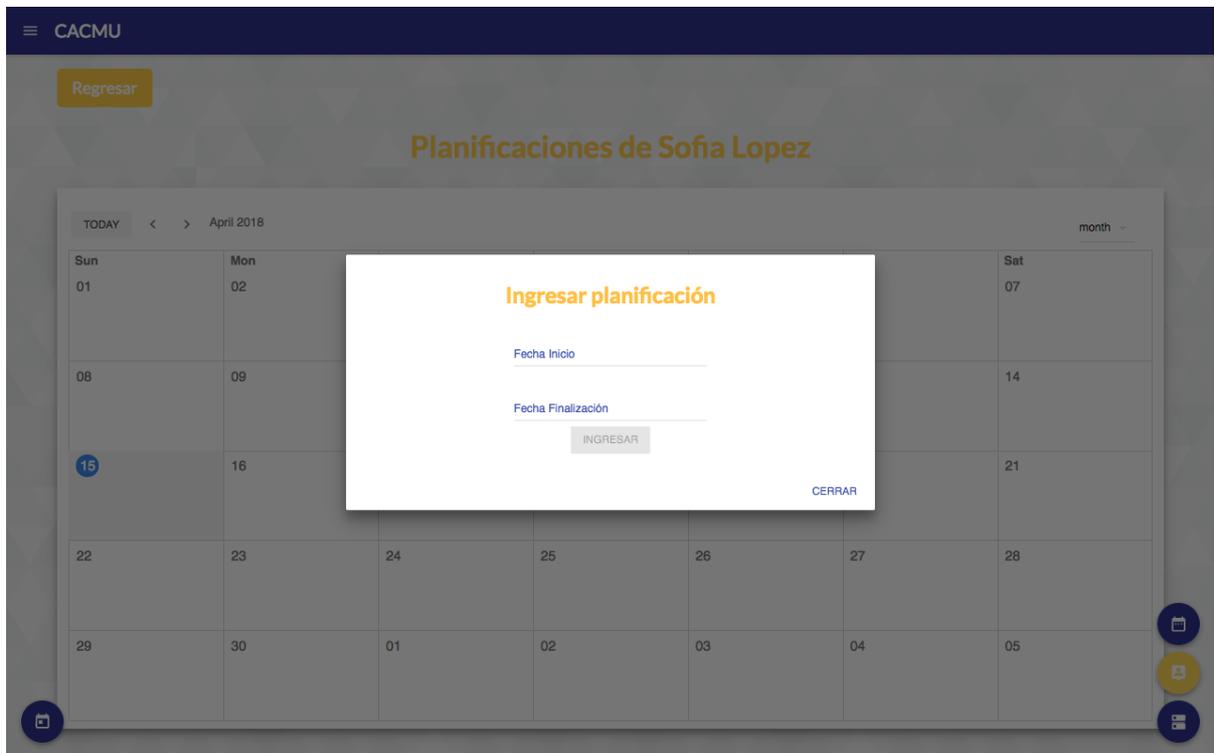


Fig. 81. Formulario de ingreso de planificación de vacaciones

Validación y revisión.

PLAN DE PRUEBA Y REPORTE DE EVALUACIÓN

Entidad: Cooperativa de Ahorro y Crédito Mujeres Unidas

Fecha: 17 de abril 2018

Mediante el presente informe se documenta la realización completa del plan de pruebas y evaluación del sistema para verificar y validar la funcionalidad correcta del sistema en base a los requerimientos solicitados por la cooperativa.

IDENTIFICACION DE LA HISTORIA	TAREA	ESTADO
sp52-01	Listado de los empleados de toda la cooperativa	Aprobado
sp52-02	Gestión de empleados de la cooperativa	Aprobado
sp52-03	Visualización del calendario global para mostrar las planificaciones de vacaciones	Aprobado
sp52-04	Calculo automático de vacaciones por empleado	Aprobado
sp52-05	Gestión de permisos pedidos por el empleado	Aprobado
sp52-06	Ingreso de planificaciones de permisos por empleado	Aprobado
sp52-07	Visualización del calendario de planificaciones por empleado	Aprobado
sp52-08	Culminaciones de periodos de vacaciones de empleados	Aprobado
sp52-09	Visualización y guardado de historiales de permisos por empleados	Aprobado
sp52-10	Privilegios de acceso de los usuarios de recursos humanos	Aprobado

CAPITULO 3

Evaluación de Resultados

El objetivo de este capítulo será la evaluación de resultados, cuyo objetivo es evaluar cuatro puntos fundamentales para el sistema, los cuales son: Funcionalidad, Satisfacción, Aceptación y Rendimiento.

3.1. Funcionalidad

Para evaluar la funcionalidad del aplicativo desarrollado, se contrastará las historias de usuario con la funcionalidad de los entregables resultantes de cada sprint, el cliente usará el aplicativo y podrá evaluar en porcentajes de 1% al 100% la funcionalidad que brinda el sistema con lo pedido en las historias de usuario.

