

DICIEMBRE DE 2015



DOCUMENTO DE PROYECTO DE GRADO
SOFTWARE PARA LA GESTIÓN DE INVENTARIOS CON
RADIO FRECUENCIA

JUAN ESTEBAN RÍOS HENAO
SANTIAGO VELÁSQUEZ MONTOYA
SEBASTIAN PALACIO FLOREZ
POLITÉCNICO GRANCOLOMBIANO - MEDELLÍN

DOCUMENTO DE PROYECTO DE GRADO
SOFTWARE PARA LA GESTION DE INVENTARIOS CON RADIO FRECUENCIA

JUAN ESTEBAN RIOS
SANTIAGO VELASQUEZ MONTOYA
SEBASTIAN PALACIO FLOREZ

POLITÉCNICO GRANCOLOMBIANO INSTITUCIÓN UNIVERSITARIA
FACULTAD DE INGENIERÍA Y CIENCIAS BÁSICAS
Diciembre 2015

AGRADECIMIENTOS

A pesar de los retos implícitos en el desarrollo de este proyecto, cabe resaltar un merecido reconocimiento a todas aquellas personas que directa o indirectamente aportaron con herramientas, conocimiento, tiempo y dedicación para culminarlo con éxito.

Dedicamos esta tesis principalmente a nuestras familias, especialmente a la familia Ríos Henao ya que brindó el espacio de trabajo que dio vida a este proyecto.

A Diego Iván Oliveros por permitirnos brindar lo mejor de cada uno de nosotros al exigirnos un proyecto con altos estándares de calidad, todo esto bajo su guía y enseñanza.

A los desarrolladores de herramientas y marcos de trabajo que nos permitieron agilizar en gran medida el desarrollo de las aplicaciones móvil y web.

A la comunidad activa de desarrollares en los frameworks utilizados ya que permitieron dar solución a todos los problemas que surgieron en la fase del desarrollo.

A nuestros maestros quienes nos aportaron gran conocimiento e ayudaron a crecer como personas.

A Johany Carreño por atender nuestras inquietudes y por estar pendiente en cada fase del desarrollo de nuestras carreras.

Para Álvaro Javier Pérez por la asesoría en la tecnología NFC y su correcta implementación.

CONTENIDO

AGRADECIMIENTOS	2
CONTENIDO.....	3
CONTENIDO ILUSTRACIONES	5
CONTENIDO TABLAS	6
CONTENIDO ANEXOS.....	7
GLOSARIO	8
SIGLAS	9
PRESENTACIÓN.....	10
PLANTEAMIENTO DEL PROBLEMA	11
JUSTIFICACIÓN	12
MARCO TEÓRICO.....	13
OBJETIVOS	17
METODOLOGÍA	18
Product Owner	18
El Equipo de Desarrollo.....	18
El Scrum Master.....	19
PLANEACIÓN	20
TECNOLOGIAS PARA EL DESARROLLO DEL PROYECTO.....	25
Base de datos MySQL.....	25
PHP.....	25
Symfony2	25
HTML5	25
CSS3.....	25
Javascript	25
Bootstrap.....	26
Ionic	26
AngularJs	26
Apache Cordova.....	26
Lector NFC.....	26
ALCANCE DEL PROYECTO	28
REQUERIMIENTOS.....	29
Requerimientos funcionales	29
Requerimientos no funcionales	31
RESTRICCIONES DEL PROYECTO.....	32
ROLES DEL APLICATIVO	33

DISEÑO DE LA ARQUITECTURA.....	34
MODELO RELACIONAL	38
Diccionario de datos	40
DESCRIPCIÓN DE LOS APLICATIVOS.....	45
Aplicativo Web.....	45
Aplicativo Móvil.....	46
PLAN DE PRUEBAS.....	50
Pruebas de código unitarias	50
Pruebas de API para el móvil	52
ESTUDIO DE CASO	54
BIBLIOGRAFIA	59

CONTENIDO ILUSTRACIONES

Ilustración 1 "Diagrama de Gantt I"	21
Ilustración 2 "Diagrama de Gantt II"	22
Ilustración 3 "Diagrama de Gantt III"	23
Ilustración 4 "Diagrama de Gantt IV"	24
Ilustración 5 "Diagrama de Arquitectura"	34
Ilustración 6 "El patrón MVC"	36
Ilustración 7 "Modelo Relacional"	39
Ilustración 8 "Vista inicial móvil"	46
Ilustración 9 "Lectura de tag"	47
Ilustración 10 "Consulta de producto"	47
Ilustración 11 "Asignar tag o foto al producto"	47
Ilustración 12 "Creación de producto"	47
Ilustración 13 "Consulta de productos I"	48
Ilustración 14 "Consulta de productos II"	48
Ilustración 15 "Movimiento a producto I"	49
Ilustración 16 "Movimiento a producto II"	49
Ilustración 17 "Ubicar producto"	49
Ilustración 18 "Validar inventario"	50
Ilustración 19 "Chequear producto"	50
Ilustración 21 "Unit test Categoría"	51
Ilustración 22 "Resultado Unit test"	52
Ilustración 23 "Api doc"	52
Ilustración 24 "Pruebas Api"	53
Ilustración 25 "Plano de las instalaciones"	56

CONTENIDO TABLAS

Tabla 1 "Requisitos funcionales"	30
Tabla 2 "Requisitos no funcionales"	31
Tabla 3 "Entidad categoria"	40
Tabla 4 "Entidad conteo"	40
Tabla 5 "Entidad detalle_conteo"	40
Tabla 6 "Entidad inventario"	41
Tabla 7 "Entidad moviento"	41
Tabla 8 "Entidad movimiento_detalle"	41
Tabla 9 "Entidad producto"	42
Tabla 10 "Entidad producto_unico"	42
Tabla 11 "Entidad tipo_movimiento"	43
Tabla 12 "Entidad ubicacion"	43
Tabla 13 "Entidad usuario"	44

CONTENIDO ANEXOS

Anexo 1 “LogsUnitTes”	51
Anexo 2 “Formato de Registro de Mercancía 08-2015”	55

GLOSARIO

API: Es el conjunto de subrutinas, funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

Arquitectura de software: Se refiere a las estructuras de un sistema, compuestas de elementos con propiedades visibles de forma externa y las relaciones que existen entre ellos.

Framework: Se define como un marco de trabajo diseñado para facilitar las labores de desarrollo.

Lenguaje de programación: Se define como un lenguaje formal diseñado para realizar procesos que pueden ser llevados a cabo por máquinas como las computadoras. Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana.

NFC: Es una tecnología de comunicación inalámbrica, de corto alcance y alta frecuencia que permite el intercambio de datos entre dispositivos.

Prueba unitaria: Una prueba unitaria es una forma de comprobar el correcto funcionamiento de un módulo de código. Esto sirve para asegurar que cada uno de los módulos funcione correctamente por separado.

Patrón MVC: Es un patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones

Retail: El detal o venta al detalle (en inglés retail) es un sector económico que engloba a las empresas especializadas en la comercialización masiva de productos o servicios uniformes a grandes cantidades de clientes.

SCRUM: Es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto.

Tag NFC: Etiqueta que utiliza la tecnología NFC permitiendo utilizar sus funcionalidades.

SIGLAS

API: (*Application Programming Interface*) Interfaz de programación de aplicaciones.

CSS: (*Cascading Style Sheets*) Hoja de estilo en cascada.

DB: (*Data Base*) Base de datos.

HTML: (*HyperText Markup Language*) Lenguaje de marcas de hipertexto.

HTTP: (*Hypertext Transfer Protocol*) Protocolo de transferencia de hipertexto.

NFC: (*Near field communication*) Comunicación de campo cercano.

MVC: (*Model, View, Controller*) Modelo, vista, controlador.

RFID: (*Radio Frequency IDentification*) Identificación por radiofrecuencia.

UML: (*Unified Modeling Language*) Lenguaje de modelado de sistemas de software.

PRESENTACIÓN

Los procesos de inventarios están divididos en materias primas, productos, y suministros. Un inventario es crucial para el correcto funcionamiento de una empresa y/o negocio ya que los artículos como tal forman parte del programa de mantenimiento, o de algo más complejo, como una combinación de materias primas que forman parte de un proceso de manufactura. (Müller, 2005)

Hoy en día el proceso de inventario en muchos lugares se realiza de manera manual, donde luego de un proceso demasiado extenso se pretende hallar un resultado final donde el resultado es poder conocer el estado del inventario.

Este proceso que realizan los encargados del inventario, puede llevarse a cabo en un lapso de tiempo de 1 a 3 días aproximadamente, esto dependiendo proporcionalmente del tamaño del inventario.

La gestión administrativa de una empresa tiene previsto como uno de sus principales roles analizar todos y cada uno de los procesos que repercuten en su funcionamiento, entre estas funciones se incluye la etapa de evaluación y análisis de las tecnologías existentes en la empresa y las ofrecidas por el mercado, permitiendo la adopción de sistemas tecnológicos como el RFID, tecnología que utiliza la identificación por radiofrecuencia mediante un número único de serie; la cual se inició en la segunda guerra mundial como una herramienta de espionaje que se incorporaba en los aviones para la identificación del enemigo. (Gómez, 2012)

Un sistema de inventarios con RFID le permite a los usuarios satisfacer sus necesidades tanto por agilidad y por ahorro de tiempo al momento de realizar un inventario, la facilidad de registrar uno o todo un lote de productos, el ingreso a bodegaje, el conteo que se debe de hacer a todas las existencias físicas donde indica si esta correcto o por el contrario darse el caso de faltantes y/o mercancía en deterioro, y por último la salida de algún producto del inventario.

PLANTEAMIENTO DEL PROBLEMA

Nena Te Viste es un local de ropa ubicado en el municipio de Envigado, el cual lleva más de 9 años ofreciendo a sus clientes gran variedad de ropa y accesorios importados, que por su buen precio y calidad ha logrado muy buena acogida y fidelización de sus clientes, cada 3 meses Nena te viste importa alrededor de 1.200 prendas diferentes los cuales distribuye y organiza en su almacén.

Actualmente utilizan un archivo en Excel para tener una gestión de las ventas como las compras de sus productos, el cual importan a un sistema de facturación, esto es un trabajo muy tedioso para saber cuál es el estado de su inventario real, que productos tienen y cuales no en su almacén.

El problema se pone aún peor cuando quieren hacer un inventario físico de todos los productos de su almacén, este proceso ha llegado a tardar hasta tres días, en los cuales el almacén cierra sus ventas causando algunas pérdidas.

Si luego de terminar un inventario físico hay inconsistencias en uno o varios productos, es difícil saber qué fue lo que pasó con estos productos.

Es por esto que a través este proyecto se presentara una solución detallada que responde a cada una de las necesidades anteriormente planteadas. Creando un sistema de información a la medida para satisfacer todos los requerimientos.

JUSTIFICACIÓN

-¿Por qué es relevante este proyecto en el área de conocimiento escogida?

Al ofrecer una solución basada en identificación por radiofrecuencia (RFID) y un aplicativo web que le permita a nuestro cliente tener un óptimo control de su inventario, evitando pérdida de mercancía y sobre costos ya que se agiliza en gran medida el proceso de conteo manual al remplazarlo por conteo electrónico, lo que conlleva a reducir al máximo el error humano. Todo esto genera un alto impacto a nivel de productividad y rentabilidad lo que a su vez repercute en la economía de las diferentes empresas y/o negocios dedicados a la producción y distribución tanto de prendas como accesorios de vestir.

-¿Por qué es relevante como trabajo de grado en el contexto del politécnico gran colombiano?

La importancia trasciende en el desarrollo de un proyecto innovador basado en nuevas tecnologías aplicadas a la rama de ingeniería de sistemas de manera que nos permita complementar y poner en práctica el conocimiento adquirido y del mismo modo desarrollar métodos de investigación y desarrollo que nos conviertan en profesionales de calidad.

