

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

COLEGIO CIENCIAS E INGENIERIA

Sistema de Ventas Cruzadas en Alimec
Sistematización de Experiencias prácticas de investigación y/o intervención.

Stephan Zurita Serafim

Ingeniería en Sistemas

Trabajo de titulación presentado como requisito
para la obtención del título de
Ingeniería en Sistemas

Quito, 10 de mayo de 2018

**UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ
COLEGIO CIENCIAS E INGENIERIA**

**HOJA DE CALIFICACIÓN
DE TRABAJO DE TITULACIÓN**

Sistema de Ventas Cruzadas en Alimec

Stephan Zurita Serafim

Calificación:

Nombre del profesor, Título académico

Fausto Pasmay, Master of Sciences

Firma del profesor:

Quito, 10 de mayo de 2018

Derechos de Autor

Por medio del presente documento certifico que he leído todas las Políticas y Manuales de la Universidad San Francisco de Quito USFQ, incluyendo la Política de Propiedad Intelectual USFQ, y estoy de acuerdo con su contenido, por lo que los derechos de propiedad intelectual del presente trabajo quedan sujetos a lo dispuesto en esas Políticas.

Asimismo, autorizo a la USFQ para que realice la digitalización y publicación de este trabajo en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Firma del estudiante:

Nombres y apellidos:

Stephan Zurita Serafim

Código:

00116911

Cédula de Identidad:

1715334692

Lugar y fecha:

Quito, 10 de mayo de 2018

RESUMEN

El proyecto propuesto es la obtención de pronósticos de productos de venta cruzada, sugeridos de acuerdo con las tendencias de ventas que presenta la compañía Alimec S.A. El problema presentado por la empresa consistía en que los vendedores nuevos o inexpertos tenían dificultades en ampliar la lista de productos adquiridos por cada tienda. Para solucionarlo se decidió entregar a los vendedores una lista de productos sugeridos preestablecida de acuerdo a la tendencia de los clientes. Se desarrolló un sistema que realiza los pronósticos de venta cruzada acorde con el historial de ventas, sobre la base de un modelo de regresiones logísticas desde una tabla de producto por cliente. La selección de productos, para este modelo, se gestiona a través de una correlación a partir de la tabla propuesta, de donde se obtiene la probabilidad de adquisición de producto por cliente. La solución implementada, que tiene un enfoque estadístico, predice la relación de que un evento suceda en correspondencia a las variables que están relacionadas al mismo evento. Estas regresiones entregan una probabilidad de que el evento ocurra; el cual, puede ser interpretado como una probabilidad de que el cliente desee obtener el producto sugerido.

Los resultados del modelo estadístico tuvieron un error del 8.7%, transformado todas sus probabilidades que estuvieran por encima del 50% como probabilidad de éxito. Con esto se demuestra que el modelo creado cumplía con los requerimientos de error mínimo para poder ser usado en el proyecto. Los errores de falsos positivos con probabilidad superior al 70% se usan para crear una lista de productos sugeridos que van a ser entregados a los vendedores.

El producto final que se entregó a Alimec consta de tres módulos: el programa de análisis estadístico, una aplicación móvil para Android y servicios REST. El programa de análisis estadístico tiene la funcionalidad de procesamiento de información en base al modelo planteado. La aplicación móvil para Android permite iniciar sesión y consultar los artículos sugeridos por cliente. Los servicios REST envían la información de vendedores, clientes, productos y sugeridos a la aplicación móvil de manera cifrada y segura, de tal forma que el aplicativo móvil crea una base de datos de la información y establece la estructura de base de datos.

Palabras clave: sistema, móvil, venta, cruzada, estadística, REST, servicios.

ABSTRACT

The proposed project is to achieve a cross sale products forecast, which are suggested according to sales' tendencies presented by the company Alimec S.A. The problem presented by the company was that the new or the inexperienced salesmen had difficulties increasing the list of products purchased by each store. It was decided to solve the problem by providing the salesmen with a list of suggested products which were predefined according to customer's tendencies. Therefore, a system was developed to forecast cross sales according to the sales history, based on a model of logistic regressions from one product table per customer. The selection of products, for this model, is managed through a correlation of data from the proposed table, from which the probability of product acquisition per customer is obtained. The implemented solution, with a statistical focus, predicts the relationship of an event happening in correspondence to the variables that are related to the same event. These regressions provide a probability that the event will occur; which, can be interpreted as a probability that the client wishes to obtain the suggested product.

The results of the statistical model had an error of 8.7%, due to the conversion of all its probabilities that were above 50% as the probability of success. This shows that the model created met the minimum error requirements to be used in the project. False-positive errors with a probability greater than 70% are used to create a list of suggested products that will be delivered to salesmen.

The final product delivered to Alimec contains three modules: the statistical analysis program, the application for Android and the REST services. The statistical analysis program has the functionality of information processing based on the proposed model. The Android application allows the salesmen to log in and look up the articles suggested per client. The REST services are used to send the information regarding salesmen, customers, products and suggested products to the mobile application in an encrypted and secure way, so that the mobile application creates a database of information and establishes the database structure.

Keywords: system, mobile, sales, cross, statistics, REST, services.

TABLA DE CONTENIDO

INTRODUCCIÓN	9
DESARROLLO DEL TEMA.....	11
Base conceptual	11
<i>Venta cruzada.....</i>	<i>11</i>
<i>Minería de Datos.....</i>	<i>11</i>
<i>Regresiones logísticas.....</i>	<i>12</i>
<i>Requerimientos establecidos.....</i>	<i>13</i>
<i>Alcances establecidos.....</i>	<i>15</i>
Análisis de factibilidad técnica.....	15
Diseño del sistema	18
<i>Software de análisis estadístico.....</i>	<i>19</i>
<i>Servicios REST.....</i>	<i>20</i>
<i>Aplicación de Android.....</i>	<i>21</i>
CONCLUSIONES	33
REFERENCIAS BIBLIOGRÁFICAS.....	35
ANEXO A: ACTA DE ENTREGA RECEPCION DEL SISTEMA.....	37

ÍNDICE DE TABLAS

TABLA 1: TIEMPOS DE SELECCIÓN DE VARIABLES ACORDE A SU ALGORITMO.	29
TABLA 2: ERRORES DE LOS MODELOS SELECCIONADOS.	31

ÍNDICE DE FIGURAS

FIGURA 1: CASOS DE USO	19
FIGURA 2: ESTRUCTURA DE LA BASE DE DATOS.	22
FIGURA 3: DIAGRAMA DE ACTIVIDAD DE DESCARGA DE INFORMACIÓN.....	23
FIGURA 4: DIAGRAMA DE ACTIVIDAD DE INICIO DE SESIÓN.....	24
FIGURA 5: DIAGRAMA DE ACTIVIDAD DE DESCARGA DE SUGERIDOS.....	24
FIGURA 6: DIAGRAMA DE ACTIVIDAD DE CERRAR SESIÓN.....	25

INTRODUCCIÓN

Uno de los grandes problemas que tienen las empresas de venta y distribución de productos perecibles es predecir lo que el cliente pueda necesitar. De acuerdo con Villa-López, Kuster-Boluda y Escamilla-Santana (2015) los vendedores son quienes realizan las predicciones, sin embargo, solo los mejores venden productos vinculados en las cantidades necesarias para que no se caduquen, a este tipo de venta se conoce como venta cruzada (Shemitz, Lee y Lilien, 2014). Para los autores, en un mundo ideal se puede conseguir personal capacitado que tomen estas decisiones y así encontrar la canasta perfecta para cada tienda, que a su vez depende del tamaño, localización, intereses, entre otros factores.