TABLA 61: Evaluación de la funcionalidad del sistema

HISTORIA DE USUARIO	PORCENTAJE DE FUNCIONALIDAD
Autenticación de Usuarios en el sistema	100%
Gestión de usuarios y cuentas.	100%
Visualización de un <i>dashboard</i> de módulos del sistema	100%
Autorizar o rechazar juicios	100%
Visualización de créditos judiciales	100%
Semaforización y etapas estimadas	100%
Gestión de estado de juicios	100%
Detalles de los juicios	100%
Gráfica de procesos de juicios	100%
Gestión de abogados	100%
Gestión de subactividades de juicios	100%
Monitoreos	100%
Previsualización del monitoreo	100%
Visualización de resultados	100%
Buscador de juicios	100%
Reportes de monitoreos	100%
Convertir juicios de bienes a sin bienes y viceversa.	100%
Gestión de etapas judiciales	100%
Asignación de juicios a los asesores	100%
Clasificación de procesos de los juicios	100%
Eliminación de monitoreos	100%
Gestión de las tablas como administrador.	100%
Visualización de la planificación de la cartera por agencia y asesor	100%
Visualización de la cartera vencida por agencia y asesor	100%
Visualización de la eficiencia de la cartera por agencia y asesor	100%

HISTORIA DE USUARIO	PORCENTAJE DE FUNCIONALIDAD
Visualización de las matrices de riesgo por agencia y asesor	100%
Visualización de las cosechas de créditos por agencia y asesor	100%
Visualización de los indicadores de créditos vigentes por agencia y asesor	100%
Visualización de los indicadores de créditos colocados por agencia y asesor	100%
Privilegios de acceso al módulo de créditos	100%
Visualización de informes financieros por agencia	100%
Visualización de indicadores de la SEPS por agencia	100%
Privilegios de acceso al módulo de agencias	100%
Obtención del balance social de la cooperativa por agencias	100%
Privilegios de acceso al módulo de desempeño social	100%
Listado de los empleados de toda la cooperativa	100%
Gestión de empleados de la cooperativa	100%
Visualización del calendario global para mostrar las planificaciones de vacaciones	100%
Calculo automático de vacaciones por empleado	100%
Gestión de permisos pedidos por el empleado	100%
Ingreso de planificaciones de permisos por empleado	100%
Visualización del calendario de planificaciones por empleado	100%
Culminaciones de periodos de vacaciones de empleados	100%
Visualización y guardado de historiales de permisos por empleados	100%
Privilegios de acceso de los usuarios de recursos humanos	100%

Después de realizar la evaluación funcional del aplicativo a los clientes, se verifica que toda la funcionalidad del sistema cumple satisfactoriamente con cada historia de usuario solicitada.

3.2. Satisfacción

Para evaluar la satisfacción de los usuarios con respecto al aplicativo, el usuario se someterá a usar el aplicativo y contestar las siguientes preguntas con respecto a su satisfacción.

TABLA 62: Encuesta de satisfacción al cliente

SOFTWARE CACMU	EXCELENTE	BUENA	DEFICIENTE
¿El uso del sistema es simple?	x		
¿La ayuda que presenta el sistema es útil?	x		
¿La información se presenta de manera clara y entendible?	x		
¿La organización de los menús es lógica?	x		
¿Los mensajes de error son entendibles?	x		
¿La interfaz es amigable e intuitiva?	x		
¿La ejecución de cada tarea o proceso es complicada?	x		
¿El funcionamiento del sistema es entendible?		x	
¿Aprender a manejar el sistema fue fácil?		x	
¿La información entregada por el sistema es lo que realmente desea ver?	x		



Fig. 82. Gráfico satisfacción del cliente

Al sumar las calificaciones Excelentes y Buenas obtenemos un total del 100%, las calificaciones excelentes representan un 80% y las calificaciones buenas representan un 20%, por lo que se concluye que el cliente está satisfecho con el producto presentado.

3.3. Aceptación

Para evaluar la aceptación del cliente con el aplicativo, los usuarios tendrán que evaluar las historias de usuario en tres tipos de calificaciones: excelente, buena, ineficiente.

TABLA 63: Aceptación del aplicativo

HISTORIA DE USUARIO	EXCELENTE	BUENA	INEFICIENTE
Autenticación de Usuarios en el sistema	x		
Gestión de usuarios y cuentas.	x		
Visualización de un dashboard de módulos del sistema	x		
Autorizar o rechazar juicios	x		
Visualización de créditos judiciales	x		
Semaforización y etapas estimadas	x		
Gestión de estado de juicios	x		
Detalles de los juicios	x		
Gráfica de procesos de juicios	x		
Gestión de abogados	x		
Gestión de sub actividades de juicios	x		
Monitoreos	x		
Previsualización del monitoreo		x	
Visualización de resultados	x		
Buscador de juicios	x		
Reportes de monitoreos		x	
Convertir juicios de bienes a sin bienes y viceversa.	x		
Gestión de etapas judiciales	x		
Asignación de juicios a los asesores	x		
Clasificación de procesos de los juicios	x		
Eliminación de monitoreo	x		
Gestión de las tablas como administrador.	x		
Visualización de la planificación de la cartera por agencia y asesor	x		
Visualización de la cartera vencida por agencia y asesor	x		
Visualización de la eficiencia de la cartera por agencia y asesor	x		
Visualización de las matrices de riesgo por agencia y asesor	x		
Visualización de las cosechas de créditos por agencia y asesor	x		
Visualización de los indicadores de créditos vigentes por agencia y asesor	x		
Visualización de los indicadores de créditos colocados por agencia y asesor	x		
Privilegios de acceso al módulo de créditos	x		
Visualización de informes financieros por agencia	x		
Visualización de indicadores de la SEPS por agencia	x		
Privilegios de acceso al módulo de agencias	x		
Obtención del balance social de la cooperativa por agencias	x		
Privilegios de acceso al módulo de desempeño social	x		
Listado de los empleados de toda la cooperativa		x	
Gestión de empleados de la cooperativa	x		
Visualización del calendario global para mostrar las planificaciones de vacaciones	x		
Calculo automático de vacaciones por empleado	x		
Gestión de permisos pedidos por el empleado	x		
Ingreso de planificaciones de permisos por empleado	x		
Visualización del calendario de planificaciones por empleado	x		
Culminaciones de periodos de vacaciones de empleados	x		
Visualización y guardado de historiales de permisos por empleados	x		
Privilegios de acceso de los usuarios de recursos humanos	x		