MARCO TEÓRICO

En esta investigación se buscó como referencias o conceptos anteriores frente a su problema en nuestro caso los conceptos que hemos abstraído para desarrollar nuestra solución plena o parcial frente al problema dado son los siguientes fragmentados en disciplinares (análisis de requerimiento, ingeniería de requisitos, requerimientos funcionales y no funcionales, metodología del desarrollo de software, procesos del desarrollo de software, metodología del proceso del desarrollo de software, procesos del desarrollo del software) y temáticos (documentos, archivos, gestión documental), los primeros son los conceptos bases para desarrollar nuestra solución, los últimos son del entorno en el cual vamos a investigar.

En el ámbito de la informática y los sistemas de información basada en aplicativos y sistemas software existe un mundo de necesidades y requerimientos por solucionar, por tal motivo existen pasos para el desarrollo de un sistema software.

En esto procesos existen conceptos que indican los intereses a resolver como los siguientes a tratar.

El requerimiento se define como “condición o capacidad que un usuario necesita para poder resolver un problema o lograr un objetivo, condición o capacidad que debe exhibir o poseer un sistema para satisfacer un contrato, estándar, especificación, u otra documentación formalmente impuesta. Una condición o capacidad que debe ser conformada por el sistema (SCRUM). Algo que el sistema debe hacer o una cualidad que el sistema debe poseer (Robertson - Robertson) [n1].

Entonces entendemos por análisis de requerimiento todo el proceso y análisis que se le hace a las condiciones que da un usuario, con relación a esto existe la ingeniería de requisitos la cual hace parte del análisis de requerimiento, el IEEE define requisito como condición o aptitud necesaria para resolver un problema o alcanzar un objetivo. Este concepto se ve reflejado en la ingeniería de requisitos lo cual se define por el IEEE como proceso de estudio y refinamiento de un sistema, aquí entran las formas de recolección de información para adaptarla a los requerimientos y requisitos del sistema y usuario. Luego de esto se definen los requisitos del sistema o mejor los requisitos funcionales los cuales se definen y relacionan con los no funcionales como “comportamiento interno del software: cálculos, detalles técnicos, manipulación de datos y otras funcionalidades específicas que muestran cómo los casos de uso serán llevados a la práctica. Son complementados por los requerimientos no funcionales, que se enfocan en cambio en el diseño o la implementación”. [n1]

Después de haber investigado, referenciado y examinado fuentes y tener fijos los requisitos y requerimientos del problema, se llega al punto del estudio de la viabilidad la cual es la “condición que hace posible el funcionamiento del sistema, proyecto o idea al que califica, atendiendo a sus características tecnológicas y a las leyes de la naturaleza involucradas. [n1]

Este estudio debe ser en el ámbito legal, económico y de la empresa para su validación, este paso dice en pocas palabras si se hace o no el proyecto en este caso el software. Como se dijo antes en este punto el proyecto es decisivo en su terminación, luego de una viabilidad aceptada en todos los aspectos antes mencionados viene el paso de un lenguaje técnico a uno costo entendible por el usuario de lo que va a hacer el software.

Los anteriores conceptos dados por pasos pueden ser rígidos sin una metodología del desarrollo de software, Cuando los proyectos que se van a desarrollar son de mayor amplitud, ahí si toma sentido el basarnos en una metodología de desarrollo, y empezamos a buscar cual sería la más apropiada para nuestro caso. Lo cierto es que muchas veces no encontramos la más adecuada y terminamos por hacer o diseñar nuestra propia metodología, algo que por supuesto no está mal, siempre y cuando cumpla con el objetivo.

Muchas veces realizamos el diseño de nuestro software de manera rígida, con los requerimientos que el cliente nos solicitó, de tal manera que cuando el cliente en la etapa final, solicita un cambio se nos hace muy difícil realizarlo, pues si lo hacemos, altera muchas cosas que no habíamos previsto, y es justo éste, uno de los factores que ocasiona un atraso en el proyecto y por tanto la incomodidad del desarrollador por no cumplir con el cambio solicitado y la incomodidad por parte del cliente por no tomar en cuenta su pedido. Obviamente para evitar estos incidentes debemos haber llegado a un acuerdo formal con el cliente, al inicio del proyecto, de tal manera que cada cambio o modificación no perjudique al desarrollo del mismo.

Por experiencia, muchas veces los usuarios finales, se dan cuenta de las cosas que dejaron de mencionar, recién en la etapa final del proyecto, pese a que se les mostró un prototipo del software en la etapa inicial del proyecto.

Los proyectos en problemas son los que salen del presupuesto, tienen importantes retrasos, o simplemente no cumplen con las expectativas del cliente.

La metodología más importante en esta investigación fue SCRUM:

SCRUM es un modelo de referencia que define un conjunto de prácticas y roles, y que puede tomarse como punto de partida para definir el proceso de desarrollo que se ejecutará durante un proyecto. En donde se denomina los marcos de desarrollo ágiles para adoptar una estrategia de desarrollo incremental, en lugar de la planificación y ejecución completa del producto, al igual que basarse en la calidad del resultado más en el conocimiento tácito de las personas en equipos auto organizados, que en la calidad de los procesos empleados.

Un principio clave de SCRUM es el reconocimiento de que durante un proyecto los clientes pueden cambiar de idea sobre lo que quieren y necesitan, y que los desafíos impredecibles no pueden ser fácilmente enfrentados de una forma predictiva y planificada.

En conclusión esta metodología, se recomienda para proyectos de corto plazo y A través de este marco de trabajo se conoce la velocidad media del equipo por sprint, con lo que es posible estimar de manera fácil cuando se podrá hacer uso de una determinada funcionalidad. [n2]

Luego de definir conceptos disciplinarios pasamos a los temáticos que abarcan en el cual se va a investigar, comenzaremos desde un ámbito general a o particular, los conceptos del problema que nos interesan.

Para iniciar el camino a nuestro problema a tratar que es la gestión de información documentada o la gestión documental, definamos el inventario como “es el testimonio material de un hecho o acto realizado en el ejercicio de sus funciones por instituciones o personas físicas, jurídicas, públicas o privadas, registrado en una unidad de información en cualquier tipo de soporte (papel, cintas, discos magnéticos, películas,

fotografías, etcétera) en lenguaje natural o convencional. Es el testimonio de una actividad del hombre fijado en un soporte. [n3]. Este es nuestro concepto más importante a tratar en nuestra investigación ya que nos genera una gestión documental que debe ser administrada por el sistema, definimos entonces gestión documental como “el conjunto de normas, técnicas y prácticas usadas para administrar el flujo de productos de todo tipo, permitir la recuperación de información desde ellos, determinar el tiempo que los productos deben almacenar los productos que ya fueron ingresado y/o vendidos asegurando la conservación definida de los productos, al igual que el proceso de la realización de conteos para el inventario por medio de la tecnología de radiofrecuencia, aplicando principios de racionalización y economía. [n4]

Plantados y entendidos los conceptos disciplinares y temáticos que refleja la investigación de la gestión de un inventario y la creación de un software que aplique estas características, concluimos que se dará solución a todo el proceso de gestión de productos e inventario físico.

Actualmente en la ingeniería de sistemas y en el desarrollo de software se están utilizando patrones de abstracción de problemas como son la programación orientada a objetos el análisis y desarrollo orientados a objetos es un dato en común en la actualidad es un “enfoque de la ingeniería de software que modela un sistema como un grupo de objetos que interactúan entre sí. Este enfoque representa un dominio en términos de conceptos compuestos por verbos y sustantivos, clasificados de acuerdo a su dependencia funcional.

En éste método de análisis y diseño se crea un conjunto de modelos utilizando una notación acordada como, por ejemplo, el lenguaje unificado de modelado (UML). ADOO aplica técnicas de modelado de objetos para analizar los requerimientos para un contexto por ejemplo, un sistema de inventario, y un conjunto de módulos de software para diseñar una solución para mejorar los procesos involucrados no está restringido al diseño de programas de computadora, sino que cubre sistemas enteros de distinto tipo. Las metodologías de análisis y diseño más modernas son guiadas a través de requerimientos, diseño, implementación, pruebas, y despliegue. El lenguaje unificado de modelado se ha vuelto el lenguaje de modelado de estándar más usado en análisis y diseño orientado a objetos. [n5]

“La arquitectura de software, tiene que ver con el diseño y la implementación de estructuras de software de alto nivel. Es el resultado de ensamblar un cierto número de elementos arquitectónicos de forma adecuada para satisfacer la mayor funcionalidad y requerimientos de desempeño de un sistema, así como requerimientos no funcionales, como la confiabilidad, escalabilidad, portabilidad, y disponibilidad. De tal modo se tiene que implementar una arquitectura novedosa y robusta para nuestra aplicación dicha arquitectura es la MVC (Modelo vista controlador), “es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. El patrón MVC se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el Sistema de Gestión de Base de Datos y la Lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista. [n6]

Siendo que los datos se expresan en un modelo, y que su pertinencia necesitamos una base de datos que se define como “banco de datos (en inglés: database) es un conjunto de datos pertenecientes a un mismo contexto y almacenados

sistemáticamente para su posterior uso. En este sentido, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. En la actualidad, y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital (electrónico), que ofrece un amplio rango de soluciones al problema de almacenar datos. [n7]

La pertinencia de esos datos se debe modelar para llegar a una relación entre datos. Los datos para que se puedan ver como una información coherente, hablamos de un modelo relacional definido. Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar por un usuario no experto. La información puede ser recuperada o almacenada por medio de «consultas» que ofrecen una amplia flexibilidad y poder para administrar la información.

Este modelo considera la base de datos como una colección de relaciones. De manera simple, una relación representa una tabla que no es más que un conjunto de filas, cada fila es un conjunto de campos y cada campo representa un valor que interpretado describe el mundo real. Cada fila también se puede denominar como un registro y a cada columna se le puede llamar campo o atributo.

El lenguaje que se usó para construir las consultas a bases de datos relacionales es MySQL, es un estándar implementado por los principales motores o sistemas de gestión de bases de datos relacionales. [n8]

OBJETIVOS

Objetivo general: Desarrollar un sistema que permita gestionar los procesos de almacenamiento, despacho e inventario de negocios dedicados al detal de vestuario, a través de la tecnología RFID (Identificación por radio frecuencia).

Objetivos específicos:

- Analizar las necesidades del negocio para cada uno de los procesos de elaboración del inventario con el actual sistema basado en excel, que permitan identificar los requerimientos y de este modo diseñar una solución.
- Diseñar la aplicación móvil y web de acuerdo a las especificaciones y los análisis realizados.
- Incrementar en un 70% el nivel de detalle de las características que conforman la información de cada activo que entre y salga de la bodega.
- Aumentar la seguridad de los activos en un 50%, teniendo un mayor control de cada uno de ellos al conocer explícitamente su historial desde el ingreso a bodega hasta que sale esta.
- Optimizar en un 20% la distribución de la mercancía en bodega permitiendo asignar un lugar a cada artículo y logrando un mayor aprovechamiento de los espacios de almacenamiento.
- Incrementar hasta en un 99% la exactitud en los inventarios.
- Optimizar los recuentos de la mercancía entre 75 hasta 92% más rápido.
- Alcanzar niveles de reducción en la falta de existencias hasta en un 80%.

METODOLOGÍA

Para el desarrollo de este proyecto se utilizó SCRUM, originalmente se formalizó para proyectos de desarrollo de software, pero funciona bien para cualquier complejo alcance innovador de trabajo, esta metodología requiere una nueva forma de pensar y ajustes culturales en general. Cuando los equipos y las organizaciones se comprometen plenamente a SCRUM, descubrirán un nuevo sentido de la flexibilidad, la creatividad y la inspiración todo lo cual dará lugar a mejores resultados.