Por lo general, los vendedores no se encuentran capacitados para esta tarea, sobrevenden los productos o venden a las tiendas más de lo que necesitan y peor aún, no logran encontrar relación entre los mismos productos con otros, que termina en problemas entre empresa y proveedor. La ineficiencia e inexperiencia de los vendedores hacen daño a la compañía y evitan que se logre realizar de manera óptima las ventas.

Alimec, empresa de distribución y venta de productos perecibles, implementó un sistema para el control de pedidos para constatar la eficiencia de sus vendedores. Se detectaron varios problemas: el primero tiene que ver con la falta de conocimiento de los nuevos vendedores con las conexiones de productos, solo algunos vendedores antiguos conocen sobre estas concordancias. Otro problema suscita cuando se efectúa una sobreventa de productos, problema que no afecta directamente a los vendedores, sino que se convierte en un inconveniente por el modelo de negocio de la empresa, el cual permite hacer devoluciones de productos caducados por uno nuevo sin costo; por ende, la sobreventa de los productos genera más devoluciones, causando así costos innecesarios a la empresa. En resumen, la empresa necesita que sus vendedores sepan qué y cuánto vender para evitar costos innecesarios.

Otro problema está relacionado con el área de sistemas, ya que su sistema es antiguo administra la contabilidad, las devoluciones, los pedidos, registra los clientes, los vendedores y lleva una base de datos cerrada. El acceso a su sistema es complejo, los recursos provistos para la implementación de cualquier sistema adyacente son minúsculos, lo que disminuye la capacidad de procesamiento y memoria. Las ventanas que se encuentran abiertas para este sistema son la lectura de archivos en formato CSV para llenar plantillas prefabricadas del sistema y la obtención de datos a través de CSVs.

Esto motivo a realizar una propuesta de creación de un sistema de venta cruzada, que es segura para los vendedores, y una táctica mediante la cual el vendedor propone productos complementarios a los que ya se están comprando, esta predicción se la obtiene analizando diversos datos de los clientes, vinculando esta información entre todos los clientes actuales para así encontrar un modelo de predicción. Este sistema consta de tres componentes. El primero es una aplicación en Android donde los vendedores puedan seleccionar un cliente y ver qué productos y en qué cantidad debe ofrecer. La segunda parte son los servicios REST necesarios para realizar todas las interacciones con el servidor de acuerdo a sus requerimientos. Y por último, el programa que genera las sugerencias de productos con la información provista de su sistema principal.

DESARROLLO DEL TEMA

Base conceptual

Venta cruzada

La venta cruzada es la extensión de servicios en diferentes áreas para cubrir de manera integral la necesidad de los clientes (Harding, 2002). El autor se refería a este modelo como el mejor modelo (*BEST model*), ya que hacía referencia al acrónimo de las letras: *Buyers* (compradores), *Events* (eventos), *Signals* (señales) y *Techniques* (técnicas). Los compradores tienen flujos de información imperfectos, si un vendedor maneja la información de manera correcta, genera confianza y oportunidades que brindan un valor significativo para los clientes. Los eventos, desde una visión amplia, pueden brindar la oportunidad de comprender las inquietudes y desafíos de los clientes. Las señales se refieren al nivel de captación del vendedor con las necesidades del cliente. Finalmente, las técnicas con los componentes tácticos que hace que una venta cruzada suceda.

En el caso de este proyecto, evita que el comprador se equivoque con las necesidades del cliente, a través de un aplicativo que automatiza los flujos y eventos a través de la técnica, que sería el aplicativo de venta cruzada.

Minería de Datos

La Minería de Datos “se podría abstraer como la construcción de un modelo que ajustado a unos datos proporciona un conocimiento” (Requelme, Ruiz y Gilbert, 2006, p. 15). En esta tarea se elige el modelo y se ajusta los datos, por consiguiente es la etapa primordial de un proceso. Se conforma de las siguientes funciones: “clasificación, regresión, clustering, resumen, recuperación de imágenes, extracción de reglas” (p.13), entre otras.

En minería de datos existen reglas de asociación, las cuales producen subconjuntos de ítems que se encuentran relacionados por un factor en común. Un ejemplo donde se aplican estas reglas se puede observar en la canasta de mercado, “donde el objetivo es encontrar regularidades en los comportamientos de los clientes dentro de los términos de combinación” (García y Acevedo, 2010, p.57).

La canasta del mercado puede ser interpretada como una matriz booleana de productos por cliente, donde 0 es que no compro y 1 es que si compro. Debido a que los clientes siempre van a tener una canasta de ítems en la tienda se puede suponer que todo lo que han comprado y vendido siempre va a estar presente en la tienda. Para poder realizar predicciones de venta cruzada se tiene que tomar en cuenta que es lo otros clientes compran en conjunto con un ítem específico; de esta manera, se tiene que relacionar un ítem con sus subalternos que se compran junto con este mismo para poder conocer su vinculación con los otros (Wong, Fu y Wang, 2005).

Esta técnica ayudara a generar variables con las cuales los modelos estadísticos van a entrenarse y bajo cuales van a poder predecir. Es importante tomar en cuenta que no todas las variables obtenidas van a ser útiles al modelo. Según Derby (2017), los modelos de estadísticos no van a poder discernir si una variable predice un dato o no, y recae dentro de la persona armando el modelo de entregar las variables relevantes.

Regresiones logísticas

Las regresiones son un análisis multivariante que permite introducir variables predictoras de la respuesta que, a partir de los coeficientes de regresión introducidas en el modelo, se obtiene el resultado para ese determinado patrón (Aguayo, 2012).

El modelo estadístico a usar es el logístico, el cual según Derby (2017) los mejores modelos predictivos para pronosticar si un evento sucederá o no, estos son modelos en los cuales se entrega una probabilidad de que un evento suceda con las variables establecidas. Para

entender cómo predice un modelo de logístico primero se tiene que entender la matemática usada para su funcionamiento. Los modelos de regresión logística usan un logaritmo natural sobre una probabilidad de que un evento pase, como es un logaritmo natural todo el problema puede ser expresado como e exponencial a las variables independientes sobre 1 menos e exponencial sobre las mismas variables y pesos, lo cual entrega una probabilidad entre 1 y 0 (Peng, Lee e Ingersoll, 2002).

Obtención de los requerimientos

Para realizar cualquier sistema en una empresa es necesario establecer los requerimientos y alcances del proyecto. Tras conocer los antecedentes y realizar el planteamiento de la idea se procedió efectuar un levantamiento junto con la gerencia de Alimec. Tras varias reuniones con el cuerpo gerencial se obtuvieron los requerimientos y se plantearon los alcances del sistema de venta cruzada. Cabe recalcar que los requerimientos definen lo que la empresa pide que el sistema realice; mientras que, los alcances estipulan los límites del desarrollo del proyecto para evitar peticiones extras de la empresa.

Requerimientos establecidos.

1. El aplicativo de móvil debe constar de un inicio de sesión.

El inicio de sesión es una condición para la protección y confidencialidad de la información. Evita que caiga en manos equivocadas, o que un vendedor desvinculado transfiera la información sobre las necesidades de sus clientes, que se encuentran en el aplicativo.

2. El aplicativo debe conectarse al servidor automáticamente, solo la primera vez que descarga el contenido.

Esta condición se aplica porque no existe una buena conexión a redes móviles, de acuerdo a la gerencia de la empresa. También, la restricción del uso del servidor, debido a que no se encuentra diseñado para el manejo de varias conexiones continuas, de manera que

prefieren que todos los procesos se realicen en el celular. Otro requerimiento importante fue el uso de datos móviles, mismo que debe ser mínimo. Estas razones, determina que los datos se descarguen automáticamente la primera vez o cuando se presione un botón de actualizar en la pantalla de inicio de sesión.

3. El aplicativo debe poder correr en Android versión 23

Alimec entrega a sus vendedores celulares Huawei que corren Android versión 23, por lo que se requiere que el aplicativo a desarrollar funcione en dichos dispositivos.