Fig. 83. Gráfico de aceptación del aplicativo

Si sumamos los resultados buenos y excelentes obtenemos una aceptación del 100%, los excelentes representan un 93% y los buenos un 7%. Se concluye que el cliente acepta el aplicativo entregado.

3.4. Rendimiento

Para evaluar el rendimiento del aplicativo se utilizará la herramienta JMeter, la cual nos ayudará a la obtención de los detalles de las peticiones, tiempos de carga, latencia, mediana, línea 90%, línea 95%, rendimientos, tiempos mínimos y tiempos máximos de carga de una muestra.

Para el análisis de los datos obtenidos nos basaremos en el estándar APDEX, el cual es un estándar abierto desarrollado por una alianza de compañías que define un método estandarizado para informar, comparar y rastrear el rendimiento de las aplicaciones. Apdex es una medida numérica de la satisfacción del usuario con el rendimiento de las aplicaciones empresariales. Convierte muchas mediciones en un solo número en una escala uniforme de 0 a 1 (0 = sin usuarios satisfechos, 1 = todos los usuarios satisfechos).

Las pruebas que realizaremos las dividiremos en dos partes. La primera parte serán las pruebas realizadas al módulo de cobranzas judiciales y al módulo de recursos humanos, los cuales trabajan únicamente con la base de datos postgresQL de la aplicación y se podrá medir su rendimiento exacto de la aplicación, la segunda parte de las pruebas se las realizará al módulo de créditos, desempeño social y agencias, los cuales trabajan con la base de datos Oracle de la cooperativa.

Para las pruebas de carga se trabajarán con los siguientes tiempos, los cuales son necesarios para el cálculo del estándar APDEX.

- Satisfecho: Tiempo de respuesta menor o igual a 2 segundos.
- Tolerante: Tiempo de respuesta entre 2 segundos y 4 segundos.
- Frustrante: tiempo de respuesta superior a 4 T segundos.

Para las pruebas del módulo de agencias, créditos y desempeño social se tomará el tiempo de carga desde que la base de datos Oracle entregue los datos al aplicativo, debido a que las consultas de datos desde Oracle hacia el aplicativo son controladas por la empresa proveedora de software de la cooperativa VIMACORP y son ajenas al desempeño propio del aplicativo.

Primero configuraremos JMeter con las peticiones de debe realizar al aplicativo, junto con los headers, caché, cookies y parámetros adecuados para la aceptación de la petición. Se probarán todos los módulos de la aplicación, configuraremos pruebas de 20 usuarios concurrentes que es el estimado de uso concurrente de la aplicación y generaremos el archivo JTL con las configuraciones.