Mediante el uso de SCRUM, se puede ser más ágil, descubriendo cómo reaccionar más rápidamente y responder con mayor precisión a los cambios inevitables que se le presente. Y al permanecer centrado, colaborar y comunicarse entre el grupo de trabajo, puede lograr lo que realmente hay que hacer con éxito

El corazón de SCRUM es un Sprint, en un tiempo de un mes o menos durante el cual se crea un incremento del producto "Hecho", utilizable, y potencialmente liberable. Los Sprints tienen duraciones consistentes a lo largo de un esfuerzo de desarrollo. Un nuevo Sprint se inicia inmediatamente después de la conclusión de la Sprint anterior.

Sprints contienen y consisten en la planificación de Sprint, SCRUM diarias, el trabajo de desarrollo, la Revisión del Sprint y Sprint Retrospectivo.

El equipo SCRUM consiste en un Product Owner, el equipo de desarrollo, y SCRUM Master. Este equipo es multifuncional ya que tienen todas las competencias necesarias para realizar el trabajo sin depender de otros que no forman parte del equipo. El modelo de equipo en SCRUM está diseñado para optimizar la flexibilidad, la creatividad y la productividad.

Product Owner

El Product Owner es responsable de maximizar el valor del producto y el trabajo del equipo de desarrollo. ¿Cómo se hace esto puede variar ampliamente a través de las organizaciones, Scrum equipos e individuos?.

El product owner puede hacer el trabajo por encima de, o tener el equipo de desarrollo de hacerlo. Sin embargo, el product owner sigue siendo responsable.

Un product owner es una persona, no un comité. El propietario del producto puede representar los deseos de un comité en el Product Backlog, pero aquellos que quieran cambiar la prioridad de un elemento Product Backlog deben abordar el propietario del producto.

El Equipo de Desarrollo

El equipo de desarrollo está formado por profesionales que hacen el trabajo de la entrega de un incremento potencialmente liberable del producto "Hecho" al final de cada Sprint. Sólo los miembros del equipo de desarrollo crear el Incremento.

Equipos de Desarrollo se estructuran y facultado por la organización para organizar y gestionar su propio trabajo. La sinergia resultante optimiza la eficiencia y la eficacia general del equipo de desarrollo.

El Scrum Master

El SCRUM Master es responsable de asegurarse de que SCRUM se entienda y se promulgue. SCRUM Masters hacen esto al asegurar que el Equipo SCRUM se adhiere a la teoría de SCRUM, prácticas y normas.

El SCRUM Master es un líder para el equipo SCRUM. El SCRUM Master ayuda a quienes están fuera del equipo SCRUM a entender cuáles de sus interacciones con el Equipo SCRUM son útiles y cuáles no. El SCRUM Master ayuda a todos a cambiar estas interacciones para maximizar el valor creado por el Equipo SCRUM.

PLANEACIÓN

El objetivo de la Planificación del proyecto de Software es proporcionar un marco de trabajo que permita al gestor hacer estimaciones razonables de recursos costos y planificación temporal. Estas estimaciones se hacen dentro de un marco de tiempo limitado al comienzo de un proyecto de software, y deberían actualizarse regularmente medida que progresa el proyecto. Además las estimaciones deberían definir los escenarios del mejor caso, y peor caso, de modo que los resultados del proyecto pueden limitarse.

Mediante el uso de un diagramas se programó las actividades, esencialmente es este diagrama que contiene barras que representan cada una de las actividades, y cuya longitud representa la duración de la actividad respectiva.

Este diagrama de Gantt de una dimensión es un calendario, siendo una técnica muy usada para la planeación de las actividades que se desarrollan en serie pero cuando las actividades que pueden llevarse de manera simultánea son varias resulta apropiado un diagrama de Gantt bidimensional.

La ventaja principal del diagrama de Gantt es su sencillez, nosotros como analistas de sistemas no solo encontramos fácil esta metodología, sino que también contamos con un excelente instrumento de comunicación con los usuarios finales. Ver *Ilustración 1 "Diagrama de Gantt I"*.

Nombre	Duración	% Completado	Inicio	Fin
☐ Proyecto - Análisis	45.5d	100%	01/08/2015	13/09/2015
Anteproyecto	15d	100%	01/08/2015	15/08/2015
Ajustes anteproyecto	7d	100%	07/09/2015	13/09/2015
Análisis y requerimientos	14d	100%	17/08/2015	30/08/2015
Modelo entidad relación	7d	100%	24/08/2015	30/08/2015
☐ Proyecto - Conclusiones	124.5d	100%	01/08/2015	29/11/2015
Estado del arte	28d	100%	02/11/2015	29/11/2015
Artículo científico	28d	100%	02/11/2015	29/11/2015
Documento de tesis	28d	100%	02/11/2015	29/11/2015
Borrador del documento	1d	100%	01/08/2015	01/08/2015
☐ Desarrollo Web	57.5d	100%	31/08/2015	25/10/2015
Login	7d	100%	31/08/2015	06/09/2015
☐ Maestro Productos	28.75d	100%	31/08/2015	27/09/2015
Listar Productos	7d	100%	31/08/2015	06/09/2015
Crear Productos	7d	100%	21/09/2015	27/09/2015
Modificar Productos	7d	100%	21/09/2015	27/09/2015
☐ Maestro Producto Único	7d	100%	28/09/2015	04/10/2015
Crear Producto Único	7d	100%	28/09/2015	04/10/2015
Modificar Producto Único	7d	100%	28/09/2015	04/10/2015
☐ Maestro Ubicaciones	21.5d	100%	07/09/2015	27/09/2015
Listar Ubicación	7d	100%	07/09/2015	13/09/2015
Crear Ubicaciones	7d	100%	21/09/2015	27/09/2015
Modificar Ubicaciones	7d	100%	21/09/2015	27/09/2015

Ilustración 1 "Diagrama de Gantt I"

☐ Maestro Categorías	21.5d	100%	07/09/2015	27/09/2015
Listar Categorías	7d	100%	07/09/2015	13/09/2015
Crear Categorías	7d	100%	21/09/2015	27/09/2015
Modificar Categorías	7d	100%	21/09/2015	27/09/2015
☐ Maestro Tipo Movimiento	21.5d	100%	07/09/2015	27/09/2015
Listar Tipo Movimiento	7d	100%	07/09/2015	13/09/2015
Crear Tipo Movimiento	7d	100%	21/09/2015	27/09/2015
Modificar Tipo Movimiento	7d	100%	21/09/2015	27/09/2015
☐ Maestro Movimientos	7d	100%	28/09/2015	04/10/2015
Listar Movimientos	7d	100%	28/09/2015	04/10/2015
Crear Movimientos	7d	100%	28/09/2015	04/10/2015
Modificar Movimientos	7d	100%	28/09/2015	04/10/2015
☐ Maestro Usuarios	7d	100%	28/09/2015	04/10/2015
Listar Usuario	7d	100%	28/09/2015	04/10/2015
Crear Usuario	7d	100%	28/09/2015	04/10/2015
Modificar Usuario	7d	100%	28/09/2015	04/10/2015
☐ Maestro Inventario (conteos)	14d	100%	12/10/2015	25/10/2015
Crear conteos	14d	100%	12/10/2015	25/10/2015
Diferencia de conteos	14d	100%	12/10/2015	25/10/2015
Ajustes de inventario	14d	100%	12/10/2015	25/10/2015
☐ Generar Reportes	7d	100%	05/10/2015	11/10/2015
Reporte - Estado del inventario	7d	100%	05/10/2015	11/10/2015
Reporte - Detalle movimientos	7d	100%	05/10/2015	11/10/2015
Reporte - Movimiento productos	7d	100%	05/10/2015	11/10/2015
Reporte - Detalle productos	7d	100%	05/10/2015	11/10/2015

Ilustración 2 "Diagrama de Gantt II"

☐ Desarrollo Móvil	50.25d	81%	14/09/2015	01/11/2015
☐ Login Móvil	10d	100%	14/09/2015	23/09/2015
Diseño vista y formulario login	7d	100%	14/09/2015	20/09/2015
Login e integración con API web	7d	100%	17/09/2015	23/09/2015
☐ Integración SDK NFC	7d	100%	22/09/2015	28/09/2015
Modulo integración SDK NFC	7d	100%	22/09/2015	28/09/2015
☐ Crear producto único	8d	100%	08/10/2015	15/10/2015
Consulta producto único maquetación	7d	100%	08/10/2015	14/10/2015
Lógica consulta producto único	7d	100%	09/10/2015	15/10/2015
☐ Cambiar ubicación	20.5d	100%	09/10/2015	28/10/2015
Maquetación vista reubicacion producto	7d	100%	09/10/2015	15/10/2015
Logica reasignación producto	7d	100%	22/10/2015	28/10/2015
☐ Conteos de inventario	18.25d	38%	15/10/2015	01/11/2015
Desarrollo modulo de conteos	14d	25%	15/10/2015	28/10/2015
Maquetación vistas para conteos	14d	50%	19/10/2015	01/11/2015
☐ Crear movimientos	12d	100%	14/10/2015	25/10/2015
Maquetación vista movimientos	7d	100%	19/10/2015	25/10/2015
Conexión con API movimientos	7d	100%	14/10/2015	20/10/2015
☐ Plan de Pruebas	14.25d	0%	19/10/2015	01/11/2015
Puebas Aplicativo Web	7d	0%	19/10/2015	25/10/2015
Pruebas Aplicativo Móvil	7d	0%	26/10/2015	01/11/2015

Ilustración 3 "Diagrama de Gantt III"

Otra ventaja del uso de los diagramas de Gantt que detectamos es que las barras que representan las actividades se dibujan a escala, esto es el tamaño de una barra indica la duración relativa de la actividad. Ver *Ilustración 4 "Diagrama de Gantt IV"*

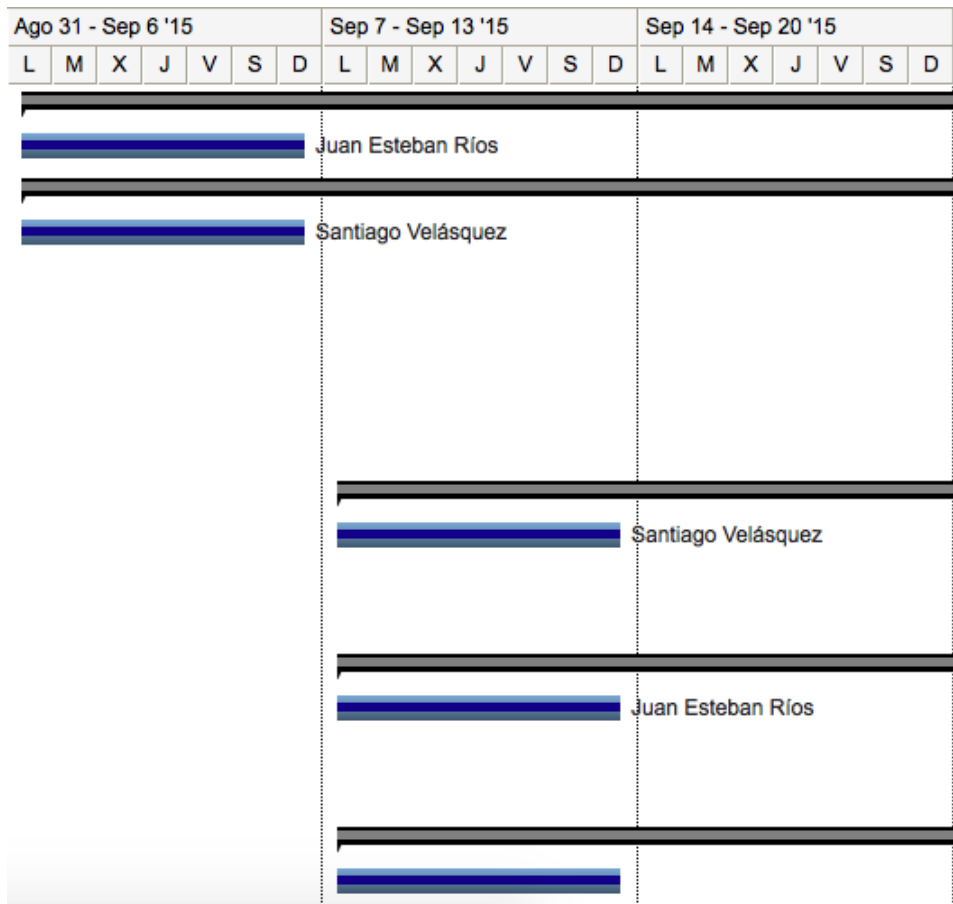


Ilustración 4 "Diagrama de Gantt IV"

TECNOLOGIAS PARA EL DESARROLLO DEL PROYECTO

Base de datos MySQL

MySQL es un sistema de gestión de bases de datos relacional muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones. Sea cual sea el entorno en el que va a utilizar MySQL, es importante monitorizar de antemano el rendimiento para detectar y corregir errores tanto de SQL como de programación.