4. El aplicativo debe entregar un máximo de 5 productos sugeridos por cliente.

Este requerimiento fue planteado debido al problema de la calidad de los vendedores, de manera que estos tengan una limitación al momento de elegir, si aparecieran más sugerencias puede que los vendedores no encuentren a estas opciones como posibilidades reales.

5. El aplicativo debe predecir una cantidad sugerida por producto para cada cliente.

En la validación del aplicativo, la gerencia solicitó que se incluya la cantidad de producto, ya que a los vendedores no les sirve saber que vender si no saben cuánto vender. Para este evento, solicitaron que sea un promedio entre cantidades compradas y aparezca una cantidad sugerida.

6. El aplicativo debe poder seleccionar cliente y local antes de mostrar las sugerencias.

Este requerimiento solicita una lista de selección de clientes de los cuales el vendedor pueda elegir. Tras la primera elección, debe ubicarse los locales correspondientes al cliente, ya que cada local debe de tener su propia lista de productos sugeridos.

7. Toda la información enviada al aplicativo debe estar encriptada.

Por seguridad para que ninguna parte de la información pueda ser interceptada o accedida por un tercero, la gerencia solicitó que se encripte toda la información para que

solamente aquellos con un usuario y contraseña puedan acceder, además que la información solo esté disponible dentro de la aplicación.

Alcances establecidos.

1. El aplicativo funcionara exclusivamente en Android.

Se estableció que todo el desarrollo va a ser dedicado para una aplicación únicamente para Android.

2. Toda la información necesaria es entregada por su departamento de sistemas.

Debido a que no se puede hacer conexiones directas a sus bases de datos, se estipulo que la información a utilizarse, tanto para el inicio de sesión como para el análisis de venta cruzada, serán provistos por el área de sistemas y es responsabilidad de Alimec que estos datos estén disponibles.

3. La seguridad únicamente cubrirá los datos visibles públicamente.

Se dispuso que la seguridad a ser usada estará únicamente integrada a datos que se muestran de manera pública; es decir, datos que se pueden ver con el servicio REST que envía la información por http. Cualquier otra seguridad como la del servidor o la de las conexiones con el mismo recae en el área de sistemas de Alimec, el cual se comprometió a mantener el servidor seguro y con conexiones de https.

Análisis de factibilidad técnica

Para la factibilidad técnica se analizó los implementos provistos para el proyecto, tanto de software como de hardware, así se determinó la factibilidad de llevar a cabo el proyecto y se determinaron los limitantes del mismo.

En hardware la empresa posee como servidor una máquina virtual corriendo Windows server 2016 con 4 GBs de memoria RAM y se utilizan como periféricos los teléfonos celulares Huawei Y5 para los vendedores.

En software se instauro un servidor XAMPP bajo un IP público estático con Apache y PHP habilitados. Los dispositivos Android corrían la versión 23.

La información requerida para el diseño de la aplicación y su correcto funcionamiento fue entregada en varios archivos CSV, mismos que mostraban datos tanto para el inicio de sesión como para el análisis. Cuando entregaron estos datos dejaron en claro que algunas columnas no tenían un valor valido ya que no todos los datos eran confiables debido a problemas en la integridad de los mismos.

Los archivos de texto plano que la empresa entregó constaban de lo siguiente:

1. Clientes

El archivo de clientes constaba de 14 columnas las cuales contenían: el id interno del cliente, la razón social, el nombre comercial, el id del local, la dirección del cliente, la categoría de precios, la ruta, el vendedor, el canal, el sub-canal, el teléfono, la ciudad y el email. De todos los datos provistos los únicos que eran utilizables fue: el id, el nombre comercial, el local, la dirección y la lista de precios. El resto de datos se encontraban en dos secciones, o no eran confiables como el vendedor y la ruta, o eran irrelevantes para el sistema como el mail y el teléfono.

2. Vendedores

El archivo de vendedores fue entregado con cuatro columnas: el id, el nombre, la ciudad y la clave para la aplicación. Se usaron todos los datos con excepción de la ciudad para poder hacer el inicio de sesión.

3. Productos

Para los productos se entregaron cinco columnas en los archivos: el id, la descripción, las unidades de medida, un booleano para definir si paga IVA y un código de importancia. No se usó la unidad de medida ya que no existía relación con el sistema y el código de importancia si se encontraba vacío no se tenía que tomar en cuenta el producto.

4. Precios

La lista de precios ayuda tanto para el análisis como para el filtro en los clientes. Este archivo consta de tres columnas: el id del producto, una letra de categoría de precio y el precio. Todo lo del archivo fue usado y la categoría de precios establecía que productos podían ser comprados por cada cliente.

5. Pedidos

El archivo de pedidos fueron los datos requeridos para poder realizar el análisis estadístico, este contaba de cinco columnas: el código del cliente, el código del local, el código del producto, un promedio de la cantidad vendida por pedido y la cantidad total vendida a lo largo de la existencia del cliente. El archivo fue pedido con estos datos para poder realizar una matriz de clientes por productos y hacer el análisis estadístico, además de poder obtener una media de la cantidad de producto pedido para poder calcular la cantidad sugerida.

Una vez se obtuvo toda la información esencial, tanto de límites de hardware, software e información, se prosiguió a revisar la factibilidad del proyecto. La necesidad de que el procesamiento se realice en los celulares y no en el servidor era evidente debido a la poca capacidad que fue entregada; por lo que, se podría usar al servidor únicamente para el procesamiento básico y requerido. Los celulares por otra parte constaban de 4 GBs en RAM y 16 GBs en memoria interna. Los archivos entregados en conjunto tenían un tamaño de 8.91 MBs; siendo el archivo más pesado el de pedidos, pesando 7.26 MBs, el cual no se debía enviar a los celulares.

Para la parte del inicio de sesión y navegación dentro del aplicativo se analizó los datos que se tenían en el archivo de vendedores, precios, productos y clientes. Debido a que no se deseaba que la aplicación tenga constante conexión con el servidor, se decidió mandar de manera segura los archivos para que, al interior de la aplicación los carguen en una base de datos que replique la estructura y conexiones de los archivos.

Para crear los modelos estadísticos y analizar los datos provistos en el archivo de pedidos, se decidió que estos procesos sucedieran en una aplicación aparte; la cual, podría ser implementada en servidor y usada cuando nadie este activo en la aplicación. El análisis tendría que crear un archivo de sugerencias que el aplicativo se lo descarga con los servicios REST, así el servidor no tiene que estar generando estadísticas constantemente. En el caso de que el servidor no pueda procesar las estadísticas, por limitaciones de memoria o capacidad de procesamiento, el programa que genera el archivo de sugerencias podría ser ejecutado en una computadora a parte que tenga una copia de los archivos provistos por la empresa y el archivo de sugerencias obtenido solo se tendría que dejar en el servidor.

Diseño del sistema

Una vez conocida las posibilidades y limitaciones que se encontraban presentes en Alimec, se prosiguió a armar un diseño del sistema. El sistema requería de tres módulos, la aplicación, los servicios REST y el análisis estadístico.

Primero se crearon los casos de uso: el inicio de sesión, el cerrar la sesión, la generación de los sugeridos, la descarga de los archivos necesarios, la consulta de los sugeridos y el envío de los archivos necesarios. El envío de los archivos es un caso de uso que no requiere de un agente externo para poder ser llevado a cabo, por lo que no se le tomara en cuenta para análisis posteriores.