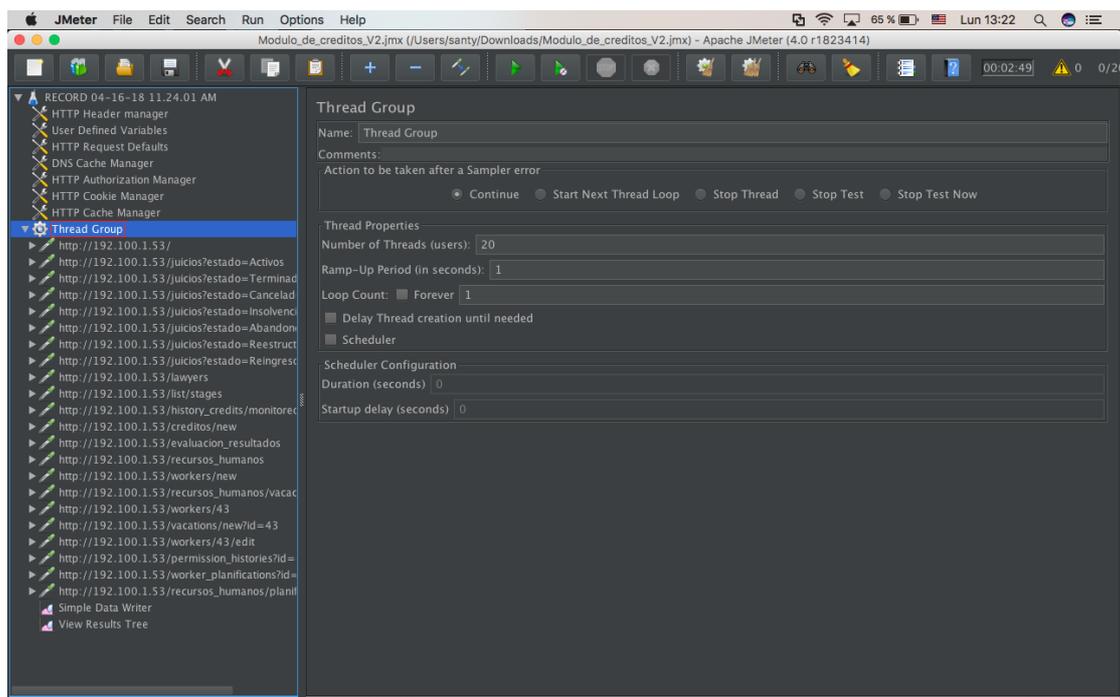


Fig. 84. Configuración del JMeter primera parte

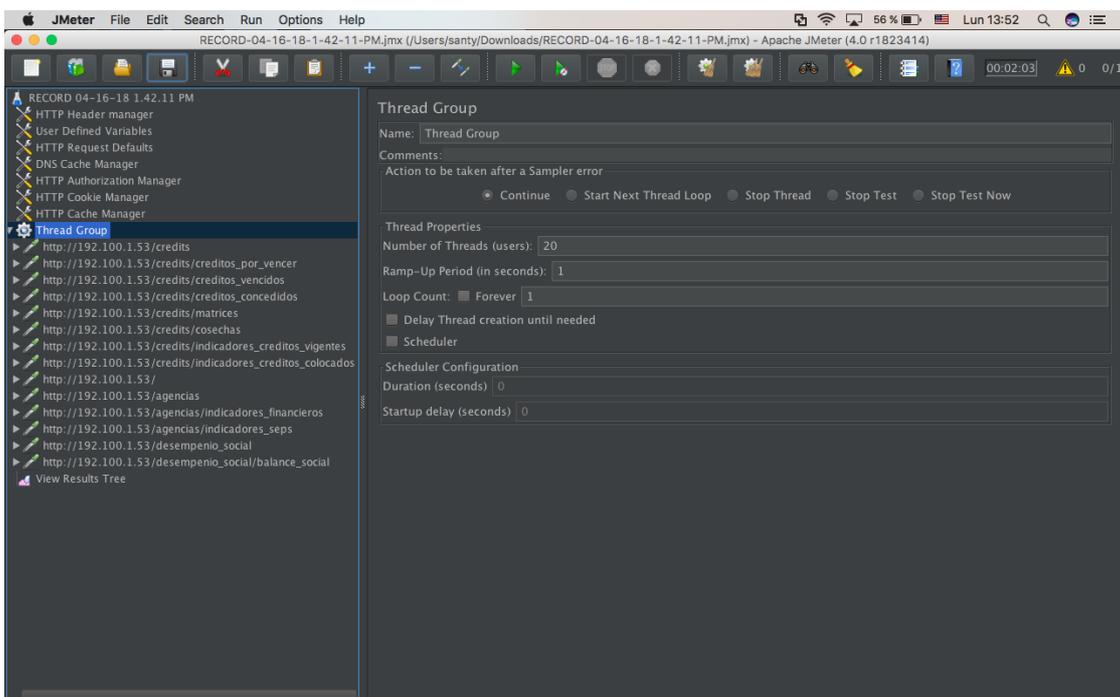


Fig. 85. Configuración del JMeter segunda parte

JMeter desde su versión 3.0 y superiores nos permite extraer un reporte HTML con el estándar APDEX, lo que debemos configurar es el umbral de aceptación y el umbral de tolerancia para las pruebas de rendimiento que ya establecimos con anterioridad para los dos tipos de pruebas.