PHP

PHP es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página Web resultante.

Symfony2

Symfony es un conjunto de componentes de PHP reutilizables Y un framework PHP para proyectos web y es la fundación de serie en la que se construyen las mejores aplicaciones PHP.

Es un conjunto de componentes de PHP, un marco de aplicación Web, una filosofía y una Comunidad - todos trabajando juntos en armonía. El marco que lleva PHP para crear sitios web y aplicaciones web.

HTML5

HTML5 es la última versión de HTML. El término representa dos conceptos diferentes. El primero es que se trata de una nueva versión de HTML, con nuevos elementos, atributos y comportamientos. El segundo es el que contiene un conjunto más amplio de tecnologías que permite a los sitios Web y a las aplicaciones ser más diversas y de gran alcance. A este conjunto se le llama HTML5.

CSS3

CSS es un lenguaje para definir el estilo o la apariencia de las páginas web, escritas con HTML o de los documentos XML. CSS se creó para separar el contenido de la forma, a la vez que permite a los diseñadores mantener un control mucho más preciso sobre la apariencia de las páginas.

Javascript

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y

páginas web dinámicas⁴ aunque existe una forma de JavaScript del lado del servidor. Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

Bootstrap

Bootstrap es libre y de código abierto, provee una colección de herramientas para la creación de sitios y aplicaciones web. Contiene HTML - y CSS plantillas de diseño basados en la tipografía, formas, botones, navegación y otros componentes de la interfaz, así como extensiones opcionales en JavaScript . Su objetivo es facilitar el desarrollo de sitios dinámicos y aplicaciones web.

Ionic

Ionic permite crear aplicaciones multi-plataforma utilizando HTML5, CSS y componentes Javascript muy ligados de AngularJS. Posee un buen rendimiento, uno de los mayores inconvenientes de las aplicaciones híbridas con respecto a las nativas, con decisiones como la ausencia de jQuery, la minimización de la manipulación del DOM o transiciones aceleradas por hardware. Otra de las características más destacadas es la inclusión de una potente interfaz de línea de comando

Es gratuito y open source, corre sobre Cordova, tiene una parte visual realmente atractiva y cuenta con una documentación más que aceptable.

AngularJs

AngularJS es un framework de JavaScript de código abierto, mantenido por Google, que ayuda con la gestión de lo que se conoce como aplicaciones de una sola página. Su objetivo es aumentar las aplicaciones basadas en navegador con capacidad de Modelo Vista Controlador (MVC), en un esfuerzo para hacer que el desarrollo y las pruebas sean más fáciles.

Apache Cordova

Apache Cordova es un framework de licencia libre que cuenta con muchas Apis de diversos dispositivos móviles para desarrollar aplicaciones nativas dentro de un smartphone. Cada vez está tomando más énfasis en el mundo de los programadores y facilita el desarrollo de aplicaciones que utilizan tecnologías web tales como HTML, CSS y JavaScript.

Lector NFC

NFC significa (Near Field Communication). Se trata de una tecnología inalámbrica que funciona en la banda de los 13.56 MHz (en esa banda no hace falta licencia para usarla) y que deriva de las etiquetas RFID, las cuales están presentes en abonos de transporte o incluso sistemas de seguridad de tiendas físicas.

También es una plataforma abierta pensada desde el inicio para teléfonos y dispositivos móviles. Su tasa de transferencia puede alcanzar los 424 kbit/s por lo que

su enfoque más que para la transmisión de grandes cantidades de datos es para comunicación instantánea, es decir, identificación y validación de equipos/personas.
[n9]

ALCANCE DEL PROYECTO

La Gestión de inventarios es mucho más que solo comprar y vender productos, ya que con el software se podrá realizar de manera más exacta no solo el manejo de productos, como son las compras o ventas, las devoluciones ya sea según el caso, la realización de conteos, la gestión como lo son modificación y/o eliminación de productos con los cuales se están trabajando.

Todos los productos van a estar asociados a categorías y otras opciones más, para así poderlos identificarlos más fácil, todos los encargados del manejo del inventario podrán realizar el proceso de los conteos, el cual consiste por medio de un dispositivo móvil realizar el sondeo de los productos a contar, ya por medio del aplicativo web será el encargado de aceptar y verificar estos conteos realizados u esto debe concordar en todo momento con el contenido físico y el inventario de los mismos que deben ser previamente validados por el software. Y ya por último se podrá fácilmente generar movimientos de los productos luego de tener un acceso y comodidad fácil con la gestión y orden del inventario

Mirando hacia el futuro y contando con una plataforma tecnológica dentro de una empresa que permite la conectividad al servidor vía Internet, se tendrá una aplicación que permitirá realizar una gestión del inventario adecuada.

REQUERIMIENTOS

Requerimientos funcionales

Tabla I Requisitos funcionales

REQUERIMIENTO	DESCRIPCION DEL REQUERIMIENTO
RF1	El sistema debe de permitir al usuario con ingreso a las categorías poder ingresar, modificar, eliminar y visualizar las categorías que se encuentran en el aplicativo permitiendo por medio de las categorías diferenciar los productos por medio de las categorías registradas para así poder observar diferenciar los productos, unos de otros
RF2	La persona encargada podrá asignar o reubicar las ubicaciones los productos registrados, con el motivo de poder llevar un orden de en qué lugar se encuentra ubicado algún producto en específico, dirigiéndose a la ubicación asignada que se puede encontrar en el registro del producto
RF3	El usuario registrado en el sistema le permitirá registrar tanto como modificar, eliminar y listar los tipos de movimientos, puesto que este proceso es principalmente para poder distinguir los movimientos que se van a registrar para el manejo del inventario
RF4	El sistema permitirá al usuario en registrar, modificar, eliminar y listar todos los productos necesarios, y así se poder tener un control de los productos que se están en comercializando ingresar los productos con los que se van a realizar el proceso de inventario a operar en el inventario
RF5	El usuario registrado tendrá la opción de ingresar más usuarios si son requeridos, ingresándolos por el sistema indicando su información, no olvidando el usuario y la contraseña para un ingreso posterior
RF6	Cualquier usuario que tenga acceso a realizar los movimientos que se generan en el inventario, podrá ingresar todo lo que es todas las entradas y salidas indicando su tipo de movimiento, una descripción y los productos que se requieran para su proceso indicado. Los productos tanto para entradas como para ventas no son los mismos, ya que los productos que se pueden visualizar son depende de tu movimiento, si un producto se va a ingresar se podrá ver en entradas, pero si un producto va hacer vendido u otra opción

	se observara los productos ya ingresados como un movimiento de entrada
RF7	Se podrá registrar la información completa de los productos, ingresando su costo, talla, entre otros. Es indispensable que un producto tenga un TAG, pero este proceso se debe de realizar por medio del aplicativo móvil donde también se ingresa su ubicación pero solo si es requerida
RF8	La persona que acceda al sistema y este autorizada para realizar la gestión de los inventarios físicos, podrá crear los conteos donde se encuentra el proceso de registra los productos y poder diferenciar entre los conteos los productos registrados por medio del dispositivo móvil

Tabla 1 "Requisitos funcionales"

Requerimientos no funcionales

Tabla II Requisitos no funcionales

RNF1	Gestor de base de datos
RNF2	Requisitos del servidor
RNF3	Para acceder a la aplicación será necesaria una conexión a internet
RNF4	Para acceder a la aplicación será necesario un navegador web
RNF5	La aplicación será compatible con el navegador Mozilla Firefox, Google Chrome, Opera, Internet Explorer y Safari
RNF6	La aplicación será desarrollada con el framework Symfony2, Ionic, Bootstrap y Angular
RNF7	La aplicación será desarrollada con PHP, JavaScript como lenguaje de programación
RNF8	La aplicación será desarrollada en PhpStorm y Brackets como IDE
RNF9	La aplicación será desarrollada con MySql como Base de datos y la aplicación utilizará DigitalOcean como servidor web.
RNF10	Para el desarrollo de la aplicación se utilizará GIT para el control de versiones.

Tabla 2 "Requisitos no funcionales"

RESTRICCIONES DEL PROYECTO

Para la lectura de los códigos se utilizará un lector de códigos de tipo NFC/ RFID.

El sistema solo contempla el tema de inventario (conteo, reportes) mas no de valorización del mismo ya que contamos con un cronograma estrecho en cuanto a tiempo y la implementación de valorización comprende un tema legal que nos acarrea varias reuniones con los encargados de la parte contable, esta extensión en el alcance podría afectar seriamente los tiempos de desarrollo y por ende la culminación exitosa del proyecto.

ROLES DEL APLICATIVO

Administrador: El Administrador es el usuario que tiene acceso a toda la configuración del aplicativo donde se ingresa toda la información principal para poder usar el aplicativo y así no tener ningún problema al momento de realizar sus movimientos e inventario. Este usuario también tiene autorización para realizar los movimientos como crear entrada y salida, y en inventario se podrá ingresar los productos únicos y al igual que poder visualizar los reportes en el estado de inventario.

Administrador de inventario: El usuario de administrador de Inventario solo tiene acceso al proceso del inventario físico donde se realiza el proceso de los conteos que se debe de realizar con el usuario móvil activo para así poder tener el proceso de inventario físico con su funcionamiento correcto.

Móvil: Por medio del usuario Móvil se puede acceder a los procesos de los productos, permitiendo buscarlos y a su vez crearlos, al igual que el administrador puede acceder al proceso de movimientos para crear entrada y salida, también el usuario Móvil tiene acceso a la configuración de las ubicaciones para los productos registrados ya sea por el administrador o el móvil. Por ultimo también se tiene acceso a la fase del inventario donde se realiza el proceso de los conteos de productos el cual depende del usuario de administrador de inventario que es el encargado de indicar el número de inventario que se va a validar.

Un usuario puede tener uno o todos los roles indicados, estos roles se pueden diligencian por el módulo de configuración en la opción de usuarios.

DISEÑO DE LA ARQUITECTURA

Los datos reflejados en el sector TI a escala global permiten evidenciar la penetración que han adquirido las tecnologías móviles en gran diversidad de industrias, mercados y a nivel de sociedad, permitiendo acceder a información en tiempo real independientemente del lugar en que se encuentre el usuario que la solicita, es por esto que se escoge la implementación de un servicio web que cuente con una alta disponibilidad en la nube y que pueda ser consumido tanto por clientes móviles como clientes web.

Web Service

Se puede describir como un componente de software que se encuentra alojado en un servidor y este a su vez puede ser accedido a través de internet, Posee métodos que resuelven las diferentes necesidades en cuanto a información y datos que un sistema de información pueda requerir en su ciclo de vida, es considerada hoy en día la mejor opción para compartir datos en aplicaciones móviles y web, permitiendo gran escalabilidad ya que solamente se deben actualizar los servicios centralizados disponibles en el *web service*, mas no en cada cliente final, brindan mayores estándares de seguridad ya que los aplicativos tanto web como móviles se conectan al *web service* y no directamente a la base de datos.

