

UNIVERSIDAD DEL VALLE

**AgroInd - Sistema de información para el
registro y análisis de indicadores de la
metodología de sostenibilidad agroecologica
propuestos por la asociación asoprovegas-IAF en
la zona rural de Tuluá Valle**

por

Oscar Eduardo Ramirez Arevalo, Juan sebastian Velasquez Valencia.

Director: Ingeniero Royer David Estrada Esponda

Una tesis presentada en cumplimiento parcial para el
grado de Ingeniero en sistemas

en la

Facultad de Ingeniería

Escuela de Ingeniería de Sistemas y Computación

Abril 2017

“Sólo podemos ver poco del futuro, pero lo suficiente para darnos cuenta de que hay mucho que hacer”

Alan Turing.

Nota de aceptación:

Firma del coordinador

Firma del jurado 1

Tuluá, Abril de 2017

Resumen

La asociación Asoprovegas, en la zona rural de Tuluá, ha llevado a cabo el proceso de evaluación de indicadores de la Metodología de sostenibilidad Agroecológica, con los que se busca el mejoramiento de la calidad de vida de los habitantes de la región en los aspectos que abarca la metodología (ambiental, social y económico). Actualmente la asociación cuenta con un significativo número de asociados, situación que ha hecho que los resultados de las mediciones tarden más en consolidarse ya que no se contaba con un sistema de información que permitiera la manipulación y el almacenamiento de los datos obtenidos en las fincas evaluadas.

AGROIND es la solución a la carencia de un sistema de información que se adapte a la Metodología de sostenibilidad agroecológica; haciendo uso de nuevas tecnologías de desarrollo de software que permitieron la implementación de una aplicación -que cuenta con múltiples clientes (Web, Escritorio y móvil)- en la que se provee un ambiente de trabajo que satisface las necesidades que se presentan en el proceso de evaluación de indicadores.

Abstract

Asoprovegas is an association that has carried out the process of evaluation of indicators of the Methodology of Agroecological sustainability, in Tulua's rural area, which seeks to improve the quality of life of the habitants of the region in the aspects covered by the methodology (environmental, social and economic). At the moment the association has a significant number of associates, a situation that has made the results of the measurements take longer to consolidate since there was no information system that allowed the manipulation and storage of the data obtained in the farms evaluated.

AGROIND is the solution to the lack of an information system that adapts to the methodology of agroecological sustainability; Using new software development technologies that allowed the implementation of an application which has multiple clients (Web, Desktop and Mobile) which provides a work environment that meets the needs that are presented in the process of evaluate the indicators of the methodology...

Incide

Resumen	iii
Abstract	iv
Lista de Figuras	vii
Lista de Tablas	viii
Abreviaciones	ix

1 Introducción.	1
1.1 Descripción General	1
1.2 Estructura del documento	2
1.3 Problema	3
1.3.1 Descripción del problema	3
1.3.2 Formulación del problema	3
1.4 Objetivos	4
1.4.1 Objetivo General	4
1.4.2 Objetivos Específicos	4
1.5 Alcance	4
2 Marco Referencial	6
2.1 Marco teórico:	6
2.1.1 Metodología:	6
2.1.2 Metodología de indicadores de sostenibilidad agroecológica:	6
2.1.3 ¿Qué es un sistema?	10
2.1.4 ¿Qué es información?	10
2.1.5 Sistema de información	11
2.2 Marco conceptual	12
2.2.1 Asociación	12
2.2.2 Asociación campesina	12
2.2.3 Kanban.	12
2.2.4 Kanban en el desarrollo de software.	14
2.2.5 Bases de datos relacionales.	14
2.2.6 Bases de datos no relacionales.	14
2.2.7 MVC	15

2.2.8	ORM:	15
2.2.9	Migraciones:	16
2.2.10	API.	16
2.2.11	REST.	16
2.3	Antecedentes	17
2.3.1	Antecedentes comerciales	17
2.3.2	Antecedentes academicos:	19
3	Desarrollo y descripción del proyecto	21
3.1	Descripción del sistema	21
3.2	Herramientas utilizadas	21
3.2.1	Herramientas de desarrollo de software:	22
3.2.2	Herramientas de control y gestión del proyecto:	24
3.3	Desarrollo del proceso de ingeniería de software	25
3.3.1	Sprint 1:	26
3.3.2	Sprint 2:	29
3.3.2.1	Funciones con desconexión:	32
3.3.3	Sprint 3:	35
3.3.3.1	Estado de alerta:	37
3.3.4	Sprint 4:	39
4	Conclusiones y trabajos futuros.	43
4.1	Conclusiones.	43
4.2	Trabajos futuros.	44
	Anexos	45
	Referencias	46

Lista de Figuras

2.1	Calificación de indicadores.	9
2.2	Gráfica de red. Fuente: elaboración propia a partir de la librería ChartsJs. . . .	9
2.3	Kanban Card. Fuente: Tomado de [11].	13
2.4	Kanban. Fuente: Tomado de [11].	14
3.1	Diagrama de componentes general de la aplicación. Fuente: elaboración propia.	22
3.2	Interfaz base de la aplicación. Fuente: Elaboración propia, haciendo uso de [46]	28
3.3	Formato de casos de prueba. Fuente: Elaboración propia.	28
3.4	Formulario manual para evaluación de indicadores. Fuente: Tomado de anexos.	30
3.5	Interfaz de usuario para evaluación de indicadores. Fuente: Elaboración propia, utilizando [46]	31
3.6	Interfaz de usuario para Evaluación de indicadores, versión móvil. Fuente: Elaboración propia, utilizando [46]	31
3.7	gráfica de penetración de internet a nivel nacional. Fuente: tomado [47]	32
3.8	gráfica de penetración de internet en el Valle del Cauca. Fuente: tomado [47] .	33
3.9	Diagrama de flujo de funciones con desconexión. Fuente: Elaboración propia, usando [45]	34
3.10	Diagrama de base de datos. Fuente: elaborado con Rail-ERD.	37
3.11	Diagrama de clases, controladores. Fuente: elaborado con RailRoady.	41
3.12	Diagrama de clases, modelos. Fuente: elaborado con RailRoady.	42

Lista de Tablas

2.1	Análisis de antecedentes comerciales.	19
3.1	Historias de usuario para el Sprint1.	26
3.2	Historias de usuario para el Sprint2.	29
3.3	Historias de usuario para el Sprint3.	35
3.4	Historias de usuario para el Sprint4.	40

Abreviaciones

API	A plicacion P rogramming I nterface
REST	R Epresentational State T ransfer
JS	J ava S cript
OS	O perative S ystem
MVC	M odel V iew C ontroller
ORM	O bject- R elational M apping
AWS	A mazons W eb S ervices

–Dedicado al Principal de los ingenieros. A Edinson Ramirez y Amparo Arevalo, mis padres, por ser ejemplo de perseverancia y constancia aunque estas lineas sean pocas para agradecer mucho. A la memoria de Rodrigo Ramirez, quien con su ingenio sorprendía a su pequeño pueblo. - O. Ramirez.

–Dedicado a mi familia, especialmente a mis padres José Vicente y Luz Stella y a mi tío José Raúl por su constante e incondicional apoyo y por ser los pilares que me han sostenido siempre. También dedico este trabajo de grado a mis amigos y a cada una de las personas que de una u otra forma creyeron en mí. - S. Velasquez.

–Un sentimiento especial de gratitud hacia nuestros profesores, guías indispensables en este proceso, principalmente a Royer David Estrada por guiarnos en esta última etapa de formación y a Pavel Franco que por sus enseñanzas confirmamos que estábamos “dónde realmente queríamos estar”. O. Ramirez S.Velasquez.

...

Capítulo 1

Introducción.

1.1 Descripción General

La metodología de sostenibilidad agroecológica fue diseñada por ingenieros ambientales con el fin de evaluar unos indicadores o métricas con las que se busca medir el rendimiento de una finca a través del tiempo y de la ejecución de un proyecto de acompañamiento y dirección a los campesinos y sus tierras en los diversos campos evaluados, como son: cuidado del medio ambiente (agua y suelos), producción mensual de cultivos, estado de la vivienda, finanzas y demás. La metodología busca fomentar el desarrollo de los propietarios de pequeñas fincas hacia un enfoque de mejoramiento en la calidad de vida, identificando los factores a optimizar para aumentar la productividad de estas y los factores ambientales que favorecen directamente al área del proyecto e indirectamente favorecen al ecosistema en general. El proyecto se llevó a cabo en la zona rural de Tuluá, más precisamente con las fincas afiliadas a la asociación Asoprovegas.

La metodología de análisis de los sistemas productivos (agrícola, social y económico) mediante los indicadores de sostenibilidad en pequeñas fincas, ha sido aplicada en diversas regiones y procesos agroecológicos en el país. Una de las experiencias más representativas es la de Asproinca (Asociación de Productores Indígenas y Campesinos) en Riosucio Caldas desde la década de los 90, arrojando un importantísimo avance en las unidades de producción evaluadas [1].

El objetivo de esta metodología no es simplemente llevar registro de las buenas condiciones de la situación socioeconómica y agroecológica de una finca evaluada, con el uso de estos indicadores busca también aprovechar los resultados negativos para fortalecer al campesino en alguno de los aspectos que esté fallando. La metodología busca generar aprendizaje a partir de la interpretación crítica de la información que surge del proceso; este aprendizaje debe ser

fácilmente comprensible, apropiable y pertinente para generar cambios de actitud que permitan un mejor desarrollo, en el paisaje y en los contextos sociales que se intervienen. Los propósitos de la sistematización están dirigidos a:

- a) Mejorar la propia práctica.
- b) Compartir los aprendizajes con otras experiencias similares.
- c) Para contribuir al enriquecimiento de la teoría.[1].

La siguiente es una visualización de la Agroecología:

“La Agroecología explora, por una parte, el conjunto de relaciones ecológicas y culturales que suceden al interior y al exterior de los campos de cultivos, abarcando incluso la integralidad de las fincas y su necesaria interconexión en patrones de los que se denomina la estructura ecológica principal del paisaje y por otra, las conexiones e interacciones complejas que existen entre los agroecosistemas y la sociedad.” [4].

1.2 Estructura del documento

El presente documento es el resultado del proceso de ingeniería de software, realizado a con el fin de suplir la problemática presentada por Asoprovegas, para optar al título de ingeniero de sistemas; este documento está compuesto por 4 capítulos distribuidos de la siguiente manera:

- En el capítulo 1 se presenta la introducción del documento, en la que se habla acerca de la metodología de sostenibilidad a través de los indicadores agroecológicos, los objetivos de esta y la labor que lleva a cabo la asociación Asoprovegas en la zona rural de Tuluá, la descripción y formulación del problema; así como los objetivos y los resultados esperados en el proceso de desarrollo de este trabajo de grado; también contiene el alcance propuesto para este proyecto.
- En el capítulo 2 se presenta el marco teórico, conceptual y los antecedentes con el que se fundamenta este trabajo.
- En el Capítulo 3 se presenta el Desarrollo del proyecto y la descripción del proceso de ingeniería de software realizado
- En el Capítulo 4 se presentan las conclusiones, producto de este trabajo y posibles trabajos futuros con respecto al tema acá abordado.

1.3 Problema

1.3.1 Descripción del problema

Asoprovegas, una asociación de pequeñas y medianas fincas, ha aplicado la metodología de sostenibilidad en 30 fincas de la zona rural de Tuluá. En el proceso de evaluación se ha identificado que la evaluación de los indicadores requiere demasiado tiempo; una de las razones es que debido a la distancia entre las fincas, el desplazamiento entre ellas no es fácil; adicional a esto, y siendo el principal problema, el proceso de calificación se vuelve lento ya que la asociación no cuenta con una forma de administrar la información de esta cantidad de fincas, así como generar consultas de los resultados para un posterior informe. Hasta el momento se ha intentado resolver la problemática utilizando una plantilla de Excel o formularios de Google para registrar la ficha de calificación de cada una de las fincas, pero esto genera una cantidad de archivos en el que sigue siendo complicado el manejo de información. Un inconveniente que se presenta con los formularios de Google es la poca cobertura del servicio de internet en la zona rural de Tuluá, haciendo esta opción poco viable al momento de recolectar información. El problema también radica en que es muy difícil analizar la información recolectada de manera grupal, para generar datos estadísticos de todo el proceso de la metodología en la zona, datos como:

- Promedio del resultado por indicador.
- Selección de la mejor calificación.
- Informe de alertas de bajo rendimiento. entre otros.

1.3.2 Formulación del problema

¿Cómo ayudar a la asociación Asoprovegas a llevar a cabo una mejor gestión de la información de sus proyectos de sostenibilidad agroecológica y contribuir a la reducción de tiempo de ejecución de estos, así como a la facilidad de gestionar usuarios e informes resultantes de estas tareas, cubriendo problemas técnicos como la poca cobertura del servicio de internet que impida la implementación de una aplicación web convencional?

1.4 Objetivos

1.4.1 Objetivo General

Implementar un sistema de información web que permita sistematizar la metodología usada en la asociación Asoprovegas para el registro y gestión de los indicadores agroecológicos, económicos y sociales.

1.4.2 Objetivos Específicos

1. Estudiar la metodología de sostenibilidad agroecológica a sistematizar, con el fin de comprender su ejecución y las herramientas que esta necesita.
2. Ejecutar el proceso de ingeniería de software, acorde a la metodología de desarrollo elegida, para implementar una aplicación web que permita la sistematización de la metodología de sostenibilidad agroecológica.
3. Integrar al sistema de información WEB la API para clientes móviles y de escritorio desarrollada en el proceso de ingeniería de software.