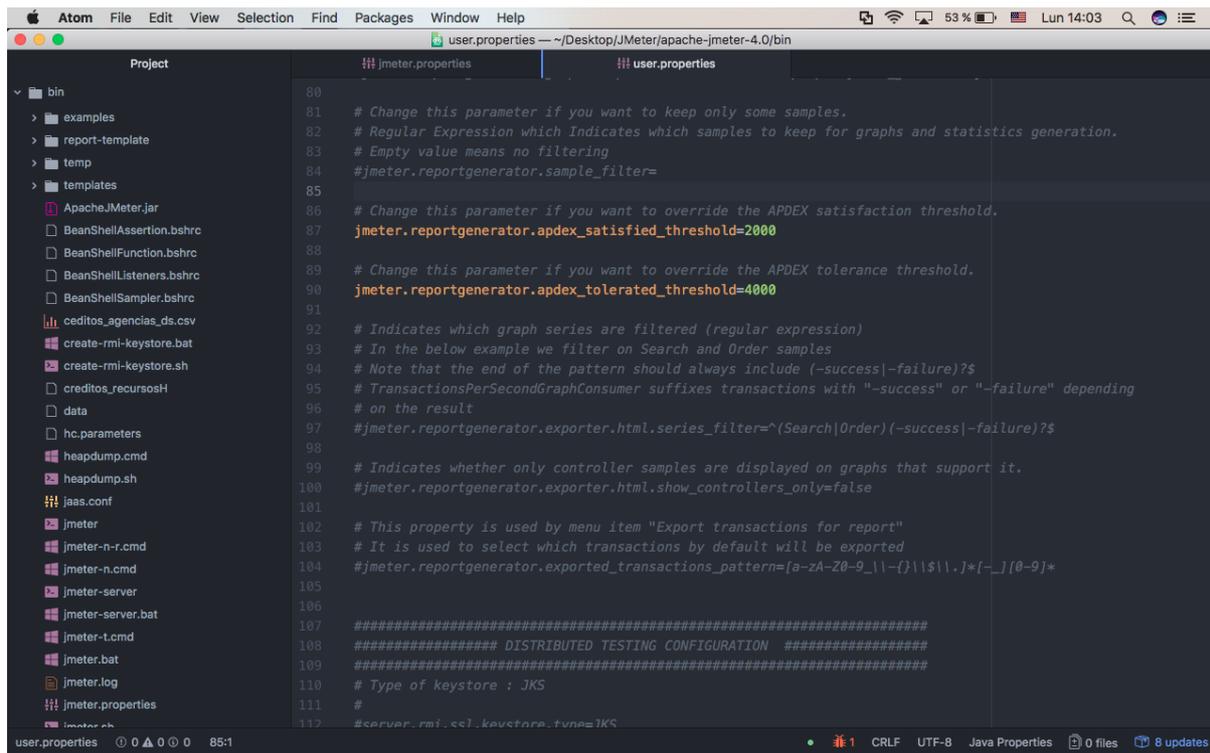


Fig. 86. Configuración de tiempos JMeter primera parte

Realizadas estas configuraciones se ejecutaron los análisis de rendimiento para la elaboración los reportes HTML con el estándar APDEX, y se obtuvo los siguientes resultados.

TABLA 64: Resultados Apdex de JMeter módulo de créditos y recursos humanos

Apdex	T (Toleration threshold)	F (Frustration threshold)	Label
0.974	2 sec	4 sec	Total
0.425	2 sec	4 sec	http://192.100.1.53/
1.000	2 sec	4 sec	http://192.100.1.53/list/stages
1.000	2 sec	4 sec	http://192.100.1.53/permission_histories?id=43
1.000	2 sec	4 sec	http://192.100.1.53/juicios?estado=Abandono
1.000	2 sec	4 sec	http://192.100.1.53/lawyers
1.000	2 sec	4 sec	http://192.100.1.53/history_credits/monitoreo
1.000	2 sec	4 sec	http://192.100.1.53/recursos_humanos/planificacion_general
1.000	2 sec	4 sec	http://192.100.1.53/creditos/new
1.000	2 sec	4 sec	http://192.100.1.53/juicios?estado=Insolvencia
1.000	2 sec	4 sec	http://192.100.1.53/recursos_humanos/vacaciones
1.000	2 sec	4 sec	http://192.100.1.53/juicios?estado=Terminado
1.000	2 sec	4 sec	http://192.100.1.53/juicios?estado=Cancelado
1.000	2 sec	4 sec	http://192.100.1.53/juicios?estado=Reingreso

Apdex	T (Toleration threshold)	F (Frustration threshold)	Label
1.000	2 sec	4 sec	http://192.100.1.53/recursos_humanos
1.000	2 sec	4 sec	http://192.100.1.53/juicios?estado=Activos
1.000	2 sec	4 sec	http://192.100.1.53/workers/43
1.000	2 sec	4 sec	http://192.100.1.53/evaluacion_resultados
1.000	2 sec	4 sec	http://192.100.1.53/vacations/new?id=43
1.000	2 sec	4 sec	http://192.100.1.53/workers/43/edit
1.000	2 sec	4 sec	http://192.100.1.53/worker_planifications?id=43
1.000	2 sec	4 sec	http://192.100.1.53/workers/new
1.000	2 sec	4 sec	http://192.100.1.53/juicios?estado=Reestructurado

TABLA 65: Resultados Apdex de JMeter segunda parte

Apdex	T (Toleration threshold)	F (Frustration threshold)	Label
0.963	2 sec	4 sec	Total
0.475	2 sec	4 sec	http://192.100.1.53/credits
1.000	2 sec	4 sec	http://192.100.1.53/agencias/indicadores_seps
1.000	2 sec	4 sec	http://192.100.1.53/credits/creditos_por_vencer
1.000	2 sec	4 sec	http://192.100.1.53/desempenio_social
1.000	2 sec	4 sec	http://192.100.1.53/agencias
1.000	2 sec	4 sec	http://192.100.1.53/credits/indicadores_creditos_vigentes
1.000	2 sec	4 sec	http://192.100.1.53/
1.000	2 sec	4 sec	http://192.100.1.53/credits/indicadores_creditos_colocados
1.000	2 sec	4 sec	http://192.100.1.53/agencias/indicadores_financieros
1.000	2 sec	4 sec	http://192.100.1.53/credits/matrices
1.000	2 sec	4 sec	http://192.100.1.53/credits/creditos_vencidos
1.000	2 sec	4 sec	http://192.100.1.53/credits/cosechas
1.000	2 sec	4 sec	http://192.100.1.53/credits/creditos_concedidos
1.000	2 sec	4 sec	http://192.100.1.53/desempenio_social/balance_social