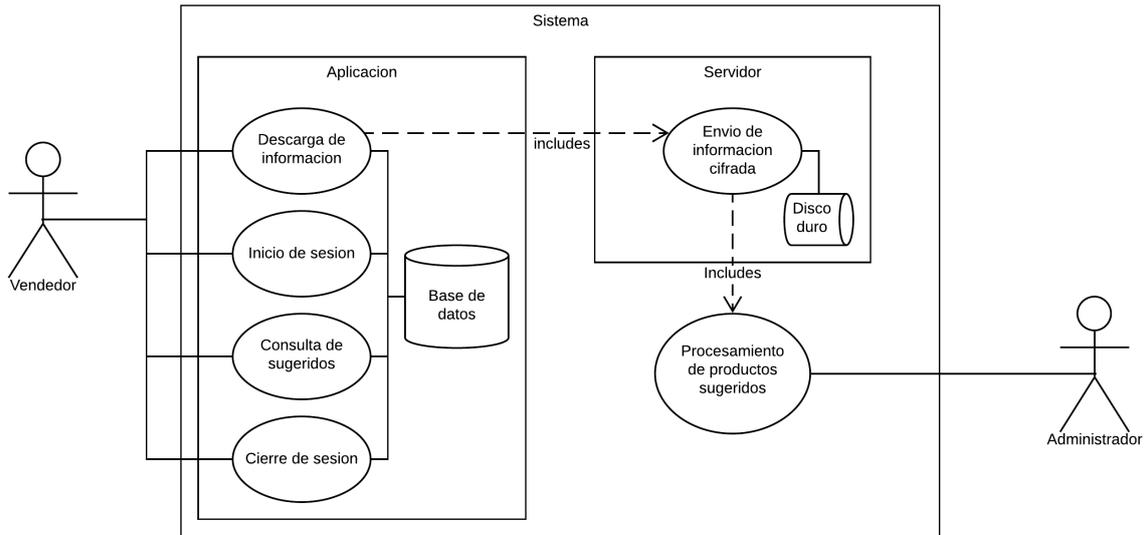


Figura 1: Casos de Uso

Una vez realizados los casos de uso se puede diferenciar dos tipos de usuarios en el sistema: el administrador de sistemas, que es la persona encargada de subir los nuevos archivos y generar los sugeridos; y el vendedor, que es el que va a interactuar directamente con la aplicación. Con los usuarios de la aplicación definidos y los casos establecidos se pudo otorgar las funcionalidades reales a cada módulo del sistema acorde con su interacción con los usuarios.

Software de análisis estadístico.

El análisis estadístico recae enteramente en el administrador del sistema y nunca va a tener interacción directa con el resto de módulos. El administrador tiene la responsabilidad de poner en el directorio el archivo generado, ya que es donde los servicios REST van a obtener los datos. Debido a que no está intrínsecamente vinculado al resto del sistema se diseñó al programa como un script que se alimenta de los archivos entregados y genera el archivo de sugerencias siempre que se lo corra. Debido a que las sugerencias solo se actualizan cuando se

genere un nuevo archivo de pedidos, que puede ser accedido a cualquier momento que el administrador lo desee, el script se entregara para ser corrido en cualquier computador.

El lenguaje en el que se decidió hacer el análisis estadístico fue R, por la facilidad que brinda al momento de crear modelos, regresiones y cálculos de manera rápida y eficaz. Con una instalación sencilla que contiene en su paquete básico todos los implementos necesarios para poder correr el script. R puede ser instalado en Windows server 2016, según su documentación. El archivo generado por el script tendría un formato CSV parecido al resto de archivos con cuatro columnas para vincular los clientes con los productos. Las columnas presentes serian: el id del producto, el id del cliente, el id del local y la cantidad sugerida para la venta.

Servicios REST.

Los servicios REST son los datos que se obtuvo del servidor, su función básica es cifrar y enviar los archivos a la aplicación para que maneje la información. Se decidió usar un algoritmo de clave privada, ya que requiere menos tiempo para cifrar y menos recursos para realizar el cifrado. La clave privada la tendría dentro el aplicativo y los servicios REST por lo que solo el aplicativo podría descifrar los mensajes enviados por los servicios sin tener que proceder a realizar varios cálculos ni establecer una sesión.

Ya que los servicios leen archivos presentes en el servidor, los cifran y los envían, no es necesario crear más de un servicio que reciba como parámetros el nombre del archivo a obtener y el número de línea en la que se desea empezar, este último parámetro seria usado en la eventualidad que no se logre enviar todas las líneas del archivo.

En función del servidor XAMPP se tiene que hacer el desarrollo en PHP, ya que es el lenguaje instalado. Se usará el cifrado AES256 con un IV aleatorio por cada conexión en modo CFB8, para así poder tener concordancia con la parte Java del aplicativo y para seguir los estándares de seguridad en claves privadas. La llave y el directorio donde se encuentran los

archivos se encontrarán en un registro de configuración que el PHP leerá para poder operar. Tanto los archivos como la configuración, no podrán ser accedidos por la red.

En resumen, se va a tener un servicio REST desarrollado en PHP que obtiene de argumentos en POST, el nombre del archivo y el número de línea en la cual empezar. Tras obtener estos datos el servicio lee cada una de las líneas del archivo, si estas se encuentran en el rango requerido cifra cada una de las líneas obtenidas por separado y la despliega para que el aplicativo las use.

Aplicación de Android.

La aplicación es la parte más importante del sistema ya que interactúa con los vendedores y lleva el modelo de relación de datos. Debido a un límite en RAM que Android establece a 64MBs por aplicación, cargar todos los clientes, en clases separadas, hace que la memoria crezca y no sea factible cargar los datos; de la misma manera establecer un modelo relacional en clases no es óptimo por lo que se usara la base de datos de los sistemas Android, SQLite, para crear un modelo que pueda cargar los datos y guardarlos de manera permanente hasta que se decida reconstruir la base para cargar nuevos datos. El modelo de la base de datos es la siguiente.

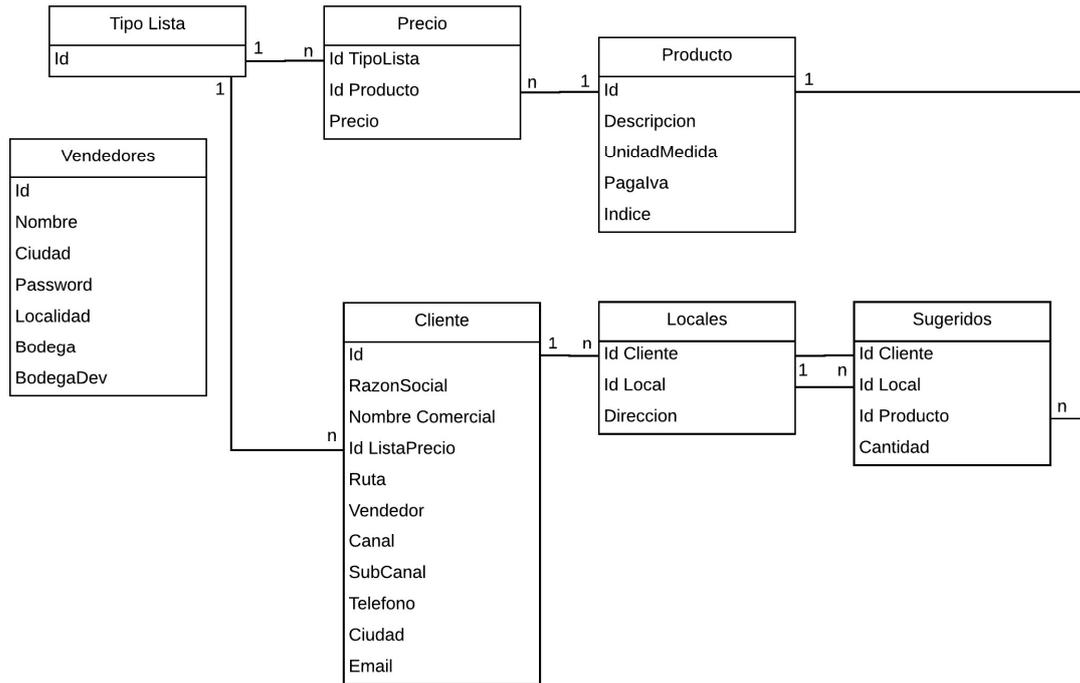


Figura 2: Estructura de la Base de Datos.