1.5 Alcance

El sistema de información producto de este trabajo de grado es una aplicación web que brinda las funcionalidades necesarias para llevar a cabo el proceso de calificación de los indicadores de la metodología de sostenibilidad agroecológica; implementando una API que permita la integración con un cliente de ordenador y un cliente móvil, así como la posibilidad de operar en el cliente sin conexión a internet y la posterior actualización en la web al momento de que este se conecte nuevamente. A continuación se presenta las diferentes funcionalidades que posee la aplicación de acuerdo al nivel de usuarios:

- Generales para todo usuario:
 - Posee un usuario y una contraseña.
 - Tendrán acceso a una sección de información personal.
- Digitadores (encargados de ingresar información al sistema):
 - Tienen acceso al módulo de calificaciones y se les permite diligenciar la calificación de los indicadores de los usuarios evaluados (beneficiarios).

- Pueden acceder al módulo de resultados y generar un reporte de evaluación para los usuarios evaluados.
- Evaluadores (ingenieros ambientales):
 - Tienen acceso al módulo de calificaciones y se les permite diligenciar la calificación de los indicadores de los usuarios evaluados (beneficiarios).
 - Pueden registrar el análisis y las recomendaciones en una calificación.
 - Pueden acceder al módulo de resultados y generar un reporte de evaluación para los usuarios evaluados.
 - Tienen acceso al módulo de estadística y generar reportes en esta sección.
- Beneficiario:
 - Posee un usuario y contraseña.
 - Puede acceder al resultado de su evaluación y generar un reporte de este
- Administrador:
 - Posee todas las funcionalidades anteriormente mencionadas.
 - Se encarga de gestionar usuarios y perfiles.
 - Controla el registro de usuarios a la aplicación.

Capítulo 2

Marco Referencial

2.1 Marco teórico:

2.1.1 Metodología:

Según la definición que ofrece la RAE, una metodología es ” un conjunto de métodos que se siguen en una investigación científica o en una exposición doctrinal”[2]; un método, según la misma entidad, es un modo de hacer o procedimiento que tiene como finalidad llegar a un resultado verídico y convertirlo en conocimiento [3].

2.1.2 Metodología de indicadores de sostenibilidad agroecológica:

Desde la década de 1980 se ha promovido el concepto de sustentabilidad o sostenibilidad con el fin de hacerle frente al deterioro del área rural y la necesidad de desarrollo que esta presenta, así como la toma de medidas contra el deterioro del medio ambiente; también abordando temas ligados a esta problemática como la inequidad social a nivel global. Las múltiples definiciones que se le ha dado a sostenibilidad coinciden en el objetivo de conservación de los recursos naturales, la búsqueda de la igualdad social y la búsqueda de estrategias para el mejoramiento del mercado y la economía de una determinada región [5].

A partir de lo mencionado anteriormente se dice que la sostenibilidad agroecológica es la ciencia dedicada al estudio de agro ecosistemas y la manera ambientalmente segura de transformarlos en sistemas de producción agropecuaria que conserven el equilibrio ambiental [5].

Como preocupación particular, la agroecología se fundamenta en el conocimiento campesino e indígena ancestral, ya que el aspecto social que se toma en cuenta para esta, vela por la conservación y apropiación de la cultura de los campesinos, considerada ancestral por los desarrolladores de la metodología; la agroecología toma ese conocimiento ancestral y lo convierte en un mecanismo que permite innovar las tareas en el campo con el fin de mejorar la productividad y a su vez la conservación del medio ambiente [5].

Uno de los principales problemas de la aplicación de esta metodología es el siguiente: “En Colombia varios programas agroecológicos han dedicado muchos esfuerzos a la promoción del desarrollo sostenible; sus métodos de evaluación de resultados se basan más en los aspectos técnicos de su implementación y menos en los impactos que estas actividades han generado en cuanto a la sostenibilidad de las formas de vida de la gente que participa en ellos. Se carece de herramientas que les permitan medir qué tanto han avanzado en los objetivos de sostenibilidad propuestos en los programas”. [5].

Alvarado también identifica que en esta metodología hace falta implementar métodos que permitan aumentar la eficiencia de la evaluación a partir de la planificación, que permita una mejor retroalimentación en el proceso en que se usa la metodología.

“El procedimiento metodológico empleado para orientar el trabajo de evaluación propuesto en este texto está basado principalmente en las propuestas del MESMIS (Metodología para la Evaluación de Sistemas de Manejo Incorporando Indicadores de Sustentabilidad) adelantada por el Grupo Interdisciplinario de Tecnología Rural (GIRA) en México (Astier, Masera, 1996 - 1997) y en la propuesta del IICA (De Camino, Müller, 1993) desarrollada en Costa Rica” [5]. Esta metodología centra el interés en el campesino y en sus actividades, ya que es el objetivo principal del proyecto; por otra parte, los evaluadores, asesores, asistentes y demás toman el rol de “facilitadores”, personas encargadas de hacer un puente entre el campesino y el exterior, gestionando los recursos, la tecnología y/o el asesoramiento que requiere el campesino para mejorar en los aspectos planteados en los indicadores a evaluar.

Los indicadores que evalúa la metodología surgen a partir de los aspectos críticos identificados por los evaluadores; estos son diferenciados en 3 grandes grupos que son: económico, social y ambiental. Los indicadores ambientales, se enfocan en el estado del ecosistema y de cómo las acciones de los campesinos afectan positiva o negativamente a este; los indicadores a tener en cuenta son:

- Manejo de suelos.
- Manejo de aguas.

El grupo más grande de indicadores es el de indicadores económicos, ya que es a partir de estos aspectos individuales que se puede evaluar el rendimiento de manera cuantificable de una finca; con estos se busca medir el auto consumo de sus productos (seguridad alimentaria) y la capacidad de comerciar (soberanía alimentaria) con estos, también se busca evaluar el estado financiero de la producción y la capacidad de mano de obra con la que el campesino cuenta. Los indicadores a tener en cuenta son:

- Componente pecuario/ auto consumo.
- Componente pecuario/ mercado
- Componente agrícola/ auto consumo.
- Componente agrícola/ mercado.
- Disponibilidad de mano de obra.
- Manejo administrativo.
- Dependencia de insumos externos.

Los indicadores de aspecto social, buscan medir el comportamiento y la interacción que tienen los campesinos con respecto a su comunidad; estos indicadores se consideran ya que la interacción social fomenta el desarrollo al compartir conocimiento y al formar alianzas y colaboraciones; los indicadores de esta sección son:

- Compromiso familiar.
- Estado de la vivienda.
- interacción comunitaria.

La evaluación de estos indicadores se debe representar de manera cuantificable; para construir la escala de medición, se propone una escala sencilla de cero a cinco, donde cero es el nivel menos deseable en ese aspecto para la sostenibilidad o el punto crítico extremo y cinco es el nivel óptimo o deseable para el mismo indicador. Desde cero, y de manera gradual, se irán formulando niveles intermedios que signifiquen los pequeños pasos que se deberían ir logrando hasta alcanzar el nivel óptimo formulado [5].

La Figura 2.1 es un ejemplo propuesto por acevedo para ilustrar el proceso de cuantificación del indicador desde aspectos que pueden ser subjetivos:

Cabe anotar que en el resultado se manejan “diagramas de radar”[Figura 2.2] donde se evidencia gráficamente el resultado final de cada evaluación.

Dimensión social:

Nombre del indicador	Concepto	Escala de Medición
Autosuficiencia alimentaria	Es la cantidad suficiente, la calidad orgánica, la diversidad autóctona y disponibilidad permanente de alimentos para la familia	0. Cero producción de alimentos (todo se compra). 1. Monocultivo agropecuario que genera una fuente de alimento permanente aunque no orgánico. 2. Monocultivo agropecuario que genera una fuente de alimento permanente aunque no orgánico, y, otro u otros cultivos que son fuente de alimento ocasional no orgánico. 3. Produce orgánicamente hasta un 50% de los alimentos que puede producir en su predio. 4. Produce orgánicamente hasta un 80% de los alimentos que puede producir en su predio. 5. Abastecimiento de todo lo que es posible producir en su predio y 100% orgánico

Dimensión económica:

Nombre del indicador	Concepto	Escala de Medición
Rentabilidad	Ganancia económica derivada del sistema de producción, mediante la reducción de costos y la integración vertical (transformación de productos)	0. No genera ingresos y hay endeudamiento. 1. Pérdidas altas y refinanciación de deudas. 2. Pérdidas moderadas y no hay deudas. 3. Punto de equilibrio: no hay ganancias y no hay pérdidas. 4. Generación de ganancia moderada y hay producción primaria. 5. Se genera ganancia alta, hay procesos agroindustriales.

FIGURA 2.1: Calificación de indicadores.

Fuente: tomado de [5].



FIGURA 2.2: Gráfica de red. Fuente: elaboración propia a partir de la librería ChartsJs.

2.1.3 ¿Qué es un sistema?

-Conjunto de cosas que relacionadas entre sí ordenadamente contribuyen a determinado objetivo [6].

Piattini [7] a partir de la definición ofrecida por la RAE, genera el análisis de lo que es un sistema, considerándolo como un conjunto e individualizando sus elementos de la siguiente manera:

- **Componentes** o partes del sistema.
- **Relaciones** entre componentes, que determinan la **estructura** del sistema.
- **objetivo** o propósito del sistema [[7], p 4].

Teniendo en cuenta que los sistemas poseen un conjunto de entradas y salidas, se presentan dos elementos más en su composición:

- **Entorno:** Todo aquello que rodea al sistema o donde esté ubicado. Este puede ser un sistema mucho más grande (macro-sistema/micro-sistema o sistema/subsistema).
- **Limites:** La frontera entre lo que es el sistema y lo que constituye el entorno [[7], p 4].

2.1.4 ¿Qué es información?

Los **datos** están constituidos por los registros de los hechos, acontecimientos, transacciones, etc. Por el contrario, la **información** implica que los datos estén procesados de tal manera que resulten útiles o significativos para el receptor de los mismos. En cierto modo, los datos se pueden considerar la materia prima para obtener información [[7],p 5].

La información, para que sea útil, debe presentar las siguientes características.

- **Relevante.**
- **Precisa.**
- **Completa.**
- **Adecuada.**
- **Generada a tiempo.**
- **Nivel de detalle adecuado.**
- **Comprensible** (pág 9,10 [7]).

2.1.5 Sistema de información

El concepto de Sistema de información, derivado de los conceptos presentados anteriormente se puede presentar de la siguiente manera.

- Independientemente del uso de las tecnologías de información (TI) los sistemas de información existen desde que se creó la primera organización humana; una estructura compuesta por un conjunto de personas distribuidas en departamentos o áreas dividiendo las funciones de operación de la organización y compartiendo información entre si [[7],pp 10-11].

Una definición puntual es la siguiente:

- Un conjunto formal de procesos que, operando sobre una colección de datos estructurada según las necesidades de la empresa, recompilan, elaboran y distribuyen la información(o parte de ella) necesaria para las operaciones de dicha empresa y para las actividades de dirección y control correspondientes(decisiones), para desempeñar su actividad de acuerdo a su estrategia de negocio [[7],p 12].

Teniendo en cuenta la forma en que Piattini [7] abordó los sistemas como conjuntos, un sistema de información está compuesto por:

- **Procedimientos y prácticas habituales de trabajo:**
Aquellas actividades que desempeña una organización o empresa relacionadas a su modelo de negocio, actividades productivas y demás áreas que compongan la entidad.
- **Información:** Datos organizados en contexto y analizados para su posterior aprovechamiento.
- **Personas o usuarios:** Son el recurso humano dentro del sistema de información, encargados de generar los datos del sistema y de la interacción con este.
- **Equipo de soporte:** Es el "organo" encargado de la comunicación entre todos los componentes del sistema, haciendo uso de diversos dispositivos y métodos, con el objetivo de procurar el buen funcionamiento del sistema (pág 13,14 [7]).

2.2 Marco conceptual

2.2.1 Asociación

Es un conjunto de asociados(integrantes) con un mismo fin o propósito y que forman una persona jurídica [8].

2.2.2 Asociación campesina

En Colombia, se considera como "Asociación campesina" A la figura jurídica formada a partir de un grupo de un conjunto de productores agrícolas y agropecuarios. Estas asociaciones son reguladas a partir del DECRETO 2716 DE 1994 de carácter nacional, el cual establece el marco jurídico para estas[9]. Textualmente, en este decreto se define a una asociación campesina como:

- Para los fines del presente Decreto se entiende por asociación agropecuaria la persona jurídica de derecho y sin ánimo de lucro, constituida por quienes adelantan una misma actividad agrícola, pecuaria, forestal, piscícola y acuícola con el objeto de satisfacer o defender los intereses comunes de sus asociados y contribuir al desarrollo del sector rural nacional. Se entiende por asociación campesina aquella organización de carácter privado constituida por campesinos, y que tenga como objeto principal la interlocución con el Gobierno en materia de reforma agraria, crédito, mercadeo y asistencia técnica agropecuaria... [9] Artículo 2].

2.2.3 Kanban.

Es un sistema de origen japonés de organización de trabajo para procesos de manufactura, ideado y desarrollado por la compañía Toyota en la década del 40 [10].