Se realizó un diagrama de arquitectura donde se explica cómo se divide el proceso del Cliente Web y Cliente Móvil donde ambos se conectan al *web service* donde luego para finalizar el flujo se accede a la base de datos donde se está almacenando la información del aplicativo como se puede ver a continuación. Ver *Ilustración 5 "Diagrama de Arquitectura"*

Diagrama de Arquitectura

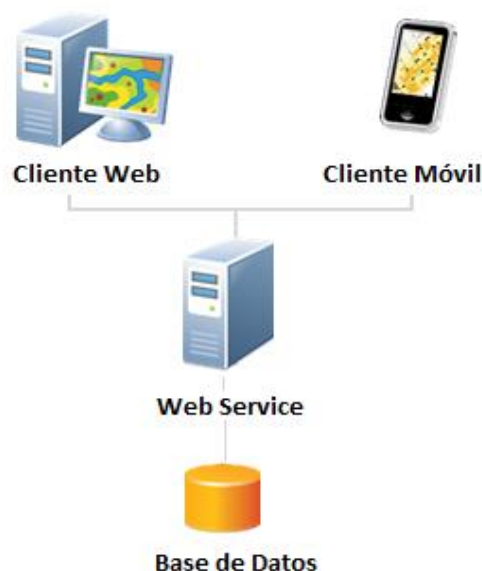


Ilustración 5 "Diagrama de Arquitectura"

La aplicación móvil debe contar con acceso a Internet como prerrequisito de funcionamiento, esta se conecta directamente a los servicios disponibles en el *web service* para solicitar o enviar información; el *web service* a su vez alimenta la base de datos con la información que el móvil le suministra manteniendo los datos del sistema de información centralizados.

En el caso del cliente web, este se conecta de la misma manera al *web service* guardando y solicitando información cada que sea necesario, los servicios de los que dispone el cliente web son en su mayoría los mismos del cliente móvil a diferencia de los módulos de reportes, creación de inventarios, creación de categorías, creación de ubicaciones, creación de usuarios.

Un ejemplo de la interacción de ambos aplicativos con el *web service* que ayude a clarificar el funcionamiento interno del sistema sería la creación de un inventario nuevo el cual se explica en detalle a continuación:

1. El usuario encargado de los inventarios debe iniciar sesión en la aplicación web con las credenciales que lo acrediten.
2. Una vez adentro del sistema debe dirigirse al menú lateral izquierdo y seleccionar la pestaña de "Inventario", por consiguiente en el submenú desplegado debe seleccionar "Inventario Físico", en este paso el cliente web envía una petición http tipo "Get" al *web service* con el fin de obtener la lista de todos los inventarios que se han llevado a cabo hasta la fecha.
3. En la parte superior derecha de la lista generada en el paso anterior el usuario dispone de un botón que le permite generar un nuevo inventario, una vez creado el inventario se le solicita al usuario diligenciar el formulario con la descripción del inventario que se está creando, posteriormente debe inicializar el inventario creando el conteo #1. En este paso se le provee al usuario un código que identifica el inventario actual y que posteriormente debe proporcionar al usuario que realiza los conteos.
4. El usuario que realizara los conteos de la mercancía debe loguearse en la aplicación móvil con las credenciales que lo acrediten, ambos aplicativos (Móvil y Web) comparten el mismo servicio de autenticación en el sistema ya que la lógica es implícitamente la misma.
5. Posteriormente debe seleccionar la opción inventarios localizada en el menú desplegable en la parte derecha del aplicativo, en esta vista se encuentra un campo de texto donde el usuario encargado del inventario debe digitar el código que se le proporciona al crear un nuevo inventario, cuando se envía el formulario la aplicación móvil envía una petición al *web service* que le debe retornar si hay o no un inventario pendiente, y si lo hay en que conteo se debe iniciar.
6. EL usuario debe iniciar el conteo en el dispositivo móvil y obtener todos los códigos que se encuentran en las etiquetas NFC, una vez ha terminado y enviado los datos, la aplicación debe enviar una petición al *web service* con los datos que se obtuvieron en el conteo, el *web service* posee un método que verifica que productos figuran como existencias en bodega.
7. El usuario encargado del inventario puede ahora proceder a verificar la diferencia de los productos que encontró el usuario operador e identificar los productos faltantes.

El proyecto web se realizó por medio de la arquitectura del framework Symfony2, que se basa en un modelo llamado Modelo Vista Controlador (MVC) que se distingue de la siguiente forma:

Symfony está basado en un patrón clásico del diseño web conocido como arquitectura MVC, que está formado por tres niveles

El Modelo representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.

La Vista transforma el modelo en una página web que permite al usuario interactuar con ella.

El Controlador se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista.

La arquitectura MVC separa la lógica de negocio (el modelo) y la presentación (la vista) por lo que se consigue un mantenimiento más sencillo de las aplicaciones. Si por ejemplo una misma aplicación debe ejecutarse tanto en un navegador estándar como en un navegador de un dispositivo móvil, solamente es necesario crear una vista nueva para cada dispositivo; manteniendo el controlador y el modelo original. El controlador se encarga de aislar al modelo y a la vista de los detalles del protocolo utilizado para las peticiones (HTTP, consola de comandos, email, etc.). El modelo se encarga de la abstracción de la lógica relacionada con los datos, haciendo que la vista y las acciones sean independientes de, por ejemplo, el tipo de gestor de bases de datos utilizado por la aplicación. Ver *Ilustración 6 "El patrón MVC"*.

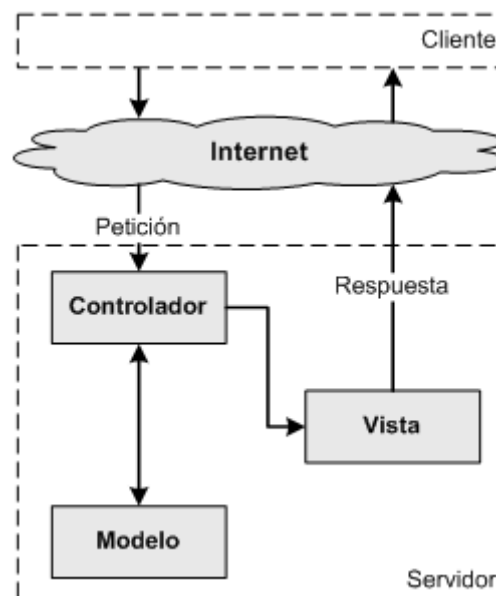


Ilustración 6 "El patrón MVC"

Las capas de la arquitectura MVC

Para poder entender las ventajas de utilizar el patrón MVC, se va a transformar una aplicación simple realizada con PHP en una aplicación que sigue la arquitectura MVC. Un buen ejemplo para ilustrar esta explicación es el de mostrar una lista con las últimas entradas o artículos de un blog.

Programación simple

Utilizando solamente PHP normal y corriente, el script necesario para mostrar los artículos almacenados en una base de datos se muestra en el siguiente listado [n10]

Para el proyecto móvil se realizó con el framework Ionic.

Ionic es un framework de desarrollo de aplicaciones Web basado en JavaScript y en el patrón Modelo-Vista-Controlador. A diferencia de otros framework, proporciona herramientas sencillas para hacer que el código HTML5 sea dinámico, es decir, que se actualice en tiempo real.

El patrón Modelo Vista Controlador que se sigue en IonicJS, separando la parte visual de la funcionalidad y las estructuras de datos.

El archivo HTML de la aplicación representa la vista y debe ser separada del controlador y el modelo.

El controlador es un objeto JavaScript que debe ser importado en el HTML que se encarga de capturar los eventos de la vista y realizar las acciones pertinentes sobre el modelo para modificar los datos.

El modelo es un objeto de JavaScript nombrado como elemento del controlador. Contiene los datos a los que va a acceder la vista y deberá contener también métodos de acceso y modificación para separar totalmente la forma a la que se acceden o modifican los datos del controlador. [n11]

MODELO RELACIONAL

Este modelo considera la base de datos como una colección de relaciones. De manera simple, una relación representa una tabla que no es más que un conjunto de filas, cada fila es un conjunto de campos y cada campo representa un valor que interpretado describe el mundo real. Cada fila también se puede denominar registro y a cada columna también se le puede llamar campo o atributo.

Una **base de datos relacional** es un conjunto de una o más tablas estructuradas en registros (líneas) y campos (columnas), que se vinculan entre sí por un campo en común, en ambos casos posee las mismas características como por ejemplo el nombre de campo, tipo y longitud; a este campo generalmente se le denomina ID, identificador o clave. Por medio de este proceso se creó este modelo relacional con base a la creación de la base de datos.

Para dar una definición más adecuada desde el punto de vista de las bases de datos, es preciso distinguir dos conceptos en la definición de la relación:

Esquema de relación: Es la parte definitoria y estática de la relación (cabecera cuando la relación se percibe como una tabla). Es invariante en el tiempo.

Extensión de la relación: Conjunto de tuplas que, en un momento determinado, satisface el esquema de la relación y se encuentran almacenadas en la base de datos. Es variante en el tiempo.

Cada relación de un diagrama Entidad/Relación se transforma directamente en una tabla. Los campos de esta tabla son las claves primarias de todas las entidades que participen en la relación más todos aquellos atributos que pudiera tener la relación.

Relación --> Tabla

El número de relaciones que componen la base de datos debe mantenerse en el mínimo posible.

Esto mismo es aplicable a las relaciones binarias con cardinalidad uno a uno, puesto que son un caso particular de las anteriores. Las tablas con un único atributo se pueden eliminar. Ver *Ilustración 7 "Modelo Relacional"*.

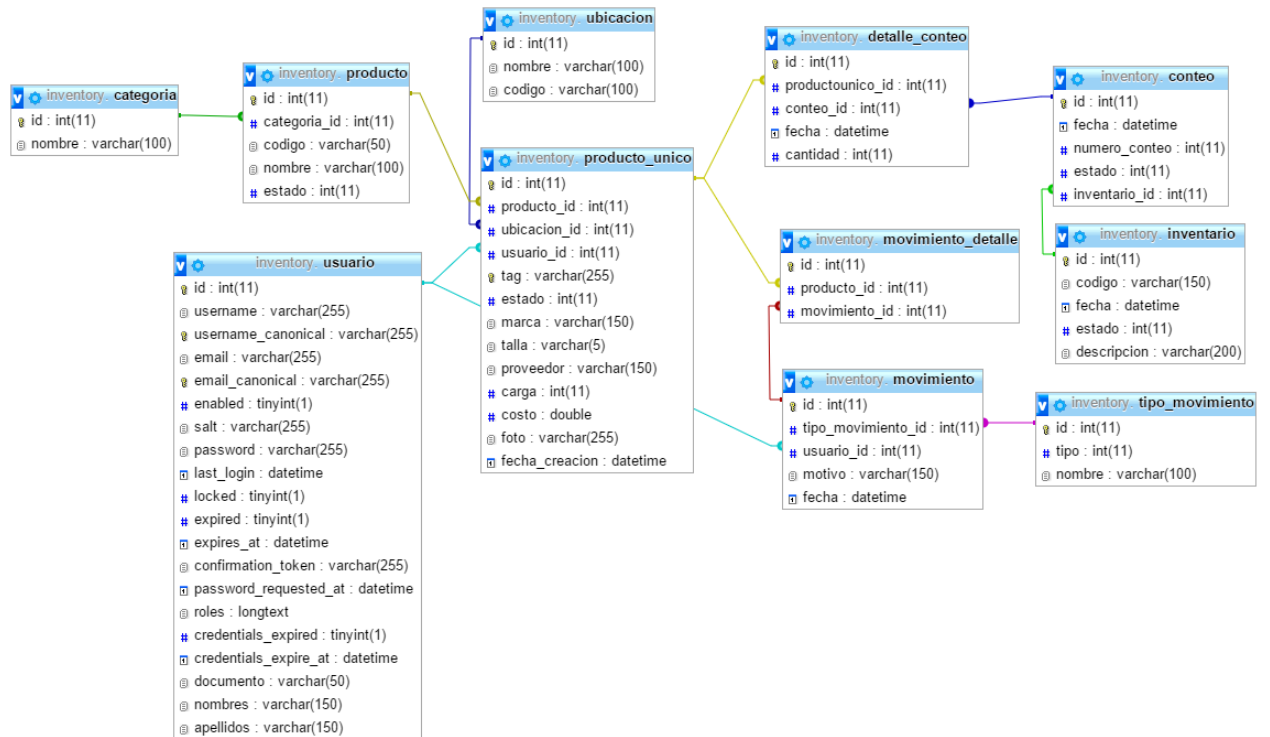


Ilustración 7 "Modelo Relacional"

