

Índice de Contenidos

| | | |
|-------|----------------------------------|----|
| 1 | Introducción | 1 |
| 1.1 | Antecedentes | 1 |
| 1.2 | Planteamiento del problema | 3 |
| 1.3 | Sistematización | 4 |
| 1.3.1 | Diagnóstico | 4 |
| 1.3.2 | Pronóstico | 9 |
| 1.3.3 | Control Pronóstico | 9 |
| 1.4 | Objetivos | 10 |
| 1.4.1 | Objetivo General | 10 |
| 1.4.2 | Objetivos Específicos | 10 |
| 1.5 | Justificación | 10 |
| 1.5.1 | Justificación Práctica | 10 |
| 1.5.2 | Justificación Teórica | 11 |
| 1.6 | Alcance | 12 |
| 1.7 | Estudio de Factibilidad | 13 |
| 1.7.1 | Factibilidad Técnica | 13 |
| 1.7.2 | Factibilidad Operativa | 16 |
| 1.7.3 | Factibilidad Económica | 16 |
| 2 | Marco de Referencia | 19 |
| 2.1 | Marco Teórico | 19 |
| 2.2 | Marco Conceptual | 22 |
| 2.2.1 | API's de Programación | 22 |
| 2.2.2 | RUP | 23 |
| 2.2.3 | Modelado UML | 24 |
| 2.2.4 | Programación Orientada a Objetos | 25 |
| 2.2.5 | Tarjeta Inteligente | 25 |
| 2.3 | Marco Temporal/Espacial | 26 |
| 2.4 | Marco Legal | 26 |
| 3 | Metodología | 27 |
| 3.1 | Metodología de la Investigación | 27 |
| 3.1.1 | Unidad de Análisis | 27 |
| 3.1.2 | Tipos de Investigación | 27 |

| | |
|--|----|
| 3.2 Metodología Informática | 27 |
| 3.2.1 Metodología Orientada a Objetos | 27 |
| 3.2.2 Proceso de Ingeniería | 28 |
| 4. Proceso de Desarrollo | |
| 29 | |
| 4.1. Fase de Inicio | |
| 29 | |
| 4.1.1. Flujo Modelado del Negocio | |
| 29 | |
| 4.1.2. Flujo de Requisitos | |
| 30 | |
| 4.1.3. Flujo Análisis | |
| 30 | |
| 4.1.4. Flujo Requisitos (Iteración 2) | |
| 32 | |
| 4.1.5. Flujo Diseño (Iteración 2) | |
| 32 | |
| 4.2. Fase de Elaboración | |
| 37 | |
| 4.2.1 Flujo Modelado del Negocio | |
| 37 | |
| 4.2.2 Flujo Requisitos | |
| 42 | |
| 4.2.3 Flujo Análisis | |
| 48 | |
| 4.2.4 Flujo Diseño | |
| 48 | |
| 4.2.5 Flujo Despliegue | |
| 52 | |
| 4.3 Fase de Construcción | |
| 54 | |
| 4.3.1 Flujo de Requisitos | |
| 54 | |
| 4.3.2 Flujo de Análisis | |
| 54 | |
| 4.3.3 Flujo Diseño | |
| 56 | |
| 4.3.4 Flujo Pruebas | |
| 57 | |
| 4.3.5 Flujo Requisitos (Iteración II) | |
| 58 | |
| 4.3.6 Flujo Pruebas (Iteración 2) | |
| 64 | |
| 4.3.7 Flujo Despliegue (Iteración 2) | |
| 66 | |
| 4.4 Fase de Transición | |
| 68 | |
| 4.4.1 Flujo Pruebas | |
| 68 | |

| | | |
|-----|--------------------------------|----|
| 5. | Conclusiones y Recomendaciones | |
| | 71 | |
| 5.1 | Conclusiones | |
| | 71 | |
| 5.2 | Recomendaciones | |
| | 72 | |
| | BIBLIOGRAFÍA | |
| | 73 | |
| | ANEXOS 1 | 75 |
| | ANEXOS 2 | 78 |
| | ANEXOS 3 | 83 |

1 INTRODUCCIÓN

1.1 ANTECEDENTES

Hoy en día el mercado farmacéutico ha crecido notablemente, exigiendo estas nuevas estrategias para el apoderamiento del mercado una de ellas son las franquicias de farmacias de las más conocidas Fybeca, Sana Sana, Cruz Azul y Pharmacy's estas cadenas están en constante competencia para ser las pioneras utilizando alianzas estratégicas con proveedores, sistemas informáticos y tecnología de punta de manera de dar un mejor servicio al cliente.