Este sistema de trabajo nace como respuesta a las circunstancias negativas que rondan a Japón frente al mercado mundial; teniendo en cuenta que este país, históricamente, no ha contado con grandes reservas de recursos naturales, lo que obliga a la importación de materiales, generando así una desventaja competitiva con regiones como Europa o América Latina; sin embargo, Kanban se apropia de características muy marcadas de la cultura japonesa, cómo lo son: conciencia grupal, sentido de igualdad, deseo de mejorar; entre otras cualidades culturales que abogan por la calidad en sus proyectos y productos así como un ambiente laboral en el que la responsabilidad es compartida y no recae en un solo individuo según su jerarquía [10].

La palabra japonesa Kanban se traduce como "tablero", aunque por su uso frecuente en los procesos empresariales se ha convertido en sinónimo de la programación de demanda. Esta

Part Number:	80800-14898
Part Name:	10 V Power Supply
Production Line:	Line A
Container Type:	Plastic 12 × 14
Container Quantity:	20
Storage Location:	Portable Radio Line
Production Operation:	50
Bin Location:	C-3

FIGURA 2.3: Kanban Card. Fuente: Tomado de [11].

”Programación Kanban” consiste en la implementación de unas señales visuales que permiten a los operadores determinar cuanto se ha recorrido en un proceso y cuando se culmina este; Entre las reglas de Kanban también se especifican pasos a seguir, en caso de presentar problemas de ejecución, así como a quién acudir en estas situaciones; contribuyendo esto a mejoras en tiempo de ejecución y por lo tanto en costos operacionales.[11]

Con el uso de señales visuales y el conjunto de reglas de decisión, Kanban reemplaza el tradicional programa semanal, presente en muchas otras metodologías de organización, convirtiéndose así en una herramienta de ejecución más que en una herramienta de planificación. El uso de esta reemplaza por completo son las actividades de programación diarias y reduce la necesidad de que los planificadores de producción y los supervisores revisen continuamente el estado del plan para determinar el siguiente punto de producción y cuándo cambiar este.

Gross y McInnis [11] mencionan lo siguiente acerca de Kanban:

- ...Piense en la programación kanban como una herramienta de ejecución en lugar de una herramienta de planeación. El kanban, que puede tomar muchas formas, dirige la operación del proceso en el día a día. La programación Kanban no reemplaza la planificación material, sino que toma la información del planeamiento material y la usa para crear el kanban...[11].

El diseño y la implementación de Kanban consiste en identificar los procesos que se deben ejecutar, la estimación del tiempo en cuanto al cambio, tiempo de ejecución y tiempo de inactividad; también se establecen las reglas de ejecución y la asignación de responsabilidades en el equipo de producción. Eventualmente se elaboran las ”Kanban cards” [Figura 2.3], el elemento visual en el que se centra esta herramienta; estas tarjetas consisten en un consolidado de la información requerida para una tarea [11].

Estas tarjetas son puestas en un tablero de actividades[Figura 2.4] que representan el proceso de producción de inicio a fin, lo que contribuye a el entendimiento del estado de producción con un simple vistazo.



FIGURA 2.4: Kanban. Fuente: Tomado de [11].

2.2.4 Kanban en el desarrollo de software.

Ante la actual demanda y la dinámica en el ambiente tecnológico, se han buscado alternativas para reducir costos y tiempos de producción para satisfacer los continuos cambios en los requerimientos proporcionados por el cliente. Esta situación ha llevado a la implementación de Kanban en el proceso de ingeniería de software, clasificada en las denominadas "metodologías ágiles". El uso de Kanban en el desarrollo de software se ha basado principalmente en el éxito que este ha presentado en la industria manufacturera; adicional a esto, las características de adaptabilidad, cambios en los requisitos y visualización constante del estado de producción, hacen de esta herramienta una alternativa viable ante las nuevas exigencias del mercado [12].

2.2.5 Bases de datos relacionales.

Es un tipo de base de datos cuya organización se basa en el modelo relacional propuesto por E. F. Codd en 1970 [13]. El modelo organiza los datos en tablas (relaciones) de columnas y filas, con una llave como identificador único de cada fila; las filas también son llamadas registros o tuplas [15]. Generalmente cada tabla representa un tipo de Entidad, las filas representan instancias de esa entidad y las columnas representan valores pertenecientes a esta instancia.

2.2.6 Bases de datos no relacionales.

Conocidas también como bases de datos NoSQL, hacen referencia a un sistema de gestión de base de datos no relacional, este sistema no se construye principalmente sobre Tablas y la

manipulación de datos no se realiza utilizando SQL. Los sistemas NoSQL son bases de datos distribuidas y no relacionales diseñadas para el almacenamiento de datos a gran escala y para el procesamiento masivo de datos en paralelo a través de un gran número de servidores de productos básicos. También utilizan lenguajes y mecanismos no-SQL para interactuar con datos [14].

2.2.7 MVC

Es un patrón de arquitectura de software donde se separan los datos de la lógica de negocio, la interfaz y el módulo encargado de controlar los eventos y comunicaciones. MVC propone la creación y uso de 3 componentes que son el modelo, la vista y el controlador [16].

- **Modelo:** Es la representación de la información con la cual el sistema opera, por lo tanto gestiona todos los accesos a dicha información, tanto consultas como actualizaciones. Envía a la 'vista' aquella parte de la información que en cada momento se le solicita para que sea mostrada. Las peticiones de acceso o manipulación de información llegan al 'modelo' a través del 'controlador' [17].
- **Controlador:** Responde a eventos y realiza peticiones al 'modelo' cuando se hace alguna solicitud sobre la información. También puede enviar comandos a la 'vista' asociada si se solicita un cambio en la forma en que se presenta el 'modelo' , se podría decir que el 'controlador' hace de intermediario entre la 'vista' y el 'modelo' [17].
- **Vista:** Presenta el 'modelo' en un formato adecuado para interactuar por parte del usuario [17].

Aunque originalmente MVC fue desarrollado para aplicaciones de escritorio, ha sido ampliamente adoptado como arquitectura para diseñar e implementar aplicaciones web debido a que promueve la reutilización de código y la posibilidad de dividir el software en componentes separados facilitando la depuración y realización de pruebas. En la actualidad se han desarrollado innumerable cantidad de frameworks para construir aplicaciones usando este patrón ayudando a mejorar la productividad de los desarrolladores.

2.2.8 ORM:

Es una técnica para convertir datos entre sistemas de tipo incompatibles en lenguajes de programación orientados a objetos. Esto crea, una "base de datos de objetos virtual" que se puede utilizar dentro del lenguaje de programación. Existen varios patrones de diseño; uno de los más usados es ActiveRecord donde una tabla de la base de datos es encapsulada en una clase. Así

una instancia de la clase es una fila en la tabla. Después de la creación de un objeto, se crea una nueva fila en la tabla. Cualquier objeto cargado, obtiene información de la fila a la que está atado. Cuando se actualiza un objeto también se actualiza la información en la fila de la tabla.

2.2.9 Migraciones:

Se refiere a la administración y manejo de cambios incrementales a un esquema de base de datos relacional. Una migración se lleva a cabo cada vez que se debe actualizar el esquema o revertir este a una versión anterior.

Las migraciones se realizan mediante una herramienta de migración de esquema. Cuando se invoca con una versión de esquema específica, la herramienta automatiza la aplicación sucesiva o reversión de una secuencia apropiada de los cambios de esquema hasta que es llevado al estado deseado. En otras palabras es un sistema de control de versiones para bases de datos.

2.2.10 API.

se define como a API la descripción de un componente de software en términos de sus operaciones, entradas, salidas y tipos de datos. En otras palabras un API es un componente de software que permite la comunicación de una aplicación con otra [18].

2.2.11 REST.

Es un estilo de arquitectura propio de sistemas web que basa su funcionamiento en el protocolo HTTP. Rest es usado gracias a la escalabilidad que le aporta a un servicio web gracias a ciertas características clave como un protocolo cliente/servidor sin estado donde cada mensaje http contiene la información suficiente para completar la petición, un conjunto de operaciones definidas que aplican a todos los recursos de información, una sintaxis universal para identificar los recursos donde estos solo son accesibles a través de su URI. Un concepto importante en REST es la existencia de recursos (elementos de información), que pueden ser accedidos utilizando un identificador global (un Identificador Uniforme de Recurso) [19].

2.3 Antecedentes

2.3.1 Antecedentes comerciales

Actualmente se han desarrollado diversas herramientas de software para el sector rural; varias de ellas buscan orientar al propietario de una finca hacia un nivel administrativo más sofisticado con el fin de optimizar la producción, la economía y las finanzas de esta.

Actualmente no existe un software orientado a la metodología de sostenibilidad agroecológica tratada en este planteamiento; salvo algunas herramientas como las plantillas de Excel o formularios de Google, las cuales pueden servir para gestionar la administración de la información de los indicadores que necesita Asoprovegas. A continuación se presentan algunos sistemas de software agroecológicos con similitudes al trabajo de la metodología; cabe resaltar que estos son de uso comercial:

- **Daruma:** Software para Gestión ambiental, a partir de la norma ISO 14001. Este software sirve para gestionar una serie de indicadores, muy similares a los indicadores de sostenibilidad agroecológica, con la diferencia que estos corresponden a los indicadores impuestos en la norma ISO 140001; su campo de acción es en el medio ambiente, la salud y la demografía[20].
- **Sismagro:** Software que permite llevar un control y generar simulación de cultivos y demás procesos agrícolas. Este software tiene similitud con la metodología ya que le permite al usuario conocer el estado en que se encuentra su productividad y el desempeño de sus cultivos [21].
- **AgroWin:** Software contable y administrativo, enfocado al campo. Gracias a este enfoque administrativo, el usuario se mantiene informado de su estado productivo y puede tomar decisiones con respecto a sus procesos de producción [22].
- **Agroptima:** Aplicación móvil para gestionar actividades de cultivos y demás tareas del campo, así como generar mapas de los cultivos; permite trabajar en la aplicación sin conexión y almacena toda la información en el perfil web del usuario [23].
- **Synagro:** Software que posee un módulo administrativo y un módulo contable que permite gestionar ambos campos de acción; en el módulo administrativo se encuentran las funcionalidades para gestionar aspectos de las fincas como agricultura, ganadería y maquinaria; en el módulo contable se encuentran los aspectos financieros a tener en cuenta como lo son: impuestos, cuentas corrientes y capital disponible [24].
- **EGam:** Software de gestión ambiental, basado en la norma ISO 14001; muy similar a Daruma, cumple la misma funcionalidad de gestión para la norma ambiental [25].

- **Aranda:** Dashboard para la gestión de indicadores de la infraestructura tecnológica en una compañía; permite diseñar los indicadores que desea para dicha labor y ofrece una presentación de resultados de manera muy gráfica en una interfaz con un buen nivel de usabilidad, lo que reduce costos al ser una herramienta que no requiere personal especializado para su uso [26].

Teniendo en cuenta que los indicadores de sostenibilidad agroecológica con los que Asoprogas ha realizado sus mediciones, abordan 3 aspectos a evaluar: económico, social y ambiental; se realizó un análisis de los servicios y funcionalidades que los software mencionados que puedan proveer una solución para el proceso de evaluación de los indicadores.

Hay que recordar que la lista de 12 indicadores de sostenibilidad agroecológica a tener en cuenta son los siguientes:

- Ind 1: Manejo de suelos.
- Ind 2 Manejo de aguas.
- Ind 3 Componente pecuario/ auto consumo.
- Ind 4: Componente pecuario/ mercado
- Ind 5: Componente agrícola/ auto consumo.
- Ind 6: Componente agrícola/ mercado.
- Ind 7: Disponibilidad de mano de obra.
- Ind 8: Manejo administrativo.
- Ind 9: Dependencia de insumos externos.
- Ind 10: Compromiso familiar.
- Ind 11: Estado de la vivienda.
- Ind 12: interacción comunitaria.

Los resultados obtenidos se encuentran consolidados en la Tabla 2.1.

TABLA 2.1: Análisis de antecedentes comerciales.

Fuente: Elaboración propia.

Software	Ind1	Ind2	Ind3	Ind4	Ind5	Ind6	Ind7	Ind8	Ind9	Ind10	Ind11	Ind12
Daruma	X	X							X			
Sismagro			X	X	X	X						
AgroWin			X	X	X	X		X	X			
Agroptim			X	X	X	X		X	X			
Synagro			X	X	X	X		X	X			
EGam	X	X										
Aranda							X	X				

Se observa que en el mercado existe la tendencia de software para gestión agropecuaria y administración, centrándose en el aspecto económico dejando a un lado los aspectos ambientales y sociales; respecto al caso de Aranda, un software que se anuncia como gestor de indicadores, se queda corto ante aspectos técnicos ya que este está desarrollado principalmente para medición de rendimiento de tecnologías informáticas.

2.3.2 Antecedentes académicos:

A nivel académico existen múltiples casos de estudio, diseño y desarrollo de metodologías enfocadas a múltiples áreas. Las métricas son usadas en todo tipo de proyectos y son una base confiable para la viabilidad y el desempeño de estos. Hasta el momento, no se ha encontrado una herramienta de software dedicada a una metodología específica, solo algunos dashboard para la gestión de indicadores.

A continuación se expone un ejemplo de investigación de metodología que hace uso de métricas para el mejoramiento en la calidad del software:

- **Visual Indicator Component Software to Show Component Design Quality andCharacteristic. D. Irwanto 2010:** Consiste en el desarrollo de una metodología con la que los desarrolladores de software podrán medir el buen diseño de software, su implementación y la calidad que tendrá su producto final y concientizarlos acerca de la importancia de estas métricas al momento de diseñar. Esta propuesta se hace debido a que se ha identificado que muchos desarrolladores no tienen en cuenta este proceso y conlleva a un mal diseño de software [27].