Diccionario de datos

Tabla III Categoría

Columna	Tipo	Nulo
id	int(11)	No
nombre	varchar(100)	No

Tabla 3 "Entidad categoría"

Tabla IV Conteo

Columna	Tipo	Nulo	Predeterminado
id	int(11)	No	
fecha	datetime	No	now()
numero_conteo	int(11)	No	
estado	int(11)	No	
inventario_id	int(11)	Sí	

Tabla 4 "Entidad conteo"

Tabla V Detalle Conteo

Columna	Tipo	Nulo	Enlaces a
id	int(11)	No	
productounico_id	int(11)	Sí	producto_unico -> id
conteo_id	int(11)	Sí	conteo -> id
fecha	datetime	No	now()
cantidad	int(11)	No	

Tabla 5 "Entidad detalle_conteo"

Tabla VI Inventario

Columna	Tipo	Nulo	Predeterminado	Comentarios
id	int(11)	No		
codigo	varchar(150)	No		
fecha	datetime	No	now()	
estado	int(11)	No	0	0->Iniciado, 1->Finalizado
descripcion	varchar(200)	Sí		

Tabla 6 "Entidad inventario"

Tabla VII Movimiento

Columna	Tipo	Nulo	Enlaces a
id	int(11)	No	
tipo_movimiento_id	int(11)	Sí	tipo_movimiento -> id
usuario_id	int(11)	Sí	usuario -> id
motivo	varchar(150)	No	
fecha	datetime	No	

Tabla 7 "Entidad movimiento"

Tabla VIII Movimiento Detalle

Columna	Tipo	Nulo	Enlaces a
id	int(11)	No	
producto_id	int(11)	Sí	producto_unico -> id
movimiento_id	int(11)	Sí	movimiento -> id

Tabla 8 "Entidad movimiento_detalle"

Tabla IX Producto

Columna	Tipo	Nulo	Enlaces a
id	int(11)	No	
categoria_id	int(11)	Sí	categoria -> id
codigo	varchar(50)	No	
nombre	varchar(100)	No	
estado	int(11)	No	

Tabla 9 "Entidad producto"

Tabla X Producto Unico

Columna	Tipo	Nulo	Predeterminado	Enlaces a
id	int(11)	No		
producto_id	int(11)	Sí		producto -> id
ubicacion_id	int(11)	Sí		ubicacion -> id
usuario_id	int(11)	Sí		usuario -> id
tag	varchar(255)	Sí		
estado	int(11)	No		
marca	varchar(150)	Sí		
talla	varchar(5)	Sí		
proveedor	varchar(150)	Sí		
carga	int(11)	Sí		
costo	double	Sí		
foto	varchar(255)	Sí		

fecha_creacion	datetime	Sí	now()	
----------------	----------	----	-------	--

Tabla 10 "Entidad producto_unico"

Tabla XI Tipo Movimiento

Columna	Tipo	Nulo	Comentarios
id	int(11)	No	
tipo	int(11)	No	1-> Entrada, 2->Salida
nombre	varchar(100)	No	

Tabla 11 "Entidad tipo_movimiento"

Tabla XII Ubicación

Columna	Tipo	Nulo
id	int(11)	No
nombre	varchar(100)	No
codigo	varchar(100)	No

Tabla 12 "Entidad ubicacion"

Tabla XIII Usuario

Columna	Tipo	Nulo	Comentarios
id	int(11)	No	
username	varchar(255)	No	
email	varchar(255)	No	
enabled	tinyint(1)	No	
salt	varchar(255)	No	
password	varchar(255)	No	

roles	longtext	No	(DC2Type:array)
documento	varchar(50)	No	
nombres	varchar(150)	No	
apellidos	varchar(150)	No	

Tabla 13 "Entidad usuario"

DESCRIPCIÓN DE LOS APLICATIVOS

El aplicativo web puede acceder a una serie de funcionalidades solo pertenecientes a él, esto debido a que en el dispositivo móvil lo más conveniente es diseñarlo con una estructura fácil y que brinde una respuesta rápida cuando se solicita información, además de que los recursos del hardware son mucho más limitados en el dispositivo móvil. Es sobre este que se implementó el módulo de lectura de etiquetas NFC, y lo primordial es dejar el aplicativo lo más liviano posible dotándolo solo con los requerimientos necesarios para el sistema de información.

Aplicativo Web

Configuración: En esta sección encontraremos todos los datos que son parametrizables en el sistema.

- Categorías: Aquí se puede listar, crear y modificar las categorías de los productos.
Link: <http://rfidinventory.xyz/categoria/list>
- Ubicaciones: Aquí se puede listar, crear y modificar las ubicaciones y estanterías de las bodegas.
Link: <http://rfidinventory.xyz/ubicacion/list>
- Tipo Movimiento: Aquí se puede listar, crear y modificar los tipos de movimientos (Entradas y Salidas del inventario).
Link: <http://rfidinventory.xyz/tipomovimiento/list>
- Productos: Aquí se puede listar, crear y modificar los productos del inventario.
Link: <http://rfidinventory.xyz/producto/list>
- Usuarios: Aquí se pueden gestionar los usuarios de la plataforma y sus respectivos roles.
Link: <http://rfidinventory.xyz/usuario/list>

Movimientos: En esta sección se pueden ver y gestionar todos los movimientos de los productos en la bodega.

- Crear entrada: Ingresar productos al inventario, estos productos solo pueden estar en estado: Creado o fuera del inventario.
Link: <http://rfidinventory.xyz/movimiento/entrada/new>
- Crear Salida: Hacer salidas de productos del inventario, estos productos solo podrán estar en estado: En inventario.
Link: <http://rfidinventory.xyz/movimiento/salida/new>
- Lista de Movimientos: Aquí se pueden listar todos los movimientos realizados.
Link: <http://rfidinventory.xyz/movimiento/list>

Inventario: En esta sección se podrá crear y consultar los productos dentro de la bodega.

- Ingresar Producto: Crear un producto nuevo en el inventario, estos productos no se ven reflejados en la bodega hasta que se le cree un movimiento de tipo entrada, quedará en estado Creado.
Link: <http://rfidinventory.xyz/productounico/new>
- Estado del inventario: Reporte de los productos actuales en la bodega.
Link: <http://rfidinventory.xyz/reportes/estadoinventario>

Reportes: En esta sección se podrá evidenciar las métricas requeridas por el cliente.

- Productos vendidos hoy.
- Total en ventas del día actual.
- Productos vendidos este mes.
- Total en ventas del mes actual.
- Gráfico de ventas de la última semana.
- Total de ventas por categoría en este mes.
Link: <http://rfidinventory.xyz/>

Aplicativo Móvil

Vista inicial: Comprende un módulo de lectura de etiquetas NFC, cuyo principal objetivo es obtener el código contenido en la etiqueta, una vez es capturado se emite una petición a la API con el fin de obtener la información detallada del producto.



Ilustración 8 "Vista inicial móvil"

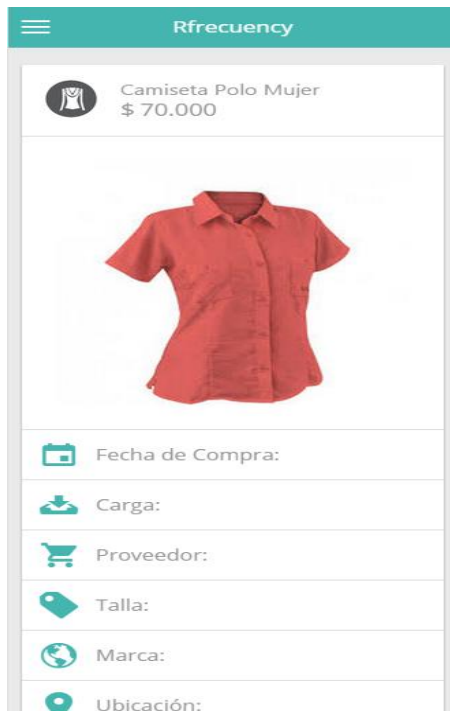


Ilustración 9 "Lectura de tag"

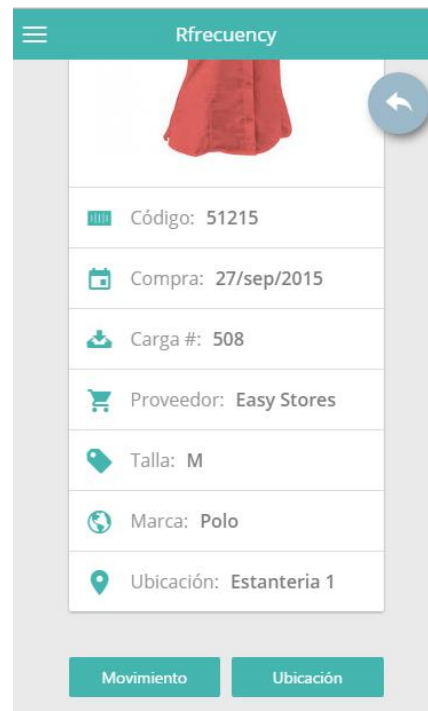


Ilustración 10 "Consulta de producto"

Módulo de creación de productos e ingreso a inventarios:

En este módulo se emite una petición a la API para obtener los productos pendientes por asignación de TAG, para esta lista hay disponibles dos acciones (asignar fotografía, asignar tag); Aparte de esto encontramos la opción de crear producto único la cual le permite al usuario codificar la mercancía mediante el TAG asociando la información suministrada en el formulario.

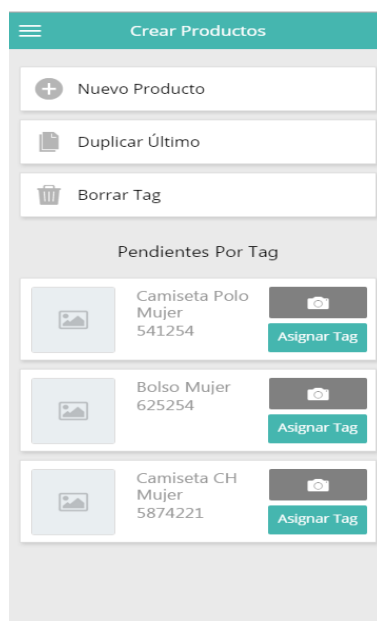


Ilustración 11 "Asignar tag o foto al producto"



Ilustración 12 "Creación de producto"

Módulo de consultas: Herramienta que le permite al usuario encargado realizar consultas de productos ya existentes en el inventario para posteriormente realizar algún tipo de movimiento al producto arrojado en la consulta.