Después de obtener las tablas de resultados de las dos pruebas se observa que, en la primera prueba realizada al módulo de cobranzas judiciales y recursos humanos el indicador Apdex total indica un 97.4% de satisfacción del rendimiento del aplicativo, una petición a la aplicación tiene un índice menor al 50%, todas las demás pruebas de carga obtuvieron un valor de 1 por lo que concluye que los usuarios concurrentes tuvieron experiencias positivas con el tiempo de carga de la aplicación, al analizar los resultados de la de las pruebas a los módulos de créditos, agencias y desempeño social se observa que el indicador Apdex indica un 96.3% de satisfacción del rendimiento del aplicativo, que una petición a la aplicación tiene un índice menor al 50%, por lo que se da por aprobado las pruebas de rendimiento de la aplicación.

CAPITULO 4

Conclusiones y Recomendaciones

Después de haber concluido el proyecto en su totalidad cuyo objetivo principal era “Desarrollar un sistema web para la Cooperativa de Ahorro y Crédito Mujeres Unidas en la cual se encuentren automatizados todos los procesos de gestión judicial y obtención de indicadores de gestión mediante la utilización del *Framework* Ruby on Rails y la librería ReactJs con la metodología Scrum” se puede garantizar que se entrega a la cooperativa el producto de software que satisface el problema planteado una vez cumplido el objetivo del proyecto al 100%.

Del desarrollo del proyecto podemos extraer las siguiente conclusiones y recomendaciones:

4.1. Conclusiones

- ✓ Después de realizar el proyecto utilizando la metodología de desarrollo ágil scrum se determinó que esta metodología es ideal cuando nos enfrentamos a entornos de desarrollos inciertos, cuando el cliente no tiene claro los requerimientos que necesita para el aplicativo, esta metodología al estar basada en iteraciones y al no tener un *product backlog* fijo hace posible que se implementen nuevos cambios durante el desarrollo del proyecto.

- ✓ La metodología scrum en determinados escenarios nos ayuda a aumentar la calidad del software al entregar en cada iteración mejoras y correcciones al sistema detectados en iteraciones pasadas, a diferencia de otras metodologías tradicionales en el que se basa a una generación exhaustiva de documentos para obtener los requerimientos del usuario y se realiza únicamente la entrega del software terminado después de un periodo de tiempo, en estas metodologías si el cliente no tuvo claro sus necesidades al inicio de la metodología el software tendrá falencias en sus procesos y su manera de actuar disminuyendo la calidad y beneficio que se entrega al cliente.

- ✓ Scrum como metodología ágil ayudó a mantener el interés del cliente durante todo el proceso de desarrollo, ya que en cada iteración se fue entregando una parte del aplicativo 100% funcional, validando la información y mejorando los procesos planteados al principio del desarrollo, ayudó a que desde la culminación del primer

sprint el cliente pueda utilizar en su totalidad el primer módulo ayudando a la satisfacción del cliente con el producto final.

- ✓ El sistema mejoró el rendimiento interno de la cooperativa al entregar reportes necesarios que antes de la implementación del sistema el personal debía elaborarlos en hojas de Excel y les tomaba un tiempo de dos horas realizarlos, ahora mediante el aplicativo lo pueden obtener en segundos.
- ✓ Mediante la planificación de cartera la cooperativa logró mejorar la productividad y recuperación de cartera por parte del personal al tener un fácil acceso a las metas de recuperación del mes.
- ✓ Con la implementación del sistema la cooperativa mejoró el control de los procesos judiciales, las jefaturas de cobranzas ahora pueden manejar de manera eficiente los procesos judiciales y dar mayor importancia a créditos judiciales que están próximos a vencer, esto reditúa económicamente a las jefaturas de cobranzas, las jefaturas de créditos ahora pueden monitorear los procesos de todos los juicios y autorizar el inicio del proceso de juico de manera rápida a los créditos que cumplan con los requisitos.
- ✓ Con el uso del *framework* Ruby on Rails se disminuyó de manera evidente los tiempos de desarrollo del aplicativo, al estar basado en convención sobre configuración como principio, y al tener una extensa gama de librerías de Ruby denominadas *RubyGems* nos proporcionó funcionalidades completas como el manejo de usuarios con solo instalar una gema en el proyecto.
- ✓ Ruby on Rails en todo el proceso de elaboración del proyecto nos ayudó a usar buenas prácticas de desarrollo y de escritura de código, gracias a las convenciones del *framework*, se obtuvo una estructura de ficheros y código más amigable y entendible.
- ✓ React ayudó a simplificar el proceso de mostrar datos al usuario, mediante librerías de Javascript se logró implementar muchas funcionalidades del sistema, como descarga de tablas HTML a hojas de Excel, implementación de las guías de diseño de google con solo usar componentes de React, programación asíncrona mediante las promesas de javascript y actualización automática de datos gracias a los *states* de React y su optimización mediante el virtual DOM.