Actualmente, la Universidad Nacional de Colombia ofrece un doctorado en agroecología:

- El doctorado en Agroecología, es un programa de formación avanzada en investigación agroecológica, orientado a la generación y aplicación de conocimientos atendiendo a la complejidad del ambiente tropical, al rol histórico de las comunidades en el uso sostenible de la naturaleza y con enfoques metodológicos holísticos, que permiten procesos de investigación básica y aplicada en las áreas temáticas del programa académico.

En el objetivo general de este doctorado se menciona la incidencia en la tecnología para la agroecología y en los objetivos específicos se menciona "Desarrollar bases teóricas y metodológicas para promover y proponer tecnologías agroecológicas [28]

Capítulo 3

Desarrollo y descripción del proyecto

3.1 Descripción del sistema

La aplicación desarrollada está compuesta por un conjunto de clientes(web, móvil y escritorio), una API y una base de datos principal que cuenta con un ORM; estos dos últimos alojados en un servidor virtual (AWS y Cloud9) al que los clientes tienen acceso mediante protocolo Http; Adicional a esto, los clientes (a excepción del cliente web) poseen una base de datos temporal elaborada a partir de un sistema de gestión NoSQL [14] haciendo uso de PouchDB [36], que permite almacenar los nuevos registros y la edición de datos cuando los clientes se encuentren sin conexión a Internet. Véase la figura 3.1

3.2 Herramientas utilizadas

Para el desarrollo de este trabajo de grado, fue necesario el uso de aplicaciones y herramientas que facilitaran las tareas propuestas, aprovechando la amplia oferta de estas; generando un ambiente de trabajo cómodo, que permitiera el acceso a este desde cualquier parte y que brinde información del estado de las etapas de desarrollo de manera clara y concisa. A continuación se listan y describen cada una de las herramientas utilizadas:

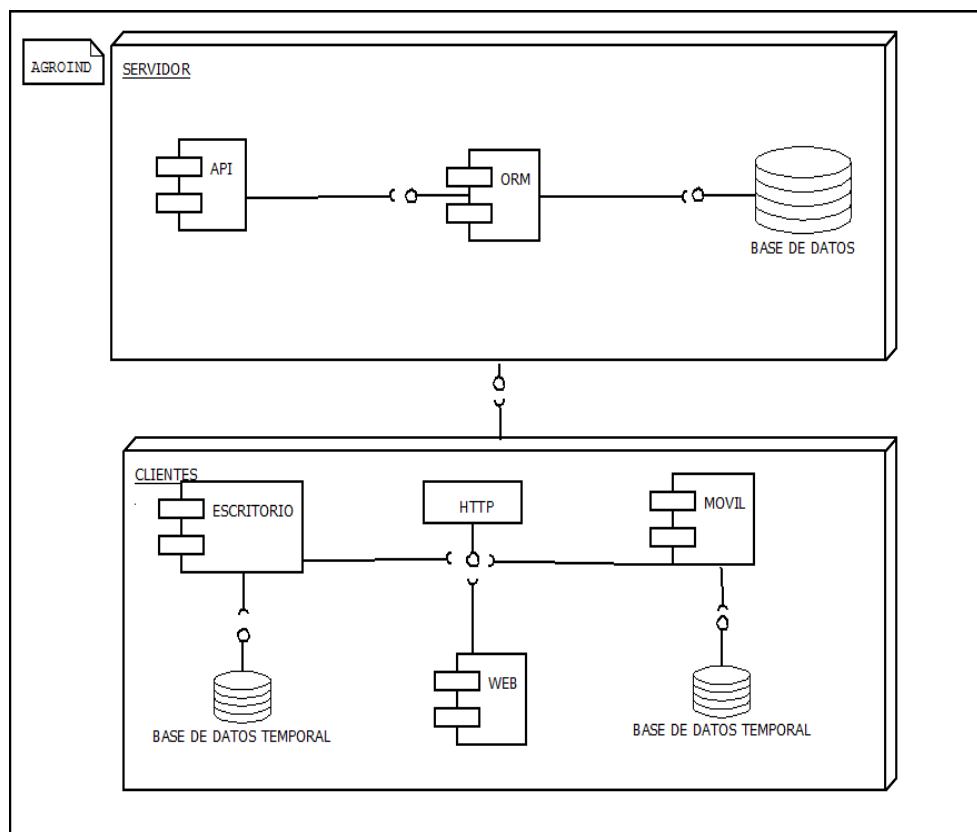


FIGURA 3.1: Diagrama de componentes general de la aplicación. Fuente: elaboración propia.

3.2.1 Herramientas de desarrollo de software:

- Ruby on Rails:** Ruby on Rails, también conocido como RoR o Rails, es un framework de aplicaciones web de código abierto escrito en el lenguaje de programación Ruby, siguiendo el paradigma de la arquitectura Modelo Vista Controlador (MVC). Sus dos principios fundamentales son Convention over Configuration, o convención sobre configuración que busca aumentar la productividad del programador al proporcionarle muchas herramientas que solo debe preocuparse por usar y no de implementarlas desde cero, esto sin perder flexibilidad, y reduciendo los archivos de configuración necesarios y DRY o don't repeat yourself, que implica que las definiciones solo deben hacerse una sola vez y reutilizar el código. Rails es distribuido por medio de RubyGems, que es el gestor de paquetes y librerías propio del lenguaje Ruby. Rails permite desarrollar una aplicación de forma muy rápida gracias a las herramientas que posee como Scaffolds, Migraciones y ActiveRecord (ORM integrado en rails).
 - Scaffold:** Un scaffold en Ruby on Rails es un conjunto completo de modelo, migración de base de datos, controlador, vistas y un conjunto de pruebas para los anteriores, todo esto generado automáticamente por el Framework [29].

- **Node.js:** es un entorno de ejecución para JavaScript construido con el motor de JavaScript V8 de Chrome. Node.js usa un modelo de operaciones E/S sin bloqueo y orientado a eventos, que lo hace liviano y eficiente.[30].
- **Electron:** Es un framework de desarrollo de aplicaciones nativas con elementos de javascript, CSS y HTML. [31].
- **Bower:** Un gestor de paquetes para la web, con este gestor se automatiza la instalación de paquetes, como librerías y gemas requeridas para la adecuación del entorno de desarrollo [32].
- **Adobe Phonegap:** Es una herramienta para e desarrollo de aplicaciones híbridas, permitiendo desarrollar aplicaciones multiplataforma con un código base a partir de Javascript, CSS y HTML [33].
- **Sistema Operativo basado en GNU/Linux:** se toma la decisión de usar sistemas operativos linux porque la configuración del entorno de desarrollo se facilitaba ya que muchas de las herramientas como Ruby on Rails y Electron funcionan mejor en este SO [29, 31]. Las versiones utilizadas son:
 - **Elementary OS** versión **Loki**.
 - **Ubuntu OS** versión **16.04**.
 - **Linux Mint** versión **18.1 Serena (Cinnamon edition)**.
- **Motor de base de datos:**
 - **SQLite:** SQLite es un motor de base de datos SQL incorporado. A diferencia de la mayoría de las otras bases de datos SQL, SQLite no tiene un proceso de servidor independiente. SQLite lee y escribe directamente en archivos de disco ordinarios[34].
 - **Postgres:** es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos comerciales [35].
 - **PouchDB:** es una base de datos de JavaScript de código abierto. PouchDB fue creado para ayudar a los desarrolladores crear aplicaciones web que funcionan sin conexión, así como lo hacen en línea [36].
Pouch DB es utilizado en la etapa de producción del software como gestor de datos.
- **Cloud9:** Es un servicio Web que ofrece un entorno para el desarrollo de software, con un sistema basado en GNU/Linux; este servicio permitió alojar la API y el motor de base de datos(Postgres) y funciononó a manera de servidor en la etapa de desarrollo de la aplicación.[37]

- **Amazon Web Services:** Es un servicio web que ofrece un entorno de producción y desarrollo para aplicaciones web y móviles. Actualmente AWS es utilizado por compañías y empresas innovadoras emergentes, gracias a sus servicios de seguridad empresarial y reducción de costos de TI; este servicio cuenta con una licencia de prueba por 12 meses, a partir del registro de la cuenta [38].
- **ChartJS:** Es una librería desarrollada en Javascript para agregar gráficos dinámicos a aplicaciones web. Esta librería presenta compatibilidad tanto para navegadores web de escritorio, como para navegadores en dispositivos móviles. Actualmente esta librería cuenta con una variedad de gráficas, entre ellas la opción de "Spiderweb", necesaria para la presentación de resultados en la aplicación como se observa en la figura 2.2. ChartJS cuenta con una licencia para uso sin ánimo de lucro[39].

3.2.2 Herramientas de control y gestión del proyecto:

- **Google Drive:** Servicio de almacenamiento en la nube. Drive te ofrece 15 GB de almacenamiento de Google gratis para empezar. Guarda fotos, artículos, diseños, dibujos, grabaciones, vídeos o lo que quieras. Esta aplicación se utilizó para almacenar todos los documentos relacionados con el proceso de desarrollo, las historias de usuario, anexos, Diagramas y demás [40].
- **Google Hangouts:** Aplicación utilizada para reuniones no presenciales en caso de ser necesaria [41].
- **Github:** es una forja (plataforma de desarrollo colaborativo) para alojar proyectos utilizando el sistema de control de versiones Git. Utiliza el framework Ruby on Rails por GitHub, Inc. (anteriormente conocida como Logical Awesome). Desde enero de 2010, GitHub opera bajo el nombre de GitHub, Inc. El código se almacena de forma pública, aunque también se puede hacer de forma privada, creando una cuenta de pago. En este caso se utilizó la licencia gratuita [42].
- **Ganttter:** es un servicio web gratuito con el que podremos realizar la planificación de nuestros proyectos y generar un diagrama de Gantt con las tareas, fases e hitos del mismo. Si bien estas funcionalidades, más o menos podemos encontrarlas en otras herramientas del mercado, Ganttter es interesante porque permite importar planificaciones procedentes de Microsoft Project que, en el ámbito profesional, está bastante extendido (aunque no todo el mundo puede asumir el coste de una licencia) por lo que esta herramienta puede encajar muy bien dentro de la dinámica de trabajo de un freelance [43].
- **Trello:** Es una herramienta de colaboración que organiza tus proyectos en tableros. Es decir, que gracias a Trello, podrás saber cuáles son las tareas que se llevan a cabo, quién trabaja en una tarea determinada y cuál es el estado de un proceso [44].

- **DIA:** Dia Portable es un editor de diagramas o módulos para muchos propósitos, Este cuenta con una serie de elementos que permiten desarrollar diagramas UML, de base de datos, circuitos, entre otros. [45]. .
- **Pencil project:** se construyó con el fin de proporcionar una herramienta de prototipado GUI gratuita y de código abierto que la gente puede instalar y utilizar para crear maquetas en plataformas de escritorio populares fácilmente [46].

3.3 Desarrollo del proceso de ingeniería de software

Teniendo en cuenta que la metodología de desarrollo elegida, Kanban,[12] la cual hace parte de las consideradas como metodologías ligeras, el proceso de desarrollo se dividió en "sprints", de estos se ejecutaron uno por cada etapa de desarrollo de los módulos propuestos en el cronograma; implementando las herramientas de gestión basadas en UML y Tableros para cada una de las etapas del proceso de ingeniería de software como:

- Historias de usuario
- Diagrama de base de datos y diagrama de clases.
- Diagrama de Componentes.
- Repositorios de Código fuente.
- plan de pruebas.
- Tablero de tareas.

3.3.1 Sprint 1:

En este sprint se abordó el modulo de usuarios y perfiles, generando la estructura base para el resto del desarrollo; siendo este modulo el encargado de la "población" en la aplicación. En esta primera iteración se trabajó en la definición de los tipos de usuario que poseería la aplicación, así como de las funciones a las que tendrían acceso en esta; todo acorde a las historias de usuario generadas en la reunión con el colaborador que en este caso desempeña la labor de cliente [figura 3.1].

TABLA 3.1: Historias de usuario para el Sprint1.

Product Backlog Sprint1	
Identificador de la HU	Enunciado de la Historia
NM-CLIENTE-0001	Como un usuario registrado. Necesito Ingresar al sistema Con la finalidad de autenticarme
NM-CLIENTE-0002	Como un Administrador Necesito crear usuarios Con la finalidad de crear usuarios y asignarles un perfil
NM-CLIENTE-0003	Como un usuario no registrado Necesito crear una cuenta Con la finalidad de ingresar al sistema
NM-CLIENTE-0004	Como un usuario registrado Necesito gestionar mi cuenta Con la finalidad de consultar o actualizar algunos datos
NM-CLIENTE-0005	Como un usuario registrado Necesito recuperar mi contraseña Con la finalidad de restablecer mi clave para acceso al sistema.
NM-CLIENTE-0006	Como un usuario registrado Necesito cambiar mi contraseña Con la finalidad de reemplazar mi clave de acceso a la aplicación.
NM-CLIENTE-0007	Como un usuario registrado Necesito cerrar sesión Con la finalidad de salir de mi sesión
NM-CLIENTE-0008	Como un administrador Necesito consultar usuarios Con la finalidad de buscar a mis usuarios
NM-CLIENTE-0009	Como un administrador Necesito editar o actualizar la información de un usuario Con la finalidad de mantener los datos de mis usuarios actualizados

Acorde a lo planificado, se abordó en el diseño de la aplicación a partir del modelamiento de la arquitectura MVC que se puede ver en la gráfica 3.11 y 3.12 en el que se encuentran los controladores y modelos referentes a este sprint(Users y Profiles) con sus respectivas funciones

y atributos, así como el modelamiento de datos a partir de los requerimientos establecidos en las historias de usuarios. Para este sprint se crearon las entidades de:

- **Usuarios:** Tabla principal de la base de datos en la que se gestiona la información básica del usuario así como su login y password para su interacción con el sistema
- **Perfiles:** Ya que no todos los usuarios ejecutarán todas las tareas que poseerá la aplicación, se requirió implementar una tabla en la que se configuraron los permisos y se generaron grupos de permisos para asignarlos a un usuario; en este caso se le denominaron perfiles.