La figura muestra siete tablas que vinculan los productos a su lista de precio, los sugeridos con los productos y con el cliente, lleva los datos de los vendedores para así poder funcionar correctamente. Por seguridad del dato se realiza una comprobación: si los productos sugeridos se encuentran dentro de las opciones reales del cliente, en el caso de que la lista de clientes haya sido actualizada pero no se actualizaron los sugeridos.

Una vez obtenido el modelo relacional de los datos, se prosigue a diseñar los diagramas de actividad de la aplicación para definir cómo se van a cargar y usar los datos. En esta fase también surgen las primeras ideas del flujo de la aplicación y las pantallas. Los diagramas de actividad son divididos en tres procesos que tiene la aplicación, el descargar la información, el inicio de sesión y la consulta de sugeridos. Los diagramas obtenidos son:

1. Descarga de información.

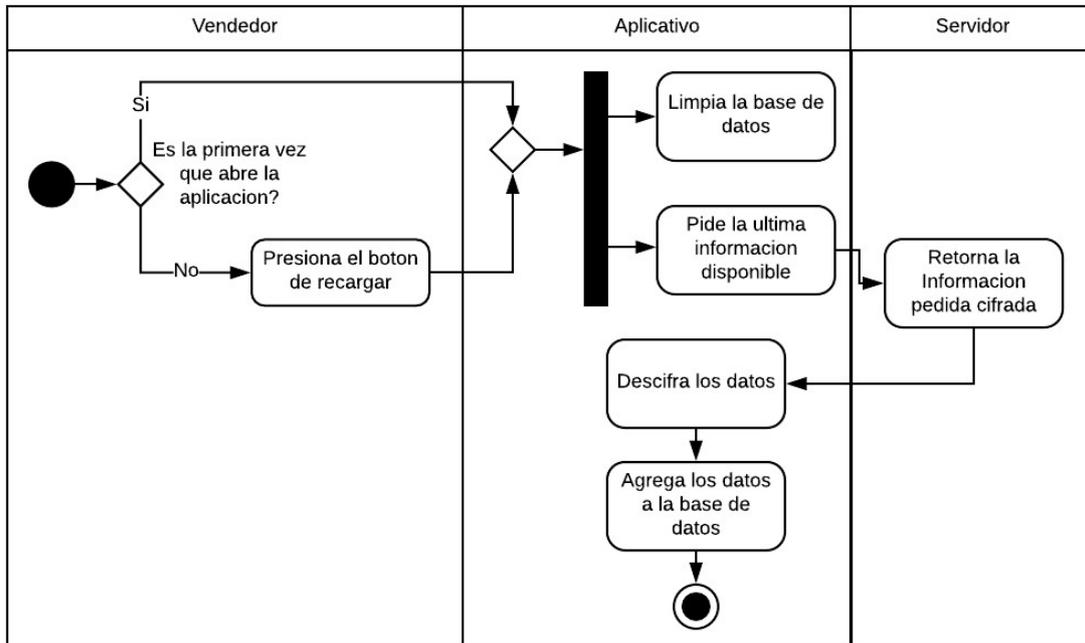


Figura 3: Diagrama de actividad de descarga de información.

La descarga de la información surge si el usuario presiona el botón de recargar o si es la primera vez que la aplicación entra en funcionamiento. El flujo muestra como siempre que se va a cargar nueva información del servidor, se tiene que limpiar la base de datos, ya que toda la información antigua no es necesaria y debe ser reemplazada. Para poder agregar los datos a la base se requiere que se los descifren. Una vez el flujo se completa la aplicación queda hábil para su uso.

2. Inicio de sesión,

El inicio de sesión es un flujo básico que consta del ingreso de los datos y la comprobación de los mismos. Para la comprobación solo se necesita una consulta a la base. Para evitar que la sesión se cierre automáticamente tras cerrar la aplicación se debe guardar en la memoria del teléfono la información de la sesión para que así siempre se haga un inicio automático hasta que se cierre la misma.

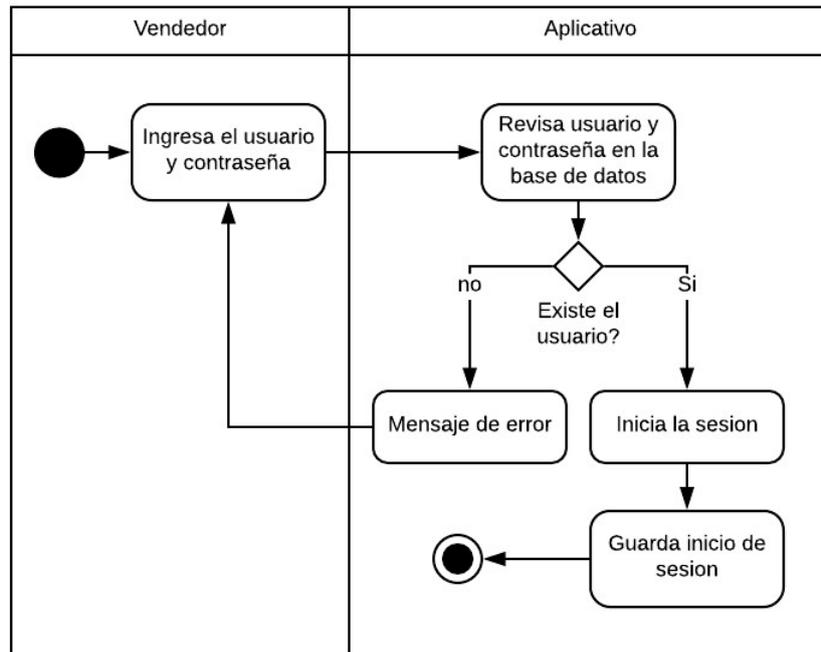


Figura 4: Diagrama de actividad de inicio de sesión.

3. Consulta de sugeridos.

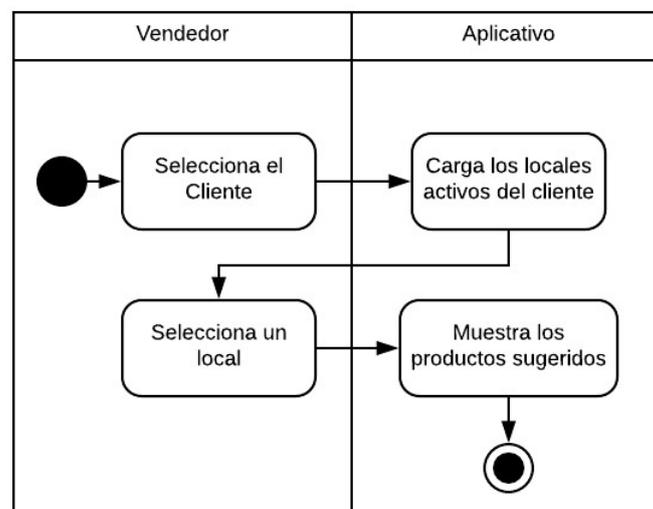


Figura 5: Diagrama de actividad de descarga de sugeridos.

Para obtener los sugeridos se debe ingresar cliente y local, el local siempre va a tener que ser elegido después del cliente, en el flujo se puede percatar que el usuario va a tener que hacer dos selecciones. Siguiendo este formato se decidió que va existir dos pantallas, una con

la lista de clientes y otra con la lista de locales vinculados al cliente previamente elegido. La pantalla de sugeridos no tiene una funcionalidad más que mostrar los datos.