Las farmacias que no pertenecen a una franquicia poco a poco van desapareciendo ya que no pueden competir con precios, atención al cliente, variedad de productos entre otros servicios como servicio a domicilio y consultas médicas por ejemplo. El cliente prefiere ir a cierta farmacia principalmente por el precio de los productos también cuando se acerca a una farmacia y tiene todos los productos de una receta y la atención de los empleados con el cliente.

En muchas de estas cadenas se usan las muy conocidas tarjetas de descuento que identifican al usuario y se le hace un descuento según el tipo de usuario que este sea, estas tarjetas pueden ser de varios tipos uno de los más usados son las tarjetas magnéticas y su característica es tener una banda magnética, en donde a través de un lector se graba y se leen los datos del cliente. Otro tipo muy conocido en el mundo son las tarjetas inteligentes que a diferencia de las tarjetas magnéticas, tienen integrado un procesador el cual permite realizar diversos procesos con los datos almacenados como pueden ser: almacenamiento, autenticación de usuarios, encriptación de claves, etc. El grupo DIFARE es un grupo empresarial que funciona en todo el Ecuador con sus cadenas de farmacias Cruz Azul, Farmacys y Farmacias Comunitarias en todas estas cadenas sus descuentos lo manejan con la tarjeta FARMAVIP que es una tarjeta electrónica la que maneja datos de usuario, tipos de usuario, planes de descuentos y crédito empresarial según la empresa a la cual pertenezca el usuario.

Actualmente las cadenas de las farmacias tienen visitadores a médicos con los que hacen convenios, para recetar cierto tipo de medicamentos que son exclusivos de cada franquicia y en la farmacia por cada receta emitida por cierto médico, tiene que

ir registrando para después de un cierto tiempo dar un incentivo al médico. Cuando el cliente por error pierde la receta o esta es totalmente ilegible causa un gran problema, ya que el cliente puede enojarse con el dependiente que no entiende la receta y puede cambiarse de farmacia o a su vez si la receta se extravía, el cliente tiene que recurrir nuevamente al consultorio médico para que le den una nueva receta, lo cual es un gran problema porque hay personas que necesitan tomar su medicamento urgentemente.

Aprovechando la tecnología de tarjetas inteligentes utilizadas por el grupo empresarial Ecuafarmacias, con la tarjeta de descuento y crédito Farmavip, se podría dar un valor agregado a ésta, de manera que cuando el cliente se acerque a la farmacia el vendedor del mostrador sepa exactamente el crédito del cliente, el descuento y la receta médica que necesita. De este modo tanto el cliente como el grupo farmacéutico saldrían beneficiados, el cliente estará seguro que le están vendiendo los medicamentos de la receta y el grupo empresarial estaría asegurando que su cliente va a comprar en una de sus farmacias.

Así con el uso de nuevas tecnologías, las cadenas de farmacias se van apoderando de un mayor sector de mercado, el cliente recibe una mejor atención, el vendedor de mostrador simplifica su trabajo y mejoran los procesos del negocio.

El origen de la tarjeta inteligente se encuentra en Europa, a comienzos de los años 70 esta tarjeta es similar a las bancarias o a las de crédito, pero capaz de incorporar un dispositivo programable.

En los 80 se dispone ya de chips suficientemente pequeños, pero con unas capacidades de memoria muy reducidas.

La tarjeta inteligente se constituye por un plástico de forma similar a una tarjeta de crédito, donde se observa un procesador (microchip) insertado en el plástico, en el cual se almacena información permitiendo mayor eficiencia, que en el sistema de tarjetas de crédito tradicional en cuanto agilidad y seguridad que innova y expande el servicio para el usuario.