La relación entre estas dos entidades consiste en una asignación en la que un usuario posee un perfil, pero dicho perfil puede ser asignado a muchos usuarios. Esto se puede evidenciar en el Diagrama 3.10

Respecto a el diseño de interfaz de usuario, se estableció que la interfaz base consta de una barra de título siempre presente en la vista y un menú principal cuyas opciones contiene submenús, siendo estos las funciones por cada uno de los módulos generados. Esta interfaz se mantiene en todos los clientes, con una leve variación en caso de que el tamaño de la pantalla sea reducido; en este caso, el menú será contenido en un botón en la parte superior izquierda de la aplicación, el cual despliega el menú. Este diseño permite aprovechar toda la zona derecha de la pantalla, evitando despliegues involuntarios de menús y permitiendo al usuario centrarse en la tarea que está realizando.

Siguiendo el procedimiento de la metodología, el proceso de desarrollo inició inmediatamente se terminó el análisis y diseño; en este sprint se centró la base de la arquitectura que seguirá siendo usada para el resto de las etapas de desarrollo. Se estableció entonces que la aplicación constaría de un cliente(cómo se estableció en los objetivos) y una API que será consumido por este. Actualmente la API corre en un servidor local, al igual que los clientes móviles y de escritorio.

Para las pruebas funcionales se estableció el formato de casos de prueba [Figura 3.3] con el fin de monitorear el estado de la aplicación y el cumplimiento de los criterios de aceptación consolidados en las historias de usuario de cada uno de los sprint. Este formato cuenta con un identificador, un nombre, la descripción de la funcionalidad a evaluar y una serie de pasos para ejecutar la prueba; finalmente, contiene una sección en la que se registra el resultado obtenido y si la prueba es aceptada o no. Cada uno de estos casos de prueba contiene los datos utilizados en su ejecución; estos se encuentran en la sección de anexos.

Con respecto a la ultima etapa de pruebas y retroalimentación, se encontraron errores menores, pero que afectaban principalmente la usabilidad de la aplicación. A partir de los resultados

obtenidos en las pruebas de este sprint, se resalta la necesidad de un feedback más amplio, con el propósito de generar un ambiente más amigable para el usuario, que lo mantenga al tanto de sus acciones dentro de la aplicación.

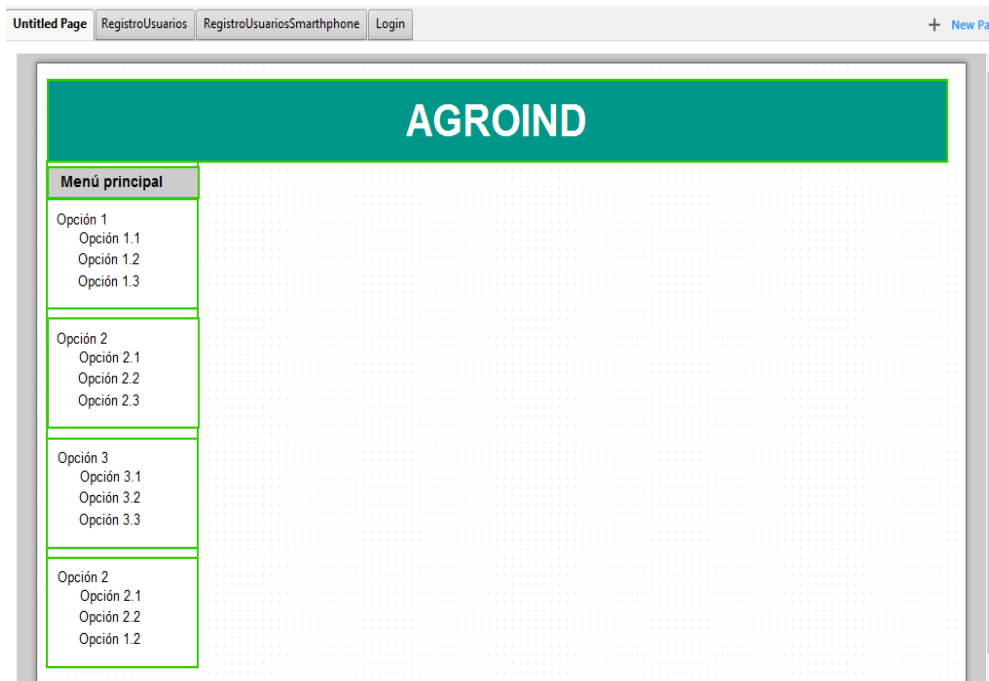


FIGURA 3.2: Interfaz base de la aplicación. Fuente: Elaboración propia, haciendo uso de [46]

Caso de prueba	REV:1	
	DOCUMENTO: CP_001	
Id caso de prueba	CP_001	
Nombre caso de prueba	CU1_ CREAR USARIOS	
Descripción	Permite a una persona cn permisos necesarios crear usuarios	
Pasos y condiciones de ejecución	* el sistema mostrara un formulario con los campos indicados en el anexo 1 . *El usuario ingresa los campos requeridos. *El sistema valida todos los campos ingresados *El sistema guarda los cambios y adiciona el nuevo usuario.	
Resultado esperado	Un nuevo usuario es registrado en la aplicación.	
Estado caso de prueba	Terminada	
Respuesta obtenida	Aprobado con algunas excepciones por corregir.	
Errores asociados	No notifica en caso de éxito o fracaso	
Responsable diseño	Oscar Eduardo Ramirez Arevalo	
Responsable ejecución	Oscar Eduardo Ramirez Arevalo	
Comentarios	El sistema no refresca los campos luego de crear un usuario	
	Aun no se cuentan con validadores de campos, esto causa que se pueda crear usuarios con identificaciones alfanúmericas y correos en mal formato	
	Al crear un usuario, la tabla de usuarios no se refresca.	

FIGURA 3.3: Formato de casos de prueba. Fuente: Elaboración propia.

3.3.2 Sprint 2:

En el segundo sprint se abordaron todos los aspectos referentes a la gestión de evaluaciones; satisfaciendo la configuración requerida para la ejecución de este proceso. De acuerdo con las especificaciones de Asoprovegas, para realizar una evaluación, se requiere identificar el predio(finca) del usuario a evaluar; de dicho predio requieren información básica como: área, altura sobre el nivel del mar (ASNM), ubicación geográfica y demás datos consolidados en los anexos de las historias de usuario de este sprint [Tabla 3.2]; también se identificaron los indicadores requeridos para el proceso de evaluación y su respectiva estructura, la cual consiste en un indicador que posee un valor total obtenido a partir de la sumatoria de valores parciales de las variables asociadas al indicador; con la condición de que el valor del indicador(valor total) sea siempre mayor o igual que 0 y menor o igual que 5, como se especifica en la metodología de sostenibilidad agroecológica [5].

TABLA 3.2: Historias de usuario para el Sprint2.

Fuente: Elaboración propia. Product Backlog Sprint2	
Identificador de la HU	Enunciado de la Historia
NM-CLIENTE-0001	Como un usuario con permisos de creación de usuarios y/o de evaluación Necesito Registrar una finca a un usuario propietario Con la finalidad de registrar el predio del propietario a evaluar
NM-CLIENTE-0002	Como un Usuario con permisos de evaluación Necesito Registrar la fecha de calificación en que se realizó alguna evaluación. Con la finalidad de llevar un registro de tiempo de evaluación.
NM-CLIENTE-0003	Como un Usuario con permisos de evaluación Necesito Registrar la calificación de la situación final de una finca Con la finalidad de almacenar los datos de evaluación en una de las dos situaciones a evaluar
NM-CLIENTE-0004	Como un Usuario con permisos de evaluación Necesito Registrar la calificación de la situación final de una finca Con la finalidad de almacenar los datos de evaluación en una de las dos situaciones a evaluar.
NM-CLIENTE-0005	Como un Usuario con permisos de evaluación Necesito editar los valores de la evaluación en cualquiera de las dos situaciones a registrar Con la finalidad de corregir posibles errores cometidos en el proceso de registro.
NM-CLIENTE-0006	Como un administrador Necesito cambiar los enunciados de los formularios de evaluación. Con la finalidad de poder editar los enunciados de los indicadores en evaluación

Considerando que los indicadores presentados para la evaluación son susceptibles a cambios; tanto en los enunciados de las variables, en el valor parcial de estas, así como en el nombre del

indicador; se optó por realizar una sección en el modulo que permitiera estos cambios a futuro, de forma tal que los indicadores fueran completamente parametrizables.

Las entidades y atributos que se agregaron al modelo de base de datos (figura 3.10) son las siguientes:

- **Propietario (Users id):** Este atributo de la tabla "Land" es utilizado para identificar los usuarios que poseen predios.
- **Finca (Land):** es una entidad con la que se registran datos básicos de un predio, expuestos en el anexo 1 de el documento de historias de usuario.
- **Indicador:** Esta entidad sirve para registrar un indicador para su posterior evaluación.
- **Variable:** En esta entidad se registran las variables que son asociadas con un indicador específico.
- **Evaluación:** Esta entidad coordina la solicitud de evaluaciones generada por un perfil administrador

Por parte de Asoprovegas, se recibió el formulario con el que se realiza la evaluación de forma manual [Figura 3.4]; a partir de esta estructura, además de modelar las entidades en la base de datos necesarias para los indicadores propuestos, se realizó un diseño de interfaz de usuario que permite una fácil adaptación a la aplicación, siendo esta lo más similar posible al formulario que se ha utilizado en previos procesos de evaluación.

	A	B	C	D
1	CALIFICACIÓN DE LAS VARIABLES DE LOS INDICADORES			
2	1. Manejo de suelos en la finca			
3	Variable	Calificación optima	Calificación obtenida inicial	Calificación obtenida final
4	Presencia de coberturas vivas y mulch (establecidas o espontaneas) en toda el área cultivada	1,2	1	
5	Labranza mínima o de conservación en las áreas de cultivos (Solo el sitio sembrado, curvas a nivel, terrazas etc)	1,2	0,5	
6	Barreras vivas y zanjas de infiltración	0,8	0,3	
7	Rotación de cultivos o rotación del pastoreo	0,8	0	
8	Estructura granular y color pardo oscuro	0,5	0	
9	Abundancia de vida visible	0,5	0	
10	Puntaje del indicador	5,0	1,8	0,0
11				

FIGURA 3.4: Formulario manual para evaluación de indicadores. Fuente: Tomado de anexos.

Criterio	valor
texto1	Combo Box
texto2	Combo Box
texto3	Combo Box
texto4	Combo Box
texto5	Combo Box
texto6	Combo Box

FIGURA 3.5: Interfaz de usuario para evaluación de indicadores. Fuente: Elaboración propia, utilizando [46]

La interfaz de usuario desarrollada [Figura 3.5], cuenta con los mismos componentes que el formulario proporcionado por Asoprovegas [Figura 3.4], con un ligero cambio en la asignación del valor obtenido para las variables, para este se tomó la decisión de agregar un combo box por cada variable, que contenga las opciones de valor máximo y valor mínimo de esta, sin retirar por completo la introducción manual de datos; para este ultimo caso el sistema verifica que el valor ingresado se encuentre entre los valores permitidos.

La interfaz en el cliente móvil no presenta mayores alteraciones, exceptuando el hecho de que el menú principal se oculta debido a las dimensiones de la pantalla del dispositivo como se ha mencionado anteriormente [Página 27].

Criterio	valor
texto1	Combo
texto2	Combo
texto3	Combo
texto4	Combo
texto5	Combo
texto6	Combo

FIGURA 3.6: Interfaz de usuario para Evaluación de indicadores, versión móvil. Fuente: Elaboración propia, utilizando [46]

Con respecto al ambiente de desarrollo de este sprint, se migró la base de datos de Sqlite a PostgreSQL[35] y junto a la API se instalaron en un servidor virtual temporal proporcionado Cloud9 [37], dando acceso a los clientes móviles y de escritorio a una base de datos centralizada. Esta medida se tomó para iniciar el proceso de implementación de las funciones con desconexión.

3.3.2.1 Funciones con desconexión:

Actualmente Colombia solo cuenta con una cobertura de Internet del 31% a nivel nacional y de un 13.9% en el departamento del Valle del Cauca, según el reporte del ultimo trimestre del año 2016 ofrecido por el Ministerio de las TIC [47]. En las gráficas 3.7 y 3.8 se evidencia que El ritmo de crecimiento de la cobertura es lento, con una variación nacional de apenas el 1.5% con respecto al periodo anterior (segundo trimestre del 2016) y una variación departamental de apenas 0.30% durante el mismo periodo.