4. Cerrar sesión.

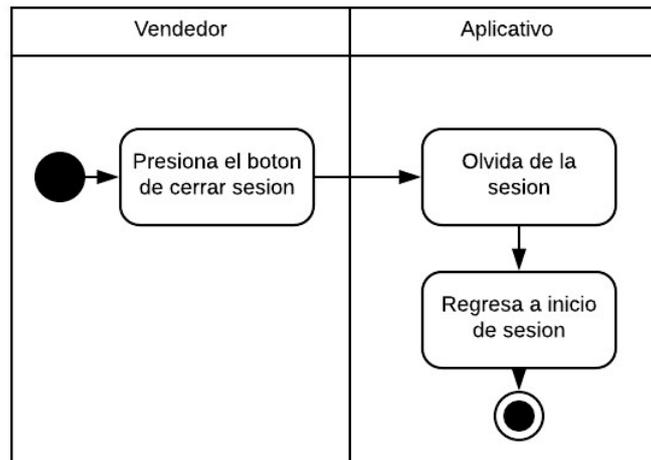


Figura 6: Diagrama de actividad de cerrar sesión.

Al final para cerrar sesión el vendedor se ve obligado a presionar un botón que cierre su sesión, la aplicación borrará de su memoria los datos de la sesión anterior y devolverá al usuario a la pantalla de inicio de sesión.

El análisis de los diagramas de la actividad ayuda a definir el número de pantallas y que funcionalidad de cada una en la aplicación. En este caso son necesarias cinco pantallas con la siguiente funcionalidad.

1. Pantalla de inicio de sesión.

La pantalla de inicio de sesión consta de dos botones: uno de recarga de datos y que empiece el inicio de sesión. Incluye dos campos de texto en lo que se pueda ingresar el usuario y contraseña que serán usados para el inicio de sesión.

2. Pantalla principal de la aplicación.

En la pantalla principal debe existir la opción de iniciar la búsqueda de sugeridos y la de cerrar sesión. Esta pantalla debe desplegar las dos opciones fáciles de seleccionar y debe

tener el logo de la empresa, ya que será lo que el vendedor observará cada vez que inicie la aplicación.

3. Pantalla de selección de cliente.

La pantalla de selección de cliente debe tener un botón para ir atrás y una lista con todos los clientes. Debido a la extensa cantidad de clientes debe existir una barra de búsqueda que permita filtrar y encontrar al cliente que se necesite.

4. Pantalla de selección de local.

La pantalla de selección de local es exactamente igual a la de selección de cliente pero tiene que cargar los datos según el local elegido anteriormente.

5. Pantalla de muestra de productos sugeridos.

La pantalla de muestra de productos solo debe contener un botón para regresar y la lista de los productos. Los elementos de la lista deben contener la descripción del producto y la cantidad sugerida de la venta.

Una vez definidas las pantallas se armó una aplicación que solo mostraba visualmente como iba a ser el flujo definido y los diseños de las mismas. Todos los diseños visuales de la aplicación fueron aprobados por la gerencia de Alimec, que a su vez aprobó la instalación del lenguaje estadístico R en el servidor con lo que se pudo empezar la fase de desarrollo.

Antes de iniciar esta fase, se realizó una investigación sobre los modelos posibles estadísticos y su factibilidad de uso. La investigación se basó en la minería de datos en archivos de texto plano y los modelos estadísticos de predicción, la cual se tratará en la siguiente sección.

Desarrollo del sistema

El desarrollo del sistema se llevó a cabo en un mes y medio de trabajo debido a complicaciones de la empresa, los datos no estuvieron disponibles en el tiempo estipulado y por ende el desarrollo e investigación de la parte estadística sufrió retrasos prominentes. Debido

a esto, se empezó con el servicio REST, el cual al solo necesitar cifrar y enviar datos sin importar la naturaleza de ellos se pudo realizar sin la necesidad de los archivos reales.

A la par del desarrollo del servicio, la aplicación también fue diseñada. Las transferencias de datos entre las clases de Java de descifrado y el cifrado del servicio fue terminado tras la entrega de los datos básicos de la aplicación, con excepción de los pedidos. Las pruebas del servicio con la aplicación fueron realizadas con archivos de producción y se logró que funcionen completamente.

Debido a la dificultad de la obtención de los datos del pedido por parte del área de sistemas, se prosiguió a armar la base de datos de la aplicación, la cual no depende del archivo de pedidos. El proceso de diseño de la base en la aplicación tuvo que aceptar todos los datos de los archivos ya que en ese momento no se conocía su validez; se decidió guardarlos en la base por la posibilidad de que, aunque no sirvan en el momento, sean necesarios para potenciales versiones siguientes del sistema con más funcionalidad. Una vez el modelo de la base fue probado se empezó a desarrollar los métodos que darían acceso a esta.

El desarrollo del servicio de inicio de usuario, obtención de clientes, locales y sugeridos fueron simples consultas a la base con parámetros que se guardaban en un singleton. La información del inicio de sesión fue guardada en las preferencias de la aplicación y esto se revisaba cada vez que la misma se reinicie. Una vez la aplicación fue terminada, se obtuvo el archivo de pedidos bajo las especificaciones solicitadas con anterioridad; lo que permitió el inicio de la investigación y desarrollo del programa estadístico.

Para poder obtener una lista de sugeridos se tuvo que investigar con que preceptos se podía obtener estos datos. Todo lo que se debía relacionar para poder obtener un resultado lógico que a su vez no necesitara de un input humano, sino que podía ser calculado de manera automática debía tener relevancia con las ventas reales de la empresa. Toda la información con

la que se contaba no se la podía relacionar directamente ya que no mostraba explícitamente las relaciones que uno deseaba encontrar.

Un limitante fue que, aunque se tuviera la información de relación, no había manera de definir qué productos eran más importantes que otros, acorde con los datos. Para solucionar estos problemas se usaron dos métodos: la minería de datos en los archivos que fueron entregados y los modelos estadísticos para así poder predecir tendencias y probabilidades de compra. Esto permitió obtener un modelo de datos que pudo ser usado para realizar predicciones basadas en el archivo de pedidos, ya que, cada cliente posee una relación con el producto y la cantidad. Así se procedió a armar una matriz vacía con todos los clientes existentes como filas y cada producto como una columna. En una segunda iteración se llenan los datos con la cantidad que se vendió. Para obtener una asociación real se crea una copia de la matriz que vuelve cualquier número que no sea 0 en verdadero y al resto como falso. Se conservan ambas matrices ya que la primera ayudara a obtener un promedio para la predicción de la cantidad sugerida.

Una vez se seleccionó el modelo y se tienen las variables con las cuales entrenar se procede a realizar una sub-selección para poder encontrar aquellas que son relevantes. En el caso de que no se haga una elección de variables el modelo puede que termine convergiendo, y siendo incapaz de predecir la probabilidad, lo cual se dio en las pruebas. Acorde con Karp (1998) las mejores maneras de seleccionar las variables independientes de un grupo, son usando los algoritmos de forward, backward o stepwise, los cuales van probando un modelo quitando o agregando variables independientes hasta que el modelo sea lo más fiable posible; una vez que se tengan varios modelos elegir el que mejor represente los datos estadísticos. Usando las observaciones del autor en los modelos logísticos se planteó el uso de estos selectores de variables independientes pero un problema con la capacidad del servidor freno el desarrollo.

Debido a la existencia de 308 productos, existen muchas variables independientes en cada modelo de cada producto, lo hacía que los cálculos se demoren más de lo que el servidor podía procesar, con 308 productos se tenían que realizar 308 modelos que contaban con 307 variables independientes cada uno y los algoritmos de selección se tardaban más de una hora por cada modelo. Con esta cantidad, el servidor podría haber tardado alrededor de 12 días y medio; cuando su tiempo de procesamiento era al máximo un día; ya que, los vendedores trabajan de lunes a sábado.