- ✓ Se cumplió el objetivo de la utilización de React al tener interfaces orientadas a componentes, los componentes de navegación, tablas, y formularios se basaron en un componente que se reutilizó para cada caso específico, la redundancia de código se redujo y se aumentó la facilidad de escalamiento de la aplicación.

4.2. Recomendaciones

- ✓ Se recomienda el uso de la metodología Scrum cuando el cliente no tenga claro los requerimientos completos del aplicativo, se puede empezar con una base sólida de requerimientos e ir incrementando el *product backlog* conforme vaya evolucionando el proyecto.
- ✓ Se recomienda el uso de Ruby on Rails cuando se necesita que un proyecto web salga a producción lo más pronto posible, ya que por su amplio listado de *RubyGems* se puede acelerar de manera considerable un desarrollo del proyecto.
- ✓ Juntamente con análisis del *product owner* en un futuro se recomienda la implementación de la herramienta de elaboración de los POA.
- ✓ Se recomienda tener un *backup* semanal de la base de datos del aplicativo para prevenir un posible escenario de pérdida de información o daño de equipos.
- ✓ Se recomienda el cambio de contraseña de los usuarios cada 2 meses, para garantizar la privacidad de datos que cruzan mediante la aplicación, y la implementación de procesos internos de la cooperativa para pedir a un administrador cambio de contraseñas o información de usuario y tablas.
- ✓ Se recomienda la integración del este sistema con un aplicativo extra para mejorar el módulo de cobranzas judiciales de tal manera que pueda mostrar direcciones de socios y mediante el aplicativo monitorear las visitas de los asesores de cobranzas a los socios.
- ✓ Para otros proyectos que se quiera implementar en la cooperativa, se recomienda utilizar el mismo modelo de evaluación de resultados ya que centra al usuario en obtener su satisfacción y aceptación del producto.

- ✓ En la fase de diseño del ciclo de vida del software se recomienda buscar librerías y herramientas que pueda ayudar al cumplimiento de las historias de usuario, ya que en este tipo de tecnologías la comunidad crea bastantes herramientas que nos facilitan la implementación de funcionalidades, esto ayudará a implementar soluciones probadas y disminuir tiempos de desarrollo.
- ✓ Se recomienda el uso de un gestor de control de versiones como Git y Github, esto ayuda bastante en el *deploy* de aplicaciones y en el control de errores entre versiones del aplicativo.
- ✓ Al finalizar el proyecto se recomienda el uso de un *framework* de *deploy* de aplicaciones como el utilizado en este proyecto (Capistrano), ya que si bien esto lleva más tiempo la configuración de estos *frameworks* se obtendrá una gran ventaja al poder *deployar* siguientes versiones con solo un comando.

GLOSARIO

Abreviaturas

- ❖ **CACMU:** Cooperativa de Ahorro y Crédito Mujeres Unidas
- ❖ **CSS:** (Cascade Style Sheet) Hoja de estilos en cascada
- ❖ **SPA:** (Single Page Application) Aplicaciones de una página
- ❖ **RoR:** Ruby on Rails
- ❖ **SEPS:** Superintendencia de Economía Popular y Solidaria
- ❖ **ORM:** (Object Relational Mapping) Mapeo Relacional de Objetos
- ❖ **DOM:** (Document Object Model) Modelo de Objetos de Documento.
- ❖ **IMAP:** (Internet Message Protocol) Protocolo de Mensajes de Internet
- ❖ **SSL:** (Secure Socket Layer) Capa de Punteros Seguros

Definiciones

- ❖ **Front-End:** Capa de una aplicación encargada de la presentación de datos a un usuario final.
- ❖ **Back-End:** Capa de una aplicación que se encarga de la lógica de negocios y acceso a datos.
- ❖ **Framework:** Es un conjunto estandarizado de conceptos, herramientas y prácticas para resolver un problema en particular.

- ❖ **Aplicación Web:** Herramienta de software en la cual, los usuarios pueden acceder a través de un servidor web mediante internet o una intranet utilizando un navegador.
- ❖ **Open Source:** El Código abierto se refiere a cualquier programa o herramienta que su código fuente se expone de manera libre a disposición del público para su uso o modificación.
- ❖ **Script:** Es un programa simple que se encuentran en un archivo de texto, el cual es interpretado línea a línea para ejecutar procedimientos secuenciales.
- ❖ **Plugin:** Un plugin es una librería o una porción de código que aumenta las funcionalidades de un programa principal en el que se ejecuta.
- ❖ **Clases:** Las clases en términos de programación es un mecanismo que se utiliza para representar los objetos de la vida real como: vehículos, edificios etc.
- ❖ **IDE:** Un IDE es un entorno de desarrollo integrado, en el que se reúnen un conjunto de herramientas en un mismo programa que son útiles a los desarrolladores para agilizar ciertas tareas.
- ❖ **Transpilación:** Es un proceso en el cual se transforma un Código de alto nivel a otro código de alto nivel cambiando su sintaxis.
- ❖ **DOM:** El DOM es la representación de la estructura de objetos que un navegador crea al cargar una página web, el cual puede ser modificado mediante JavaScript.
- ❖ **Web Sockets:** Es un canal de comunicación entre un cliente y un servidor bidireccional que se mantiene abierto mientras el cliente esté conectado.
- ❖ **Sql:** Es un lenguaje de consultas estructuradas que nos permite recolectar información específica de una base de datos.
- ❖ **Scope:** El scope es lo que limita el acceso y la visibilidad de las variables de programación en determinadas partes del código.