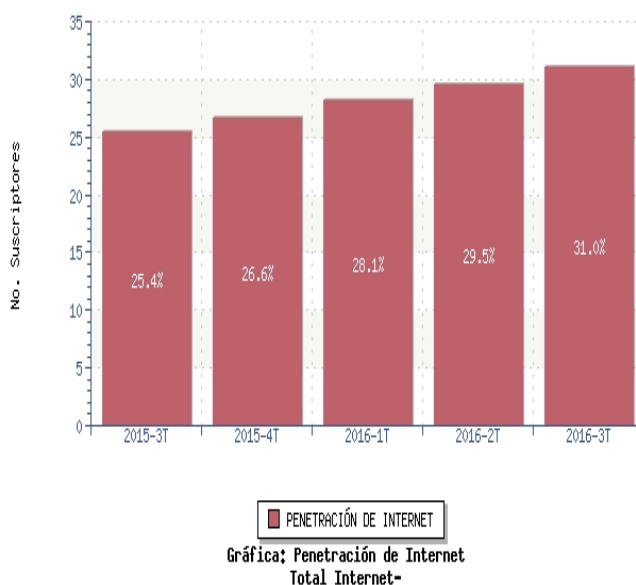


FIGURA 3.7: gráfica de penetración de internet a nivel nacional. Fuente: tomado [47]

Como se mencionó en la formulación del problema [Subsección 1.3.2], uno de los principales factores a tener en cuenta para el desarrollo de la sistematización del proceso de evaluación es la poca cobertura del servicio de Internet en la zona rural de Tuluá. Esta situación ha sido la principal razón para continuar realizando este proceso de forma manual y no considerar una opción viable como lo es un sistema informático.

Teniendo en cuenta este factor negativo para el uso de la aplicación en su campo de acción, en la aplicación se implementó una base de datos con Pouch DB [36]. El uso de esta base de datos

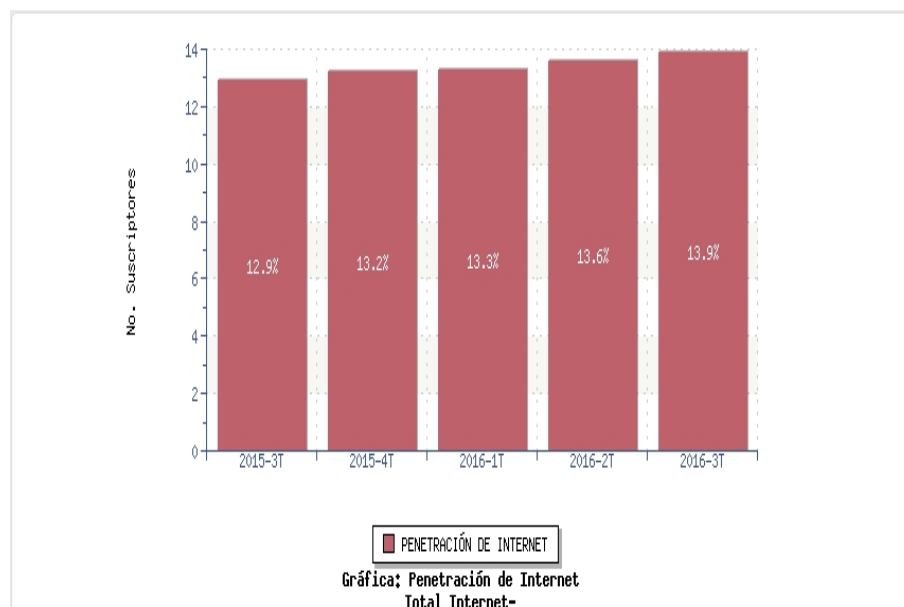


FIGURA 3.8: gráfica de penetración de internet en el Valle del Cauca. Fuente: tomado [47]

cubre el componente de "base de datos temporal" [Figura 3.1] propuesto en la arquitectura del sistema; esta se encontrará alojada en el cliente y mantendrá un almacenamiento local de manera temporal, permitiendo que los evaluadores y/o encargados de recolectar datos en el campo puedan generar datos desde los predios que no cuenten con acceso a la red y sincronizando los registros obtenidos con la base de datos principal cuando el cliente tenga nuevamente conexión. Ya que PouchDB es tecnología Javascript es compatible con cualquier tipo de cliente, gracias al uso de Electron [31] que permite el uso de Javascript en las aplicaciones nativas generadas en este.

El proceso que ejecuta la aplicación para las funciones con desconexión es explicado en la figura 3.9.

Cabe mencionar que no todas las funciones de la aplicación están disponibles cuando el dispositivo se encuentre sin conexión a Internet, principalmente las funciones de edición y algunas funciones de registro; esta decisión se tomó debido a que se presentarían ambigüedades en los datos parametrizados y en algunos registros que deben estar presentes en la base de datos principal antes que en la base de datos local, como por ejemplo un perfil, un usuario o el estado de sesión. En la sección de anexos se encontrará con la planificación realizada para las funciones con desconexión así como advertencias en el manual de usuario acerca de estas.

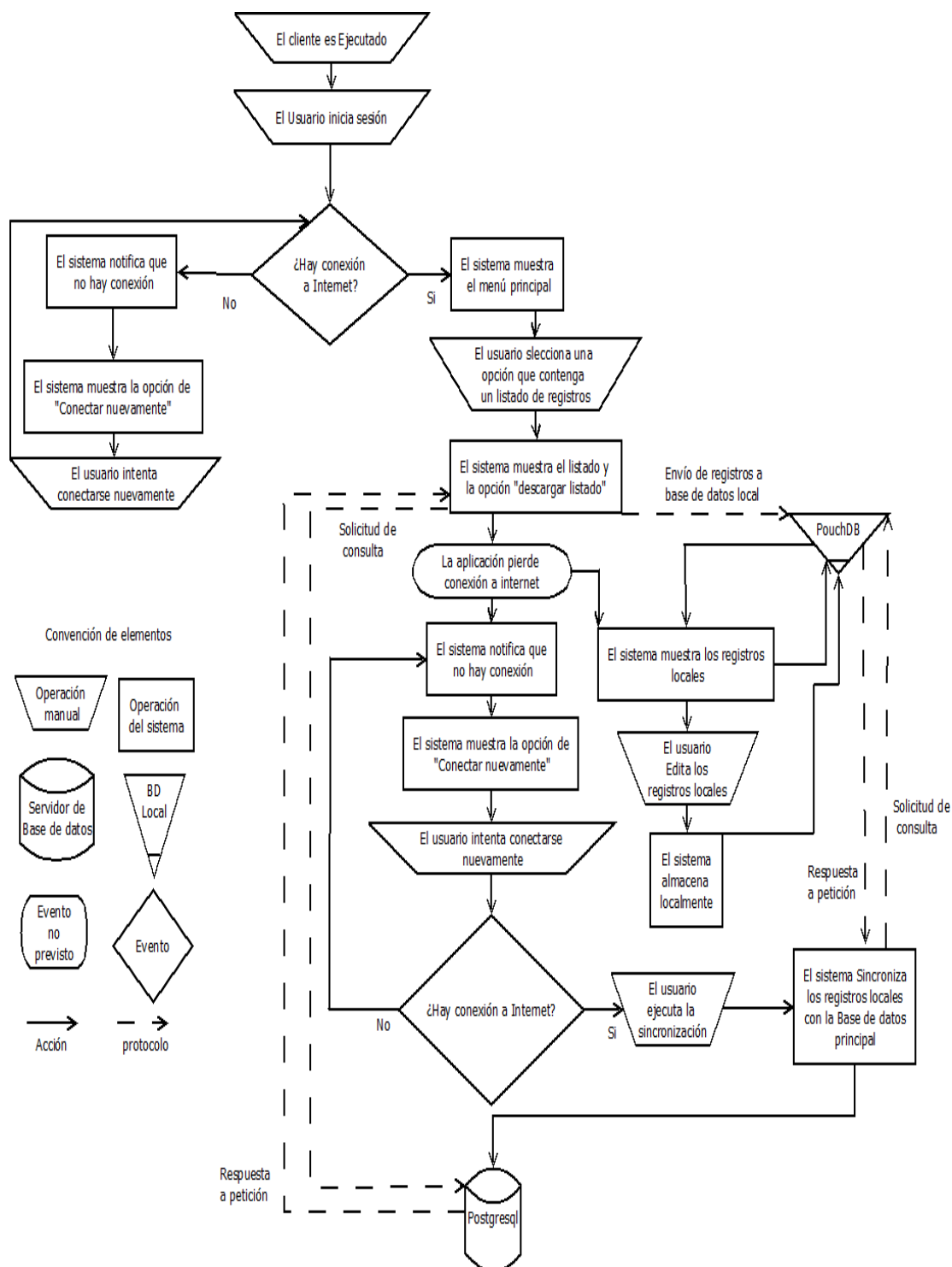


FIGURA 3.9: Diagrama de flujo de funciones con desconexión. Fuente: Elaboración propia, usando [45]

3.3.3 Sprint 3:

En este sprint se desarrolló la presentación de calificación de indicadores (propósito principal de toda la aplicación) tomando como base las tablas de evaluación desarrolladas en el sprint anterior y los resultados obtenidos en estas. Para esta sección se toma en cuenta "el estado de alerta" para la calificación, el cual será explicado más adelante en la sección 3.3.3.1 de este documento.

TABLA 3.3: Historias de usuario para el Sprint3.

Fuente: Elaboración propia. Product Backlog Sprint3	
Identificador de la HU	Enunciado de la Historia
NM-CLIENTE-0001	Como un usuario con permisos para generar calificaciones Necesito visualizar el resultado obtenido en la evaluación de indicadores de algún propietario Con la finalidad de generar una lista de resultados que presentar al propietario evaluado
NM-CLIENTE-0002	Como un usuario con permisos para generar calificaciones Necesito visualizar una gráfica de radar que ilustre los resultados obtenidos en cada uno de los indicadores Con la finalidad de presentar la información de manera más amigable al propietario.
NM-CLIENTE-0003	Como un Usuario con permisos de evaluación Necesito configurar un código de colores de estado del indicador Con la finalidad de que el propietario comprenda mejor el estado de sus resultados, así como identificar en que indicadores se requieren más esfuerzos por mejorar
NM-CLIENTE-0004	Como un usuario con permisos de evaluador Necesito agregar análisis y recomendaciones posterior a la calificación del propietario Con el fin de presentar al propietario mi interpretación de sus resultados bajo mi criterio profesional y así poder generar acciones de mejora a futuro
NM-CLIENTE-0005	Como un usuario con permisos de evaluador Necesito el resultado de la calificación en formato PDF Con el fin de generar un reporte que pueda entregar al evaluado en forma física

La calificación constará de 6 partes:

1. **Cuadro de presentación:** Sección para mostrar la información personal e información del predio evaluado.
2. **Lista de indicadores evaluados:** Se presenta una lista con todos los indicadores evaluados, sus respectivas variables, el resultado parcial de cada una y el resultado obtenido por cada indicador; este resultado se obtiene de la siguiente manera:

$$\sum_{i=1}^n i = \text{Resultado total del indicador.}$$

Donde n es el número de variable por indicador e i es el resultado obtenido para la variable.

Dicho total mostrará el estado de alerta en que se encuentra el indicador.

3. **Tabla de Resumen:** Muestra una tabla con el título de todos los indicadores evaluados y el resultado total para cada uno, así como el estado de alerta para cada indicador.
4. **cuadro explicativo de estado de alerta:** Consiste en un cuadro donde se muestra la información relacionada con cada estado de alerta.
5. **Gráfica de radar:** Como se muestra en la (figura2.2), se realiza la respectiva gráfica tomando los resultados totales obtenidos para cada indicador.
6. **Sección de análisis y recomendaciones o conclusiones:** Consta de dos espacios para escritura, donde el evaluador agrega el análisis realizado sobre los resultados obtenidos en la evaluación y posteriormente agrega una serie de recomendaciones al evaluado.

A la base de datos se le agregó la siguiente entidad y se modificó la tabla de evaluación para llevar a cabo la implementación de este sprint:

- **Alerta:** Entidad encargada de registrar los parámetros para el estado de alerta.
- **Conclusiones:** Consiste en dos campos para el análisis y las recomendaciones del evaluador, agregados a la tabla de evaluaciones..

A partir de este punto en la etapa de desarrollo, la base de datos no presenta adiciones ni cambios; el siguiente sprint se basa en consultas y reportes. El diagrama de base de datos resultante se encuentra en la figura 3.10.

En este Sprint se empezó a hacer uso de ChartJS [39] para la elaboración de la gráfica de radar necesaria para el reporte individual de calificación. Es en este sprint donde termina el desarrollo de la base de datos; adicionalmente, se inicia la ejecución de la API en el entorno de producción de AWS [38], con el fin de iniciar un seguimiento a su comportamiento. Aprovechando el desarrollo en HTML del Frontend, para la función de exportación a PDF se utilizó una librería que toma el contenido de un componente "div" en la vista, lo procesa como imagen y permite guardar el archivo en formato PDF en cualquier directorio de destino.