Es a principios de los 90 cuando las tarjetas inteligentes inician su despegue al empezar la telefonía móvil GSM, inicialmente con tarjetas con 1K de memoria. La Fase 1 de GSM requería muy poca capacidad de memoria.

Aproximadamente hace 20 años en Ecuador empezó la tendencia a la creación de franquicias dedicadas a desarrollar el mercado farmacéutico en el país, especializándose en la distribución y abastecimiento de productos para las farmacias del país.

El objetivo de las franquicias es dar un servicio especial, rápido, y frecuente de logística, el desarrollo e implementación de tecnología, a la farmacia para la administración de su negocio, y un amplio surtido de productos mejorar descuentos, así las franquicias buscan ser líderes en el mercado de distribución farmacéutica en el Ecuador.

Por ejemplo el Grupo Difare, con sus 6 empresas, Difare, Dyvenpro, Asegensa, Dires, Difarnova, y Cruz Azul, se proyecta y prepara constantemente hacia el futuro, consciente de su liderazgo y su responsabilidad ante sus colaboradores y la sociedad en general, siempre con el objetivo de cumplir con su eslogan: COMPROMETIDOS CON EL EXITO DE SUS CLIENTES. El Grupo Difare cuenta en la actualidad con la mayor fuerza de distribución farmacéutica, del país a través de 2 bodegas principales ubicadas en Quito y Guayaquil, una flota de transporte de más de 30 vehículos, y 7 oficinas comerciales y 6 puntos de venta, distribuidos estratégicamente en las poblaciones más importantes del país. Dicha infraestructura permite atender a más de 3500 clientes, manejar más de 130 proveedores, más de 4000 ítems en el inventario, y procesar más de 1200 pedidos diarios, empujada por el motor de su recurso más importante, el recurso humano conformado por más de 700 colaboradores.

1.2 Planteamiento del problema

Cuando el cliente de una farmacia no sabe el nombre de sus medicamentos, principalmente cuando son de medicación continua, la receta médica no es entendible o esta se extravió o no sabe la forma de tomar su medicamento, el cliente tendría el trabajo de ir nuevamente al médico o hacerle una llamada, con el propósito que le indique nuevamente su receta y que esta sea nuevamente escrita. Otro problema para los empleados de una farmacia es registrar la receta con los datos del médico, ya que en horas de mucha afluencia de clientes estos necesitan

ser despachados rápidamente y si no es así lo mas seguro es que vayan a otra farmacia.

Las farmacias tienen convenios con médicos cuando estos recetan cierto medicamento, este tiene que ser registrado por la farmacia para que después de cierto tiempo se de una bonificación al médico, el problema radica cuando hay mucho afluencia de clientes, este proceso se obvia por el vendedor de la farmacia con el fin de dar una atención más rápida al cliente.

¿El almacenamiento de recetas médicas en tarjetas inteligentes solucionará los problemas que conlleva las recetas manuales?

1.3 Sistematización

1.3.1 Diagnóstico.

El problema que radica la emisión de recetas manuales lo más común y que a todos nos ha pasado alguna vez es que se extravía y cuando nos acercamos a una farmacia, no nos pueden vender dichos medicamentos esto conlleva la incómoda situación de acudir nuevamente al médico para que nos realice nuevamente una nueva receta.

Otro problema muy común es que la receta es totalmente ilegible o a su vez es muy complicado leerla, con lo que el vendedor de mostrador de la farmacia se demora más tiempo hasta llegar a prácticamente adivinar que medicamento es el solicitado en la receta, el cliente se molesta por la demora y se retrasa el proceso de venta.

Cuando un cliente va con una receta esta tiene que ser registrada debido a que las farmacias tienen convenios con médicos y después de un plazo los médicos reciben una bonificación según el número de recetas emitidas, también el CONSEP exige el registro de recetas médicas, cuando un vendedor esta apurado tratando de atender el mayor número de clientes; por esta razón obvia el registro de las recetas.