Se puede realizar la consulta parametrizando por:

- Código de producto
- Proveedor
- Carga
- Ubicación

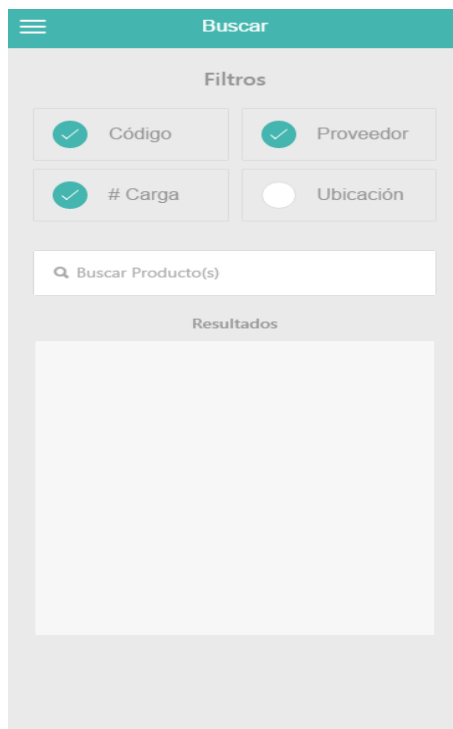


Ilustración 13 "Consulta de productos I"

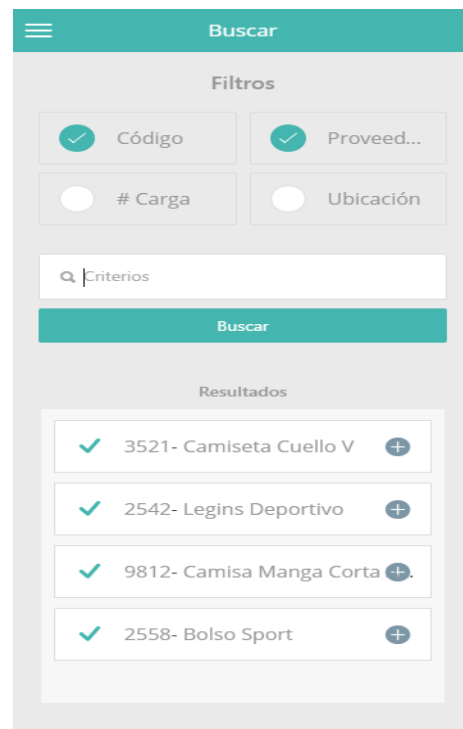


Ilustración 14 "Consulta de productos II"

Módulo de movimientos: Desde esta vista se puede realizar los movimientos a la mercancía existente en el inventario como se lista a continuación:

Movimientos de entrada

- Compra de mercancía
- Devolución de cliente

Movimientos de salida

- Pérdida o robo
- Venta

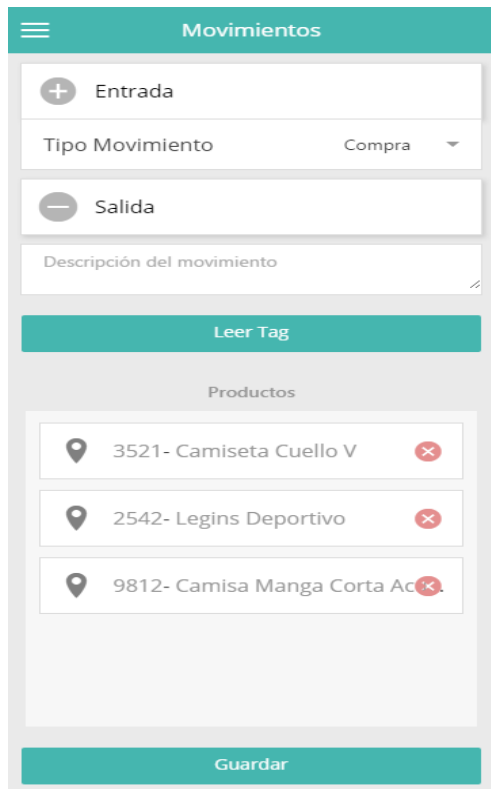


Ilustración 15 "Movimiento a producto I"



Ilustración 16 "Movimiento a producto II"

Módulo de ubicaciones: En esta sección el encargado del inventario podrá ubicar la mercancía en las estanterías o módulos de los cual dispone, a su vez se puede reubicar un lote de productos o un producto a la vez según sea el caso.



Ilustración 17 "Ubicar producto"

Módulo de inventarios: En esta sección se envía una petición a la api la cual verifica si hay algún inventario activo pendiente que haya sido generado desde el panel administrativo web, si este es el caso el usuario administrador deberá activarlo en el móvil ingresando un código de autenticación. Si todo está correcto se puede proceder con el conteo que desde ese momento estará activo.



Ilustración 18 "Validar inventario"

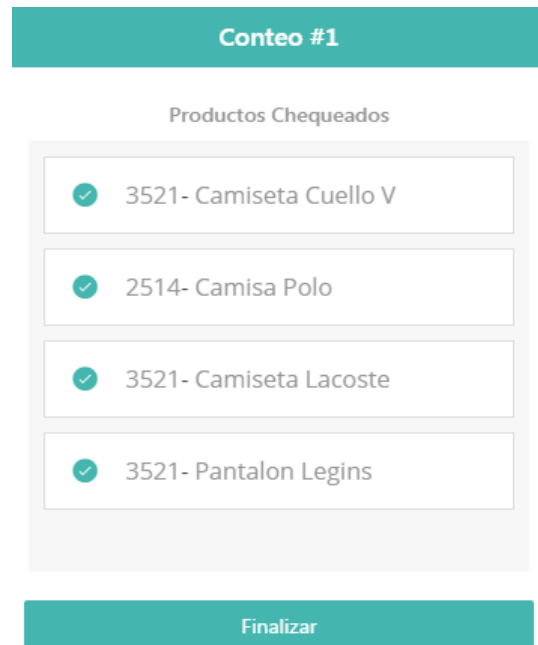


Ilustración 19 "Chequear producto"

PLAN DE PRUEBAS

La etapa de pruebas fue llevada a cabo al finalizar la aplicación. Si una funcionalidad no pasa las pruebas, se revisa su implementación y se hacen los cambios pertinentes hasta que pasa las pruebas.

Durante el desarrollo de la aplicación, se han utilizado datos de prueba, lo que aporta seguridad a la hora de trabajar con esa información. Además, se ha utilizado un servidor online, con datos de prueba, y se ha creado productos y movimientos con datos reales. Una vez se terminó el desarrollo de la aplicación, se utilizaron datos reales para probar si todo funcionaba correctamente.

Pruebas de código unitarias

Las pruebas de código unitarias se realizan sobre funcionalidades concretas de la aplicación para verificar que funcionan correctamente.

Las pruebas unitarias son secciones de código que se ejecutan con la aplicación parada y sin utilizar ninguna base de datos. Por eso, para probar, es necesario que en cada prueba se introduzcan aquellos datos que se van a utilizar en esa prueba.

En la se muestra como se hace pruebas al formulario de categorías, en el cual se le envía en la variable nombre el valor de "Categoría desde Unit Test", y luego se valida que el formulario y cada uno de los campos queden correctos.

El log completo de todas las pruebas unitarias se puede ver en el Anexo 1 "LogsUnitTes", y en el código fuente dentro de la carpeta Test.

```
<?php
namespace AppBundle\Tests\Controller;

use AppBundle\Form\CategoriaType;
use Symfony\Component\Form\Test\TypeTestCase;

class CategoriaTypeTest extends TypeTestCase
{
    public function testSubmitValidData()
    {
        echo 'Validando categoría.';
        $formData = array(
            'nombre' => 'Categoría desde Unit Test'
        );

        $type = new CategoriaType();
        $form = $this->factory->create($type);
        $form->submit($formData);
        $this->assertTrue($form->isValid());
        $view = $form->createView();
        $children = $view->children;

        foreach (array_keys($formData) as $key) {
            $this->assertArrayHasKey($key, $children);
        }
    }
}
```

Ilustración 20 "Unit test Categoría"

```

E:\Proyectos\web\inventory>phpunit -c app/
PHPUnit 3.7.21 by Sebastian Bergmann.

Configuration read from E:\Proyectos\web\inventory\app\phpunit.xml.dist

.Validando categoria.

Time: 1 second, Memory: 4.75Mb

[30;42m[2KOK (1 test, 2 assertions)
[0m[2K
E:\Proyectos\web\inventory>

```

Ilustración 21 "Resultado Unit test"

Pruebas de API para el móvil

Para las pruebas de integración con el aplicativo se utilizó una utilidad del framework llamada *API Documentation*, la cual lista en una pantalla web todos los servicios de API disponibles, con su respectiva descripción, parámetros de entrada, y tipo de método HTTP (GET o POST).

The screenshot shows the 'API documentation' interface. At the top right, there are dropdown menus for 'body format: Form Data' and 'request format: json'. The main content is organized into four sections, each with a title and three action links: 'Show/hide', 'List Operations', and 'Expand Operations'.

- Inventarios**
 - POST /api/guardarconteos.{_format} - Guardar conteo
 - GET /api/inventory/token.{_format} - Retorna los conteos de un inventario por token
- Login**
 - POST /api/logins.{_format} - Loguear un usuario
- Movimientos**
 - GET /api/all/tipomovimiento.{_format} - Retorna todos los tipos de movimientos creados
 - POST /api/crearmovimientos.{_format} - Crear movimiento
- Productos**
 - GET /api/all/products.{_format} - Retorna todos los productos creados
 - POST /api/asignarimagenes.{_format} - Asignar imagen a un producto
 - POST /api/asignartags.{_format} - Asignar tag a un producto
 - POST /api/creates/products.{_format} - Crear Producto Único
 - GET /api/product/info.{_format} - Retorna la información de un producto por tag
 - POST /api/products/filters.{_format} - Retorna los productos que hay en la bodega
 - GET /api/products/sin/tags.{_format} - Retorna todos los productos creados y no tienen Tags

Ilustración 22 "Api doc"

Además de listar, esta aplicación también nos permite hacer pruebas a los servicios, enviando los parámetros de entrada y mostrando en pantalla la respuesta del servicio.

The screenshot shows an API testing tool interface for a service named "Productos". At the top right, there are links for "Show/hide", "List Operations", and "Expand Operations". The main header displays the method "GET" and the endpoint "/api/all/products.{format}", with a button "Retorna todos los productos creados" on the right. Below the header, there are sections for "Input", "Requirements", "Headers", "Content", "Request URL", "Response Headers", and "Response Body".

Input: A "Requirements" section with a field "format" and an equals sign followed by a text input box.

Headers: A section with "Key" and "Value" input fields, an equals sign, and a "New header" button.

Content: A text area with the note "Content set here will override the parameters that do not match the url". Below it, a "Content-Type" field is followed by an equals sign, a "Value" input field, a "Set header" button, and the text "Replaces header if set".

Request URL: A yellow highlighted box containing "GET /api/all/products.json".

Response Headers: A yellow highlighted box containing "200 OK".

Response Body: A yellow highlighted box containing a JSON array of product objects:

```
[
  {
    "id": 1,
    "nombre": "Camisa manga larga",
    "codigo": "4275",
    "categoria": "Camisas"
  },
  {
    "id": 2,
    "nombre": "Camisa manga corta",
    "codigo": "4276",
    "categoria": "Camisas"
  },
  {
    "id": 3,
    "nombre": "Camisa sin cuello",
    "codigo": "4277",
    "categoria": "Camisas"
  },
  {
    "id": 4,
    "nombre": "Camiseta tipo polo",
    "codigo": "4070",
    "categoria": "Camisetas"
  }
]
```

Ilustración 23 "Pruebas Api"

ESTUDIO DE CASO

La empresa Nena Te Viste tiene como principal fuente de ingresos la importación y posteriormente la distribución de prendas de vestir, dicha gestión se realiza en un punto de venta y almacenamiento de mercancía ubicado en la ciudad de Medellín. Debido a su amplio catálogo de productos comprendido entre 1.500 hasta 2.000 prendas de vestir y a sus múltiples referencias provenientes de grandes distribuidores mayoristas, se hizo necesario la codificación de cada producto para de esta manera controlar cada activo.

La empresa necesita tener disponible la información de cada activo en tiempo real y a la mano del vendedor sin que importe en qué lugar de la sala de ventas se encuentre, se acordó con el gerente que para alcanzar este objetivo la mejor alternativa relacionando costo vs beneficio consiste en desarrollar una plataforma móvil que permita utilizar tecnologías inalámbricas de lectura de códigos presentes en la mercancía almacenada u exhibida y que esta a su vez se encuentre integrada con los módulos de servicios disponibles en la web.