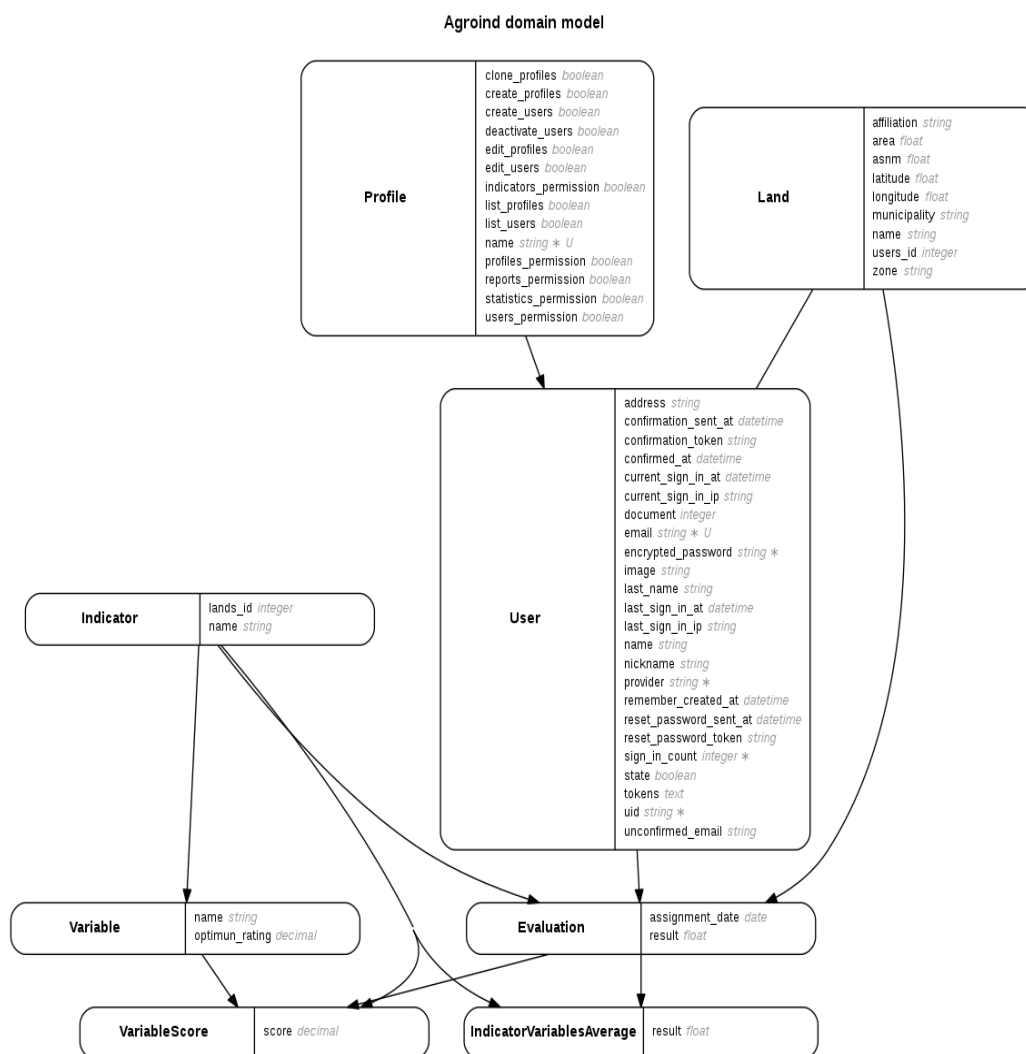


FIGURA 3.10: Diagrama de base de datos. Fuente: elaborado con Rail-ERD.

3.3.3.1 Estado de alerta:

Consiste en un "mecanismo de ayuda" para el entendimiento de los resultados obtenidos a través de un "código de colores" que representan el estado en el que se encuentra este. Dicho estado de alerta consta de las siguientes partes:

- **Estado:** Breve descripción de como se encuentra el indicador.
- **Rango de puntuación:** Rango numérico asignado al estado de alerta.
- **Código de colores:** Color asociado con el estado de alerta en que se encuentra el indicador.

- **Alerta de atención:** Describe la gravedad del resultado.
- **Plazo de atención:** Sugiere una prioridad de atención ante el resultado.

3.3.4 Sprint 4:

Una de las ventajas de la sistematización es la posibilidad de analizar grandes cantidades de datos de manera fácil y rápida (pág 12 [7]). Como se menciona en la Formulación del problema, el factor de gestión de la información es el punto clave de todo el proceso desarrollado hasta este punto; es por esto que este sprint se centró en el desarrollo de un módulo que permitiera obtener información derivada de los resultados individuales obtenidos en el sprint anterior. El módulo desarrollado en este sprint consta de reportes estadísticos generados a partir de los registros obtenidos de todas las evaluaciones generadas en la aplicación. Dicho proceso hecho manualmente implicaría una cantidad considerable de tiempo y posibles "errores humanos". Los reportes estadísticos que se planearon con Asoprovegas son los siguientes:

- **Promedio general de todos los indicadores:** Consiste en generar un valor promedio para cada uno de los indicadores, calculándolo a partir del resultado obtenido en todas las evaluaciones de fincas.
- **Ranking de mejores calificaciones:** Consiste en presentar un listado de los propietarios que hayan obtenido los mejores resultados en los indicadores.
- **Ranking de peores calificaciones:** Consiste en presentar un listado de los propietarios que hayan obtenido los peores resultados en los indicadores. calificaciones.

estos datos estadísticos se calculan de la siguiente manera:

$$\frac{\sum_{n=1}^n i}{n}$$

Para ambos casos, i es el resultado del indicador, pero n , en el caso de promedio general, representa el número total de fincas evaluadas y en ambos ranking representa el número total de indicadores evaluados

Estos requerimientos se pueden evidenciar en las historias de usuario resultantes para este sprint (Tabla 3.4).

En esta etapa del proceso de ingeniería de software, el diseño de la arquitectura de la aplicación para su posterior implementación ha culminado procurando conservar la arquitectura MVC como lo evidencia el diagrama 3.11 y 3.12 que conforman el diagrama de clases final; ya que las vistas son generadas a partir de HTML y JS no son consideradas como clases.

Respecto a la etapa de desarrollo, teniendo en cuenta que los reportes solicitados comparten casi la misma estructura del reporte individual de calificaciones, generado en el sprint anterior,

TABLA 3.4: Historias de usuario para el Sprint4.

Fuente: Elaboración propia.

Product Backlog Sprint4	
Identificador de la HU	Enunciado de la Historia
NM-CLIENTE-0001	Como un usuario con permisos para reportes estadísticos Necesito obtener el promedio del valor que los indicadores han obtenido, respecto a todas las fincas evaluadas, así como poder seleccionar el componente(subconjunto de indicadores) del cual deseo su promedio Con la finalidad de monitorear el estado general de los indicadores
NM-CLIENTE-0002	Como un usuario con permisos para reportes estadísticos Necesito que la información obtenida en el reporte de "promedio general de indicadores" sea presentada en una tabla de resumen y una gráfica de red equivalente a los datos; así como utilizar el código de colores para el estado de alerta en el resultado obtenido para cada promedio . Con el propósito de observar la información de manera más organizada
NM-CLIENTE-0003	Como un usuario con permisos para reportes estadísticos Necesito generar el ranking de los mejores y peores calificados, según el valor obtenido al promediar el resultado de todos los indicadores de su evaluación individual para poder enfocar esfuerzos en los usuarios evaluados que más lo requieran, así como identificar el progreso obtenido en las evaluaciones
NM-CLIENTE-0004	Como un usuario con permisos de evaluador Necesito el resultado de los reportes estadísticos en formato PDF Con el fin de generar un reporte que pueda almacenar en forma física

se realizaron modificaciones en las consultas que generan el reporte y se reutilizó toda la implementación del estado de alerta y la generación de la gráfica de red. En este punto, el ambiente en el que se ejecuta la API se encuentra en AWS.

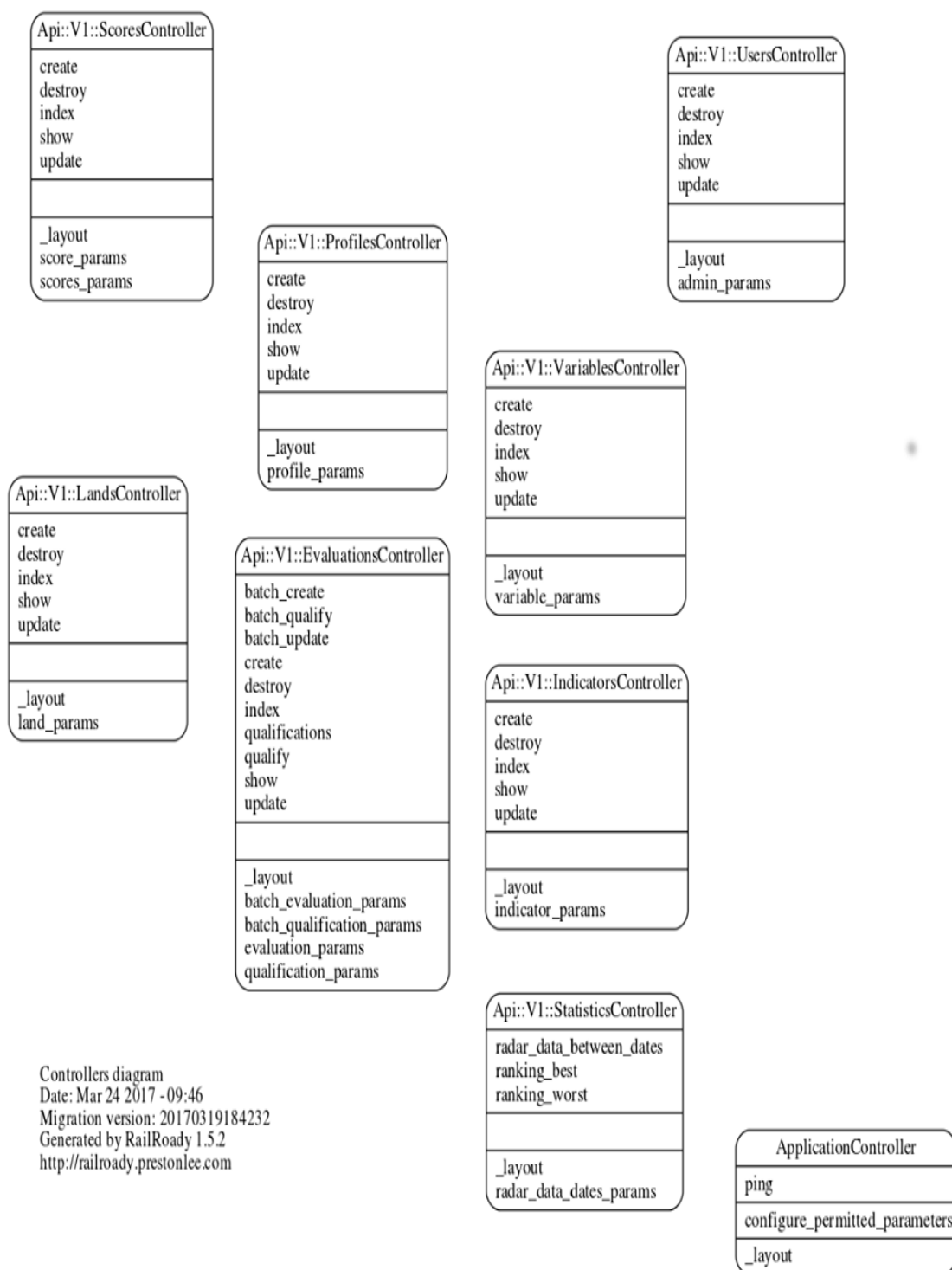


FIGURA 3.11: Diagrama de clases, controladores. Fuente: elaborado con RailRoady.

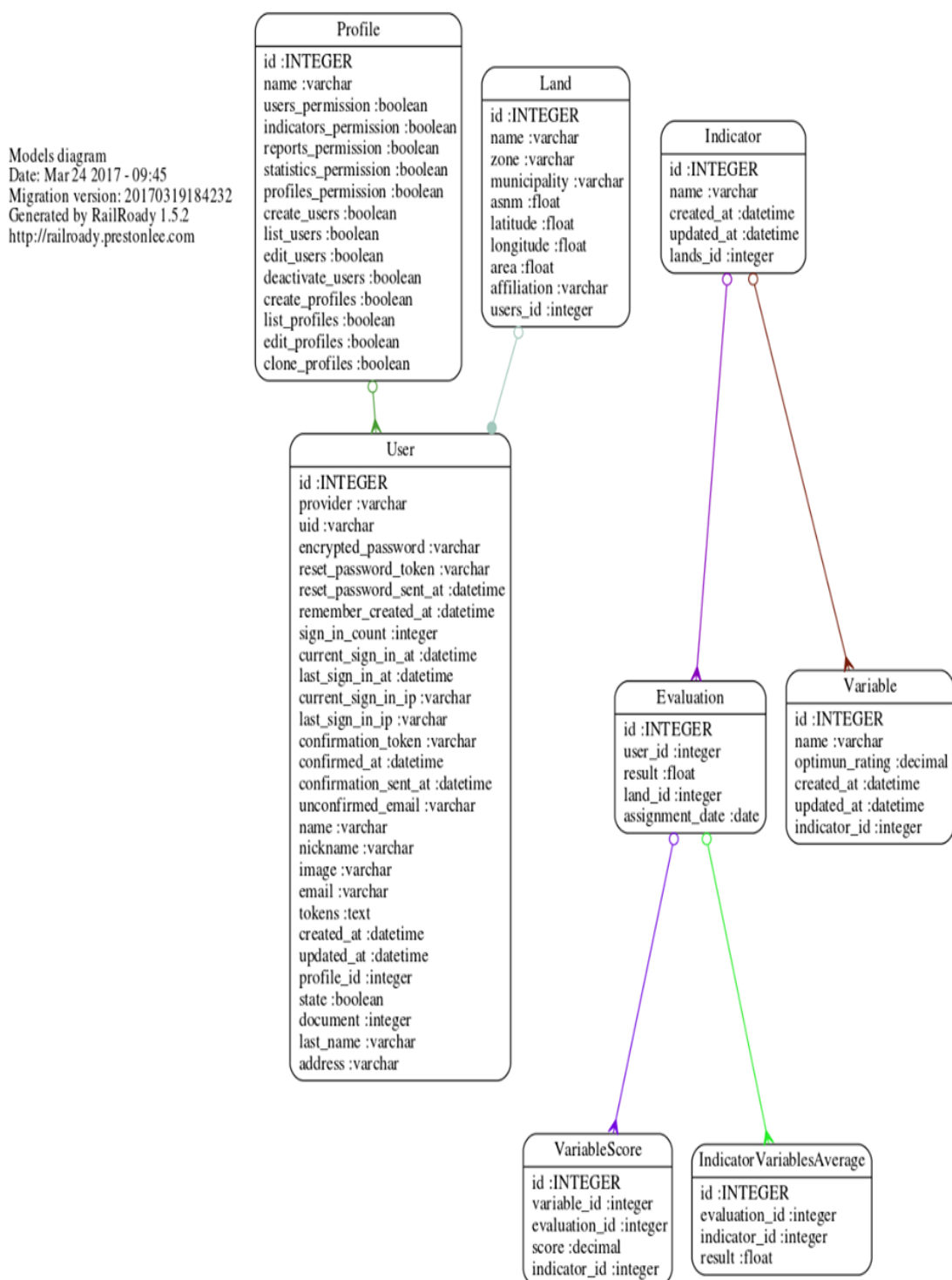


FIGURA 3.12: Diagrama de clases, modelos. Fuente: elaborado con RailRoady.