La solución del problema fue crear una matriz de correlación con los datos del promedio de las ventas de los productos, la matriz usada para la creación de la matriz de booleanos; de esta manera se recortó el grupo inicial de variables que no cumplían un requerimiento, el cual estipulaba que la correlación de las variables debía estar sobre un número mínimo específico. Así se obtuvieron los siguientes tiempos por modelo, obteniéndolos a partir de correr los algoritmos en un 10% de los registros. Se generaron también, modelos solo recortados por la correlación para que sirvan como comparación, sin usar los métodos de selección dan resultado. La tabla fue llenada por los datos hasta que se encontraba que el método de selección no estaba sirviendo o se el set completo duraba más de 24 horas.

Tabla 1: Tiempos de selección de variables acorde a su algoritmo.

Correlación Mínima	Numero de Variables Iniciales	Numero de Variables Finales	Tiempo de regresión por modelo en segundos	Tiempo de todos los modelos
Backwards				
0.5	3,73	3,20	0,795210	4,082080 m
0.45	5,03	4,00	2,107278	10,817360 m
0.4	7,10	5,10	6,499848	33,365886 m
0.35	10,20	6,57	17,106666	1,463570 h
0.3	14,43	8,47	46,942104	4,016157 h
0.25	21,97	11,67	136,374732	11,667615 h
0.2	33,80	16,63	427,016880	36,533666 h

Forward				
0.5	3,73	3,73	0,075355	23,209287 s
0.45	5,03	5,03	0,089365	27,524460 s
0.4	7,10	7,10	0,119270	36,735006 s
0.35	10,20	10,20	0,169685	52,262980 s
0.3	14,43	14,43	0,236695	1,215035 m
0.25	21,97	21,97	0,376810	1,934293 m
0.2	33,80	33,80	0,601277	3,086555 m
Stepwise				
0.5	3,73	3,20	1,428327	7,332078 m
0.45	5,03	4,00	2,326421	11,942292 m
0.4	7,10	5,10	5,974542	30,669316 m
0.35	10,20	6,57	15,587626	1,333607 h
0.3	14,43	8,47	53,384003	4,567298 h
0.25	21,97	11,67	159,541898	13,649695 h
0.2	33,80	16,63	529,363532	45,289991 h
Solo por Correlación				
0.5	3,73	3,73	0,076525	23,569730 s
0.45	5,03	5,03	0,091985	28,331296 s
0.4	7,10	7,10	0,117851	36,298046 s
0.35	10,20	10,20	0,167437	51,570719 s
0.3	14,43	14,43	0,234508	1,203807 m
0.25	21,97	21,97	0,357951	1,837479 m
0.2	33,80	33,80	0,583629	2,995963 m
0.15	54,47	54,47	1,047001	5,374605 m
0.1	90,67	90,67	2,142847	10,999945 m
0.05	148,13	148,13	4,524433	23,225423 m
0	308,00	308,00	29,209656	2,499048 h

Analizando la tabla se obtuvo que el único algoritmo válido para la estadística de los datos es backwards, debido a que en menos tiempo predice las mismas variables que stepwise. El algoritmo de forward simplemente decide añadir todas las variables y como stepwise es una combinación entre los dos su tiempo incrementado con respecto a backwards es por el conflicto entre añadir y quitar variables, llegando a la misma conclusión de la misma manera. El método de solo seleccionar por correlación también se volvió interesante al poder predecir con una velocidad bastante alta. Tras seleccionar los métodos que se van a usar se verificó que correlación mínima debía usarse. Las correlaciones inferiores o iguales a 0.25 empiezan a subir el tiempo exponencialmente y se pasan de las 24 horas por lo que dejan de ser prácticas para el proyecto en cuestión.

Para analizar el error de una regresión logística se tiene que revisar si el error que este presenta en el sistema, cuantas veces se equivocó porcentualmente, es suficientemente pequeño para que el modelo prediga de manera correcta; el otro punto que hay que tomar en cuenta es si el modelo mejora el resultado de sí, solo se hubiese elegido el resultado más repetido (Peng, Lee e Ingersoll, 2002). Se usaron los modelos anteriormente seleccionados y se obtuvieron los siguientes resultados de sus errores.

Tabla 2: Errores de los modelos seleccionados.

Correlación Mínima	Porcentaje de Error	Porcentaje de error si todos son negativos	Mejora del modelo contra todos negativos	Porcentaje de productos que Convergen	Número de Productos que no son predecibles
Backwards					
0.5	8,56%	16,79%	50,97%	36,67%	113
0.45	8,51%	16,79%	50,69%	36,67%	113
0.4	10,31%	19,21%	53,68%	26,67%	82
0.35	9,84%	17,25%	42,96%	10,00%	31
0.3	9,41%	16,65%	43,45%	6,67%	21
0.25	9,75%	16,93%	42,40%	3,33%	10
Solo por Correlación					
0.5	8,56%	16,79%	49,03%	36,67%	113
0.45	8,51%	16,79%	49,29%	36,67%	113
0.4	10,30%	19,21%	46,36%	26,67%	82
0.35	9,83%	17,25%	43,03%	10,00%	31
0.3	9,40%	16,65%	43,50%	6,67%	21
0.25	9,75%	16,93%	42,43%	3,33%	10
0.2	9,49%	16,93%	43,97%	3,33%	10
0.15	9,21%	16,78%	45,11%	0,00%	-
0.1	8,98%	16,78%	46,49%	0,00%	-
0.05	8,70%	16,78%	48,18%	0,00%	-
0	8,41%	15,83%	46,90%	96,67%	298

El porcentaje de modelos que convergen muestra la cantidad de productos que no van a poder ser predichos por los modelos. El error de si todos fuesen negativos, es variable, ya que no se toman en cuenta a productos en los cuales su modelo convergió, aun así, todos ellos mejoraron su cifra en más de un 40%. Los modelos de correlación mínima entre 3.5 a 5 tienen un porcentaje muy alto de modelos que convergen por lo que no se pueden tomar en cuenta para la selección estos modelos.

La comparación de errores entre backwards y selección de una variable solo por correlación es muy pequeña; en comparación con el tiempo que les toma predecir. Debido al bajo tiempo que la selección solo por correlación tiene se puede revisar más correlaciones mínimas. Si se seleccionan todas las variables el porcentaje de productos que convergen es demasiado alto por lo que se revisan las correlaciones entre 0.2 y 0.05. Al llegar a 0.15 de correlación mínima todos los productos pueden ser predichos y el error baja progresivamente. El mejor modelo es la elección solo por correlación con una correlación mínima de 0.05; el cual, tiene el menor error, puede predecir todos los productos y tiene la mayor mejora entre las correlaciones aceptadas.

Tras esto se corre el modelo seleccionado y se crea una matriz de cliente por producto con las probabilidades predichas. Por cada cliente se revisan los falsos positivos que se encuentren por encima de una cifra dada, en este caso del 70%. Se seleccionan máximo 5 productos por cliente, los de la probabilidad más alta, para que sean enviados al aplicativo móvil. Para decidir la cantidad sugerida se obtiene la media de los productos más correlacionados con el sugerido. Para finalizar se genera el archivo con los datos obtenidos, el cual una vez guardado en el servidor empezaron las pruebas reales del aplicativo.

Tras algunas pruebas y revisiones de consistencia de datos entre el servidor, la aplicación y los análisis se terminó el proyecto. La entrega del proyecto fue realizada y se instaló en los vendedores la aplicación para realizar la venta cruzada, terminando así el proceso de desarrollo e implementación del proyecto.

CONCLUSIONES

El proyecto de venta cruzada fue entregado exitosamente y de la experiencia se aprendió que para realizar proyectos los tiempos establecidos deben ser cumplidos por ambas partes. El lapso que se requería para completar el proyecto se vio severamente reducido por las demoras que se presentaron en la entrega de archivos, el desarrollo de modelos estadísticos tuvo que ser realizado de manera rápida junto con las pruebas de los mismos.