Entre las alternativas ofrecidas en el mercado actual encontramos:

Lectores de códigos de barra:

- Permite solo un artículo por lectura
- Necesita contacto visual entre lector y etiqueta
- Corto alcance de lectura de etiquetas
- Debido a que el código utilizado es genérico por categorización de productos no permite localizar la posición de los artículos.
- Los lectores no permiten la lectura de etiquetas o códigos doblados, sucios o con borrones en su impresión.
- Debido a que el código impreso no se puede sobrescribir es más complicado realizar algún cambio lo que puede ocasionar duplicidad de información.

Lectores y etiquetas con tecnología RFID/NFC:

- No es necesario el contacto visual entre lector y etiqueta
- La lectura de etiquetas es mucho más ágil lo que conlleva a realizar los inventarios en una fracción del tiempo acostumbrado
- El alcance de lectura es mucho más amplio
- Se mejora la trazabilidad de los activos ya que se localiza la posición exacta de cada uno de ellos
- La etiqueta se puede reescribir lo que permite, añadir, eliminar Modificar información
- Se incrementa la seguridad de la mercancía ya que cada etiqueta está ligada a un código almacenado en la base de datos de la aplicación, permitiendo codificar los productos con alto nivel de detalle

Teniendo en cuenta la anterior comparativa en las dos tecnologías mencionadas, de la anterior investigación se concluyó que la manera más rápida y eficaz para realizar inventarios es integrando en su sistema la tecnología de identificación por radiofrecuencia o RFID, que permite una fácil lectura y almacenamiento de los datos sin necesidad de que haya contacto directo con las etiquetas o códigos.

Para la gestión administrativa es necesario que el sistema cuente con un módulo de inventarios que permita corroborar las existencias en bodega con la frecuencia que la empresa requiera, para los productos que se reporten como faltantes en el inventario se debe permitir consultar su historial de movimientos y su ubicación actual; el lector de radiofrecuencia (NFC), presente en el dispositivo móvil permite realizar las tarea de conteo con una agilidad superior en contraste con los antiguos lectores de código de barras.

Cada 3 meses se lleva a cabo la importación de nueva mercancía proveniente de Estados Unidos con un promedio de 1.200 prendas por cargamento, actualmente el proceso desarrollado en la gestión de estos productos se lista a continuación:

1. Importación de mercancía
2. Recepción de la mercancía
3. Etiquetación y codificación
4. Ingreso de la mercancía en archivo contable formato Excel

Datos requeridos:

- Fecha de compra
- Almacén / Proveedor
- Marca
- Tallas
- Código
- Artículos
- Cantidad
- Tasa de impuesto
- Valor de compra
- Valor de compra unitario
- Valor de compra sin impuestos
- Valor en dólares
- Precio de venta

5. Ubicación y/o almacenaje
6. Venta o salida de mercancía

Los datos anteriores se deben diligenciar para cada uno de los productos de la carga que se codifica en el formato en Excel que podemos observar en el Anexo 2 "Formato de Registro de Mercancía 08-2015", este proceso se vuelve engorroso debido a la gran cantidad de registros.

PLANO DE INSTALACIONES DE LA EMPRESA NENA TE VISTE

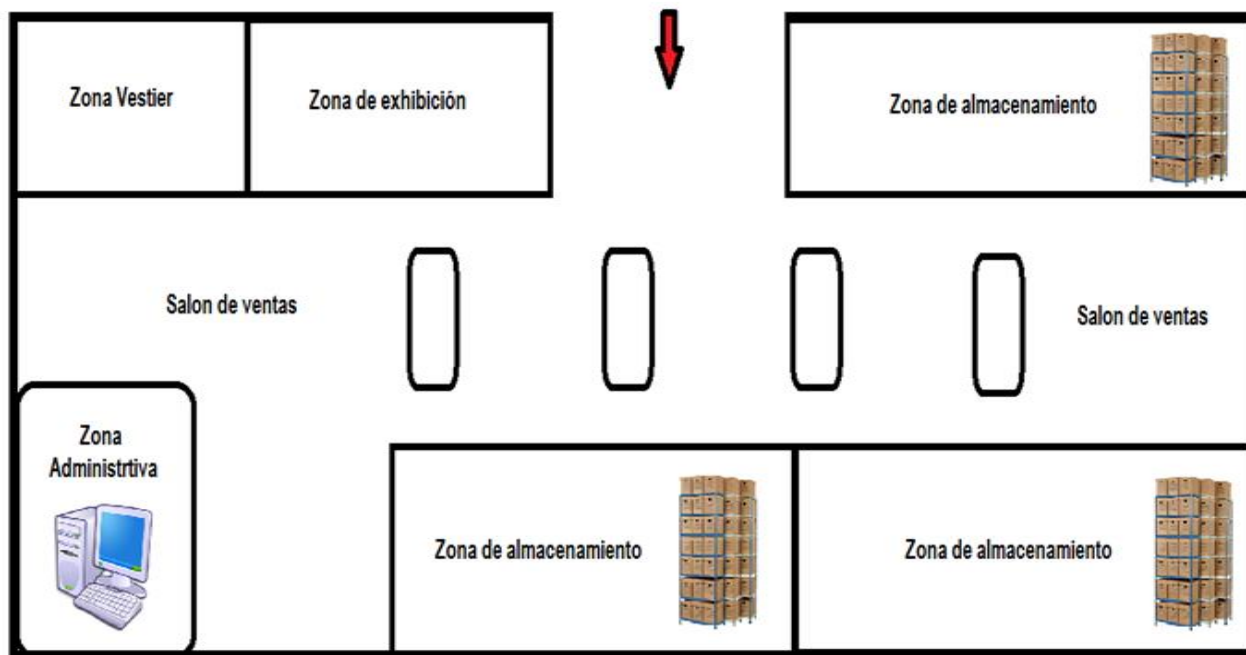


Ilustración 24 "Plano de las instalaciones"

Comparativa de la plataforma de gestión administrativa implementada, con el sistema anterior basado en Excel.

Codificación de mercancía:

Antes: Se realizaba manualmente colocando una etiqueta con el código del producto a cada artículo, posteriormente se diligenciaba el archivo en Excel conteniendo la información más relevante de este mismo.

Ahora: Se le asigna a cada artículo una etiqueta NFC, luego desde el administrador web se puede crear el producto unitario para posteriormente asignar un número en el tag que lo identifica, esta tarea es principalmente pensada para la aplicación móvil ya que esta es la única con la capacidad de generar un tag y escribirlo en la etiqueta mediante el lector.

Este proceso se agilizo significativamente hasta el punto de disminuir en un 50 % los tiempos de registro y codificación de la mercancía.

Consulta de artículos en el inventario:

Antes: Se debía buscar por categoría en el archivo de Excel teniendo en cuenta que los resultados arrojados no diferenciaban entre sí en cuanto al detalle de información de cada producto.

Ahora: Se permite consultar en tiempo real desde la aplicación web o móvil algún producto que se encuentre en la bodega y desde cualquier parte de las locaciones, lo que permitió mejorar la atención para los clientes en el punto de atención e incrementar las ventas.

Movimientos de entrada y salida de mercancía desde bodega:

Antes: Se debía buscar el artículo en el archivo de Excel y decrementar la cantidad presente en las existencias de ese producto en particular.

Ahora: Se debe consultar el producto en cualquiera de las dos aplicaciones (web o móvil) y luego aplicarle el movimiento requerido.

Módulo de inventarios a bodega:

Antes: Se procedía a contar uno por uno cada producto, para luego tratar de verificar y encontrar los productos que no fueron detectados en el conteo. En este proceso era común saltarse productos al momento de contar y por lo general se precisaban hasta 3 conteos para corroborar la información recabada en cada uno de ellos.

Ahora: Por medio de la aplicación móvil con solo acercar el dispositivo cerca de las etiquetas de radiofrecuencia el artículo figura como contado apenas es detectado por el lector, esta implementación disminuyó hasta en un 80% los tiempos necesarios para concluir un inventario así como el número de conteos a realizar, también se aumentó hasta en un 99% la precisión al contar la mercancía al eliminar el factor humano reemplazándolo por el sistema de conteos de la aplicación.

Módulo de ubicación y reubicación de mercancía en bodega:

Antes: La disposición de la mercancía en la bodega no seguía ningún patrón establecido, ni se documentaba en que parte se encontraba almacenado o exhibido algún producto en particular.

Ahora: Se permite la asignación de ubicaciones a uno o más productos en los sitios de almacenaje o exhibición, lo que permite hacer un seguimiento para cada activo en particular, esto contribuyó a un mejor aprovechamiento de los espacios y a tener un control más preciso sobre toda la mercancía en general.

Módulo de usuarios y roles:

Antes: Se desconocía que persona realizaba una venta o algún otro movimiento.

Ahora: Cada que un movimiento es realizado se registra el usuario que lo llevo a cabo, esta función permitirá en un futuro generar comisiones a los mejores vendedores de la empresa, además es un gran plus en cuanto a la seguridad de la mercancía haciendo responsable a la persona quien realiza el movimiento.

BIBLIOGRAFIA

1. Wikimedia Foundation, Inc
Link: <http://es.wikipedia.org/wiki/Empresa>
Fecha consulta: Octubre – 2015
2. Scrum Alliance
Link: <https://www.scrumalliance.org/>
Fecha consulta: Septiembre - 2015
3. Wikimedia Foundation, Inc
Link: <http://es.wikipedia.org/wiki/Documento>
Fecha consulta: Noviembre - 2015
4. Wikimedia Foundation, Inc
Link: http://es.wikipedia.org/wiki/Gesti%C3%B3n_documental
Fecha consulta: Noviembre – 2015
5. Wikimedia Foundation, Inc
Link: https://es.wikipedia.org/wiki/An%C3%A1lisis_y_dise%C3%B1o_orientado_a_objetos
Fecha consulta: Noviembre - 2015
6. Wikimedia Foundation, Inc
Link: http://es.wikipedia.org/wiki/Modelo_Vista_Controlador
Fecha consulta: Septiembre - 2015
7. Wikimedia Foundation, Inc
Link: http://es.wikipedia.org/wiki/Base_de_datos
Fecha consulta: Septiembre - 2015
8. Wikimedia Foundation, Inc
Link: http://es.wikipedia.org/wiki/Modelo_relacional
Fecha consulta: Septiembre - 2015
9. W3schools
Link: <http://www.w3schools.com/js/>
Fecha consulta: Agosto - 2015
10. Libros Web
Link: http://librosweb.es/libro/symfony_1_2/capitulo_2/el_patron_mvc.html
Fecha consulta: Agosto – 2015
11. Libros Web

Link: <http://ionicframework.com/docs/concepts/structure.html>

Fecha consulta: Agosto – 2015

12. Scrum

Link: <http://proyectosagiles.org/que-es-scrum/>

Fecha consulta: Noviembre – 2015

13. Inventario

Link: <https://es.wikipedia.org/wiki/Inventario>

Fecha consulta: Noviembre – 2015

14. Arquitectura de software

Link: http://sg.com.mx/revista/27/arquitectura-software#.Vlo8B_kvfiU

Fecha consulta: Noviembre – 2015

15. Retail

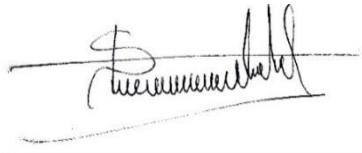
Link: <https://es.wikipedia.org/wiki/Detal>

Fecha consulta: Noviembre – 2015

16. Patrón MVC

Link: <https://es.wikipedia.org/wiki/Modelo%E2%80%93vista%E2%80%93controlador>

Fecha consulta: Noviembre – 2015



Firma:

Nombre: Sebastian Palacio Florez
C.C. N° 1037603189 de Envigado



Firma:

Nombre: Juan Esteban Ríos
C.C. N° 1036636992 de Itagüí



Firma:

Nombre: Santiago Velásquez Montoya
C.C. N° 1037621474 de Envigado