REFERENCIAS

- Acens. (2013). Servidor web Nginx, una clara alternativa a Apache. *Alcobendas*, 6.
Retrieved from www.acens.com
- Andalucia, J. (n.d.). Patrón Modelo Vista Controlador. Retrieved April 17, 2018, from <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/122>
- Baquero, J. (2017). ¿Cuáles son las principales ventajas de ReactJS? Retrieved May 13, 2018, from <https://www.arsys.es/blog/programacion/caracteristicas-reactjs/>
- Cadena, N., & López, S. (2018a). *Manual interno CACMU - herramienta I*. Ibarra.
- Cadena, N., & López, S. (2018b). *Manual interno Cacmu - herramienta II*. Ibarra.
- Catlin, H., Weizenbaum, N., & Eppstein, C. (n.d.). Sass, el manual oficial. Retrieved April 17, 2018, from <https://librosweb.es/libro/sass/>
- Comprasoft. (2018). RubyMine. Retrieved April 17, 2018, from <https://comprasoft.com/jetbrains/rubymine>
- Dorantes, C. (2015). Qué es PostgreSQL y cuáles son sus ventajas. Retrieved April 17, 2018, from <https://platzi.com/blog/que-es-postgresql/>
- Gackenheim, C. (2015). *Introduction to React*. New York: Apress.
- Giampiere Julio. (n.d.). ¿Qué es JavaScript? Retrieved April 17, 2018, from <https://devcode.la/blog/que-es-javascript/>
- Google. (2014). Elevation & shadows - Material Design. Retrieved April 18, 2018, from <https://material.io/guidelines/material-design/elevation-shadows.html#elevation-shadows-elevation-android>
- Hernandez, U. (2015a). MVC (Model, View, Controller). Retrieved April 17, 2018, from <https://codigofacilito.com/articulos/mvc-model-view-controller-explicado>
- Hernandez, U. (2015b). Por qué aprender Ruby. Retrieved April 17, 2018, from <https://codigofacilito.com/articulos/por-que-aprender-ruby>
- JetBrains. (2018). Testing Assistant - Features | RubyMine. Retrieved April 17, 2018, from https://www.jetbrains.com/ruby/features/ruby_unit_testing.html
- Jiménez, A. (2015). Estudio de servidor web: NGINX. Retrieved April 17, 2018, from <https://adrianjimenezb.wordpress.com/2015/01/23/estudio-de-servidor-web-nginx/>
- Jimenez, J. (2016). *Descubre React*. Leanpub.
- Maquetando. (2015). SASS y LESS : El presente de CSS. Retrieved April 17, 2018, from <http://maquetando.com/sass-y-less-el-presente-de-css>
- Mejia, A. (2011). Ruby on Rails Architectural Design. Retrieved April 18, 2018, from <http://adrianmejia.com/blog/2011/08/11/ruby-on-rails-architectural-design/>
- Menzinsky, A., López, G., & Palacio, J. (2016). Scrum Manager Guía de formación, 2.6, 94.
Retrieved from http://scrummanager.net/files/sm_proyecto.pdf

mozilla developer. (2017). JavaScript. Retrieved April 17, 2018, from <https://developer.mozilla.org/es/docs/Glossary/JavaScript>

Netcraft. (2018). Web Server Survey. Retrieved April 18, 2018, from <https://news.netcraft.com/archives/2018/03/27/march-2018-web-server-survey.html>

Niksinski, K. (2013). *Instant Haml*. Birmingham: Packt Publishing.

Pérez, E. (2014). ¿Qué es Material Design? Retrieved April 17, 2018, from <https://elandroidelibre.elespanol.com/2014/11/que-es-material-design.html>

PostgreSQL. (n.d.). PostgreSQL: About. Retrieved April 17, 2018, from <https://www.postgresql.org/about/>

Rauschmayer, A. (2014). *Speaking Javascript* (first edit). California.

RubyOnRails. (n.d.). Acerca de Ruby. Retrieved April 17, 2018, from <http://www.ruby-lang.org/es/about/>

Schwaber, K., & Sutherland, J. (2017). The Scrum Guide™. Retrieved from <http://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf#zoom=100>

Soni, R. (2016). *Nginx from beginner to pro*. Berkeley: Apress.

Thomas, D., Fowler, C., & Hunt, A. (2013). *Programming Ruby 1.9 & 2.0*. Dallas, Texas.

ANEXOS

ANEXO 1: MANUAL DE USUARIO (EN CD)

ANEXO 2: PLANIFICACION DEL PROYECTO (EN CD)

ANEXO 3: ARCHIVOS DE EVALUACIÓN DE RENDIMIENTO (EN CD)