Los problemas que se presentaban por no acudir directamente con la gerencia a pedir lo necesario para proseguir, sino depender en el área de sistemas de la empresa para que se entregue todo, este fue uno de los mayores problemas. Una vez se contactó con la gerencia y se expresó las preocupaciones se entregó lo solicitado en menos de dos días; de esta manera, se puede concluir que para realizar proyectos de desarrollo para una empresa es necesario tener una relación positiva con la gerencia para que se obtenga todo lo necesario rápidamente y sin tener que pasar por capas inferiores de la misma.

El modelo que se uso tuvo que ser ajustado varias veces para no usar variables irrelevantes o datos que no eran fiables. Un modelo estadístico no va a poder describir una tendencia, si esta lleva consigo variables con datos no fiables; modelos simples que usan solo lo necesario para encontrar relaciones entre variables son mucho más viables que modelos complejos que tratan de vincular información de varias índoles.

Uno de los problemas en usar un modelo que solo relaciona los historiales de venta de los clientes, es que no se tienen suficientes datos para relacionar información de venta cruzada. Por ejemplo, la venta cruzada puede relacionar también zonas geográficas, tamaño del cliente, frecuencia de compra, entre otros para así poder generar modelos mucho más complejos.

Estadísticamente se generan modelos para poder predecir tendencias que serán probadas con nuevos datos; en el caso de Alimec, deberían ser usados para poder predecir y sugerir a nuevos clientes acorde a su pedido original. El modelo usado propone que se puede

observar discrepancias entre el modelo de la tendencia y las acciones reales para así encontrar la posibilidad de que falsos positivo, que a su vez puedan ser usados como sugeridos para de esta manera, hacer que la realidad empiece a comportarse más como el modelo generado.

Según la gerencia de Alimec, la aplicación es de bastante útil y sobrepasa los estándares esperados, se obliga a los vendedores a usarla para así lograr más ventas por lo que no se pudo obtener datos concretos de la utilidad de la aplicación por los vendedores ya que se volvió una obligación su uso, los cambios en ventas solo podrán ser analizados tras un periodo de cierre trimestral de ventas al analizar la contabilidad.

REFERENCIAS BIBLIOGRÁFICAS

- Aguayo, M. (2012). *Cómo hacer una Regresión Logística con SPSS® “paso a paso”*.(I). Sevilla: Documento FABIS
- Derby, N. (2017). *Maximizing Cross-Sell Opportunities with Predictive Analytics for Financial Institutions*. Seattle: Stakana Analytics.
- García, J. A., & Acevedo, A. M. (2010). *Análisis para predicción de ventas utilizando minería de datos en almacenes de ventas de grandes superficies* (Unpublished master's thesis). Universidad Tecnológica de Pereira. Facultad de Ingeniería Eléctrica, Electrónica, Física y Sistemas. Programa de Ingeniería de Sistemas y Computación.
- Harding, F., (2002). *Cross-Selling Success: A Rainmaker's Guide to Professional Account Development*. Avon, MA: Adams Media.
- Karp, A. H. (1998). *Using logistic regression to predict customer retention*. In *Proceedings of the Eleventh Northeast SAS Users Group Conference*. <http://www.lexjansen.com/nesug/nesug98/solu/p095.pdf>.
- Peng, C. J., Lee, K. L., & Ingersoll, G. M. (2002). An Introduction to Logistic Regression Diagnostics. *The Journal of Educational Research*,96(1), 3-14.
- Riquelme Santos, J. C., Ruiz, R., & Gilbert, K. (2006). Minería de datos: Conceptos y tendencias. Inteligencia artificial. *Revista Iberoamericana de Inteligencia Artificial*, 10(29), 11-18.
- Schmitz, C., Lee, Y. C., & Lilien, G. L. (2014). Cross-selling performance in complex selling contexts: an examination of supervisory-and compensation-based controls. *Journal of Marketing*, 78(3), 1-19
- Vila-Lopez, N., Kuster-Boluda, I., & Escamilla-Santana, M. A. (2015). Formación para fuerza de ventas mexicana. *Revista de Ciencias Sociales (Ve)*, 21(3).

Wong, R. C., Fu, A. W., & Wang, K. (2005). Data Mining for Inventory Item Selection with Cross-Selling Considerations. *Data Mining and Knowledge Discovery*, 11(1), 81-112.
doi:10.1007/s10618-005-1359-6

ANEXO A: ACTA DE ENTREGA RECEPCION DEL SISTEMA

**Acta Entrega – Recepción Definitiva del Proyecto de
Predicción de Sugeridos de Venta Cruzada**

1.- Comparecientes.-

En la ciudad de Quito, a los 2 días del mes Mayo del año 2018, comparecen a la firma de la presente Acta de Entrega-Recepción del Proyecto de Predicción de Sugeridos de Venta Cruzada, por una parte en representación de la empresa Alimentos Ecuatorianos S.A. - Alimec, señor Esteban Serrano, y por otra parte el señor Stephan Zurita Serafim en representación propia.

2.- Antecedentes.-

La empresa Alimec cuenta con un aplicativo móvil para sus vendedores donde se despliegan los productos que las tiendas puedan adquirir. El aplicativo es muy funcional para los vendedores pero Alimec con el objetivo de expandir la venta de diferentes productos a sus clientes (tiendas) requiere una solución informática que pueda realizar predicciones estadísticas de venta cruzada para que los vendedores nuevos o inexpertos puedan tener oportunamente un listado de productos sugeridos para que sean ofrecidos a las tiendas.

3.- Liquidación Técnica.-

3.1 - Se realizó por parte del señor Stephan Zurita el desarrollo de las Especificaciones Técnicas solicitadas de acuerdo a lo requerido por la empresa Alimec, se corrigieron los problemas presentados durante las pruebas, y se puso en producción el sistema una vez que este fue aprobado el 15 de Enero de 2018. Cumpliéndose la totalidad del objeto de la contratación.

3.2 – En conclusión los servicios prestados fueron realizados a satisfacción.

4.- Liquidación de Plazos.-

4.1- El plazo para la ejecución del Proyecto de Predicción de Sugeridos de Venta Cruzada era de 240 días.

4.2 – Se finalizaron los trabajos contratados y la puesta en producción del sistema el 29 de abril de 2018.

4.3 – En conclusión los servicios prestados fueron finalizados dentro del plazo programado.

5.- Liquidación Económica.-

5.1 – El Proyecto de Predicción de Sugeridos de Venta Cruzada fue realizado como tesis para la graduación como ingeniero del señor Stephan Zurita y no representa una deuda monetaria para la empresa Alimec

5.2 – La liquidación económica de este proyecto es de US\$ 0,00. La empresa Alimec no debe pagar ningún valor al señor Stephan Zurita.

6.- Conformidad.-

En consecuencia de lo expuesto en los numerales 3, 4, y 5 de la presente Acta, las partes proceden a la Entrega-Recepción del Proyecto de Predicción de Sugeridos de Venta Cruzada.

Para constancia de las responsabilidades de todo lo estipulado en la presente acta entrega-recepción definitiva, en dos copias del mismo tenor y efecto, firman:

<p>EL CONTRATANTE</p> <p><i>Esteban Serrano M.</i></p> <p>Firma: _____</p> <p>Nombre: Alimentos Ecuatorianos S.A. – Alimec</p> <p>Nombre Representante: _____</p> <p>CI#: <u>17044187799</u></p>	<p>EL CONTRATISTA</p> <p>Firma: <i>Stephan Zurita</i></p> <p>Nombre: Stephan Zurita Serafim</p> <p>CI#: <u>1715334642</u></p>
--	---