Capítulo 4

Conclusiones y trabajos futuros.

4.1 Conclusiones.

1. Se encontró una solución viable a la necesidad de Asoprovegas para la de gestión de evaluación de indicadores agroecológicos y la información resultante de este proceso, mediante la implementación de un sistema de información consolidado en una aplicación que cuenta con las especificaciones funcionales propuestas por la asociación campesina para la ejecución de las actividades necesarias para llevar a cabo un proceso de evaluación.
2. Con la tecnología actual es posible desarrollar aplicaciones que se adapten a las necesidades del campo, superando obstáculos infraestructurales como la poca cobertura de Internet a nivel nacional.
3. Al generarse un sistema de información a partir de una disciplina como la metodología de sostenibilidad agroecológica, se generaron nuevas formas de mejoramiento para este tipo de procesos; un ejemplo de esto es la posibilidad de realizar la evaluación de indicadores a un considerable número de predios, lo que permite ampliar el territorio de aplicación de la metodología. Esto también permite generar un volumen de datos con los que se obtienen resultados estadísticos más confiables.
4. Kanban ofreció una plataforma laboral que permitió conocer el estado del proceso de ingeniería en todo momento, esto gracias a que el diseño del Kanban consistió en una sincronización del tablero de actividades con el cronograma establecido en la planificación del proyecto. Adicional a esto, Kanban fue una herramienta útil para el monitoreo de problemas presentados en la etapa de desarrollo, coordinando tableros con las listas de problemas obtenidos después de cada etapa de pruebas, generando así una realimentación más ágil en el proceso de implementación.

5. La elaboración de los clientes a partir de frameworks de Javascript como Angular, el uso de html y css, y el aprovechamiento de la arquitectura REST redujo considerablemente las labores de implementación, ya que estos, en conjunto con Electron y Phonegap, permitieron generar los clientes para los diversos SO, reutilizando gran parte del código fuente, principalmente entre el cliente de escritorio y el cliente móvil; todo esto sin mayor complicación de compatibilidad de las tecnologías reduciendo los conflictos a cuestiones de configuración.

4.2 Trabajos futuros.

Teniendo en cuenta que el modulo de estadística es un conjunto de funciones que permiten un análisis global de los datos obtenidos en una determinada región; y que este fue un modulo que se adicionó con la implementación de AGROIND al proceso de la Metodología de sostenibilidad agroecológica; queda abierta la posibilidad de implementar nuevas funciones estadísticas que permitan un mejor análisis de sostenibilidad de carácter regional; esto a medida de que el análisis de datos estadísticos actuales (ranking y promedio general) generen mejores bases para acciones de mejora agroecológica dando así un valor agregado a la metodología.

Aunque en este proceso de ingeniería de software se desarrollaron las funciones offline que permitieran el uso de la aplicación para la recolección de datos directamente en el campo en situaciones adversas con respecto a la cobertura del servicio Internet en la zona rural de Tuluá, con la estructura implementada en este proceso de desarrollo de software se tiene la posibilidad de generar una actualización del sistema que cuente con nuevas funciones offline que cubran otros aspectos no considerados en la planificación de actividades sin conexión, como la parametrización de usuarios y demás. Esto implicaría extender el diseño arquitectónico y de datos con el fin de posibilitar dichas mejoras.

Como AGROIND es una aplicación relacionada con el campo, sería importante generar espacios de socialización con diferentes asociaciones y carreras afines, con el propósito de expandir su utilización y beneficiar a más personas. En este proceso se podría monitorear el rendimiento de la aplicación a gran escala y generar una retroalimentación más completa.

Anexos.

Toda la documentación del proceso de ingeniería de software, así como los respectivos manuales de usuario e instaladores de la aplicación resultante de este trabajo de grado se encuentran en <https://goo.gl/D0q68I>. Organizados de la siguiente manera:

- Ingeniería
 - Sprint1.
 - Sprint2.
 - Sprint3.
 - Sprint4.
 - Planificación desconexión.
 - Cronograma.
- Manuales de usuario.
- Releases.
 - mobile
 - Desktop

Cada carpeta de sprints contiene las historias de usuario, los documentos generados en la etapa de diseño, y los casos de prueba ejecutados.

En la carpeta de manuales encontrará un manual de usuario correspondiente a cada uno de los módulos implementados, tanto para la versión móvil como la versión de escritorio de la aplicación así como un manual de instalación.

La carpeta Releases contiene los instaladores de la aplicación en cada una de sus plataformas (Android, Linux y Windows).

El repositorio del código fuente de la aplicación se encuentra en el siguiente enlace:

- API: <https://github.com/jsvelasquezv/agroind-api>
- Cliente de escritorio: <https://github.com/jsvelasquezv/agroind-desktop>
- Cliente móvil: <https://github.com/jsvelasquezv/agroind-mobile>
- Cliente web: <https://github.com/jsvelasquezv/agroind-web>

Se accede al cliente web con la siguiente URL :

<http://agroind-web.s3-website-sa-east-1.amazonaws.com/app/#/>

Referencias

- [1] Asoprovegas 2da versión *Propuesta metodológica para el análisis de resultados e impactos en el proceso de reconversión agroecológica.*
- [2] Diccionario de la real Academia (DRAE). *Definición de metodología* (2.).
<http://dle.rae.es/?id=P7eTCPD>
- [3] Diccionario de la real Academia (DRAE). *Definición de método* (2,3 y 4.).
<http://dle.rae.es/?id=P7dyaFK>
- [4] Altieri M. y Toledo. V. 2011 *La revolución agroecológica de América Latina: rescatar la naturaleza, asegurar la soberanía alimentaria y empoderar al campesino. Publicado por: Sociedad Científica Latinoamericana de Agroecología. SOCLA. Páginas 34.*
- [5] A. Acevedo Osorio *Un procedimiento metodológico para diseñar, monitorear y evaluar programas rurales con enfoque de desarrollo sostenible.*
<http://goo.gl/1EtHzB>
- [6] Diccionario de la real Academia (DRAE). *Definición de sistema* (2.).
<http://dle.rae.es/?id=Y2AFX5s>
- [7] M. G. Piattini, J. A Calvo-Manzano *Análisis y diseño de aplicaciones Informáticas de Gestión .*
- [8] Diccionario de la real Academia (DRAE). *Definición de Asociación* (2).
<http://dle.rae.es/?id=424gSON>
- [9] Presidencia de la República. *Decreto 2716 de 1994 Nivel Nacional*
- [10] Y.SUGIMORI, K. KUSUNOKI, F. CHO and S. UCHIKAWA 1977. *Toyota production system and Kanban system Materialization of just-in-time and respect-forhuman system*
<https://goo.gl/UP65fY> Visitado por ultima vez 15/03/2017
- [11] Gross, J. M., McInnis, K. R. 2003 *Kanban Made Simple : Demystifying and Applying Toyota's Legendary Manufacturing Process*

-
- [12] Ahmad M. O., Markkula J. and Ovio M. 2013. *Kanban in software development: A systematic literature review*
 - [13] Codd, E.F. (1970). *A Relational Model of Data for Large Shared Data Banks*.
 - [14] A. Moniruzzaman, S Hossain *NoSQL Database: New Era of Databases for Big data Analytics Classification, Characteristics and Comparison*
<https://goo.gl/CXQxc0> Vistado por ultima vez 25/03/2017
 - [15] Oracle *A Relational Database Overview*.
<https://docs.oracle.com/javase/tutorial/jdbc/overview/database.html>
 visitado por última vez 09/02/17.
 - [16] Rj45, 8 Apr 2008. *Simple Example of MVC (Model View Controller) Design Pattern for Abstraction*.
<http://goo.gl/RRdxRf> visitado por última vez 09/02/17.
 - [17] Microsoft. *Model-View-Controller*.
<https://msdn.microsoft.com/en-us/library/ff649643.aspx> visitado por última vez 09/02/17.
 - [18] Safelayer *Integración de servicios de seguridad Servicios web de seguridad al estilo REST*.
<https://goo.gl/Vo4xaN> visitado por última vez 09/02/17.
 - [19] Oracle *What Are RESTful Web Services?*.
<http://docs.oracle.com/javaee/6/tutorial/doc/gijqy.html> visitado por última vez 09/02/17.
 - [20] Daruma *Indicadores de calidad ambiental imprescindibles*.
<http://goo.gl/umqUpD> visitado por última vez 09/02/17.
 - [21] Sismagro *Software agropecuario y agrícola* .
<http://goo.gl/kZTW0x> visitado por última vez 09/02/17.
 - [22] AgroWin *¿Qué es AgroWin?*.
<http://www.agrowin.com> visitado por última vez 09/02/17.
 - [23] Agroptima *Software y Aplicación de gestión agrícola fácil.*.
<https://www.agroptima.com> visitado por última vez 09/02/17.
 - [24] Synagro *Evolución de control de gestion*.
<http://www.synagro.com.ar> visitado por última vez 09/02/17.
 - [25] Egam *ISO 14001 Gestión del Medio Ambiente*.
<http://www.egambpm.com/iso-14001> visitado por última vez 09/02/17.

-
- [26] Aranda *GENERE TODO TIPO DE INDICADORES PARA LA GESTIÓN DE SUS ACTIVOS, SERVICIOS Y PARA DIFERENTES PROCESOS IT DE SU ORGANIZACIÓN.*
<http://arandasoft.com/aranda-dashboard> visitado por última vez 09/02/17.
 - [27] D. Irwanto 2010. *Visual Indicator Component Software to Show Component Design Quality and Characteristic.*
<http://goo.gl/GNtKpU> visitado por última vez 09/02/17.
 - [28] Universidad Nacional de Colombia. *Doctorado en agroecología SNI ESS 55123*
<https://goo.gl/QyU2ho>
 - [29] Ruby on Rails *The Rails Command Line.*
<http://goo.gl/4PNekW> visitado por última vez 09/02/17.
 - [30] NodeJs *Presentación del entorno.*
<https://nodejs.org/es> visitado por última vez 10/02/17.
 - [31] ElectronJS *Build cross platform desktop apps with JavaScript, HTML, and CSS.*
<http://electron.atom.io> visitado por última vez 10/02/17.
 - [32] Bower *A package manager for the web.*
<https://bower.io> visitado por última vez 10/02/17.
 - [33] Phonegap *Build amazing mobile apps powered by open web tech.*
<http://phonegap.com/> visitado por última vez 10/02/17.
 - [34] SQLite *Acerca de SQLite.*
<https://www.sqlite.org/about.html> visitado por última vez 10/02/17.
 - [35] PostgreSQL. *Sobre PostgreSQL.*
<http://www.postgresql.org> visitado por última vez 10/02/17.
 - [36] Pouch DB. *The Database that Syncs!.*
<https://pouchdb.com> visitado por última vez 10/02/17.
 - [37] Cloud 9. *Powerful Workspaces.*
<https://c9.io> visitado por última vez 10/02/17.
 - [38] Amazon. *Amazon Web Services. Capa Gratuita de AWS*
<https://aws.amazon.com/es/free> visitado por última vez 20/03/17.
 - [39] ChartJS. *Simple yet flexible JavaScript charting for designers and developers.*
<http://www.chartjs.org/> visitado por última vez 20/03/17.
 - [40] Google Drive . *Todos tus archivos estés donde estés.*
<https://goo.gl/IKMS0Z>. visitado por última vez 10/02/17.

- [41] Google Drive . *Todos tus archivos estés donde estés*.
<https://goo.gl/S0Kfw1>. visitado por última vez 10/02/17.
- [42] Github *Online project hosting using Git*.
<https://github.com>. visitado por última vez 10/02/17.
- [43] Ganttter *Collaborative cloud scheduling made easy*.
<http://ganttter.com>. visitado por última vez 10/02/17.
- [44] Trello *la manera gratuita, flexible y visual de organizarlo todo con cualquiera*.
<https://trello.com>. visitado por última vez 10/02/17.
- [45] DIA *a program to draw structured diagrams*.
<http://dia-installer.de>. visitado por última vez 10/02/17.
- [46] Pencil *An open-source GUI prototyping tool that's available for ALL platforms*.
<http://pencil.evolus.vn>. visitado por última vez 10/02/17.
- [47] Ministerio de las TIC Colombia. *Estadísticas de Internet en el sector de las TIC*
<https://goo.gl/aiQldQ>