El proceso de venta con recetas médicas sigue el siguiente esquema:

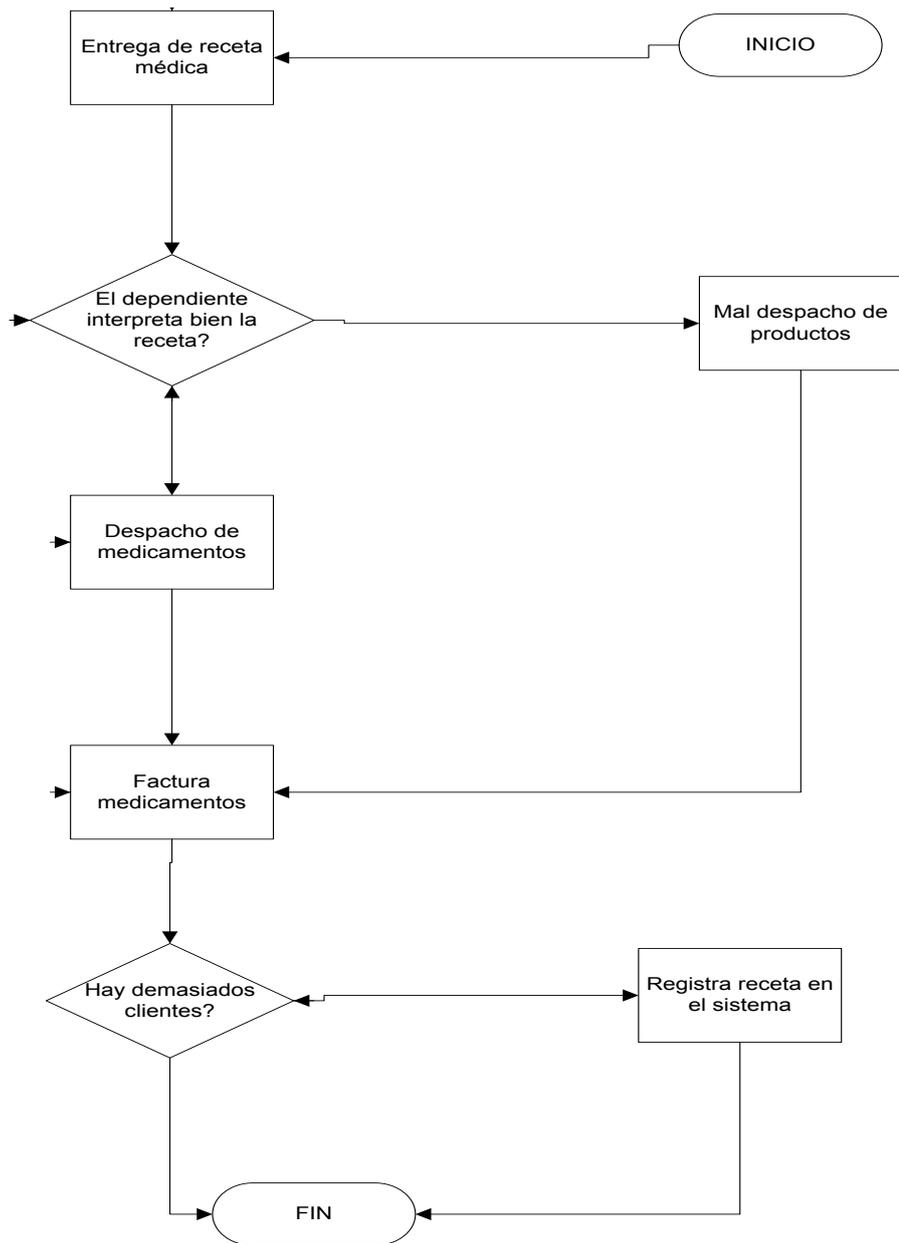


Figura 1: Diagrama de Emisión de recetas médicas

Tabla 1: Informe de diagrama de flujo

| Informe de diagrama de flujo | | | | | |
|-------------------------------------|-------------------------|---|-----------------|--------------|-----------------|
| | Nombre de patrón | Texto mostrado | Recursos | Costo | Duración |
| | Decisión | El dependiente interpreta bien la receta? | | | 1 minuto |
| | Decisión | Hay demasiados clientes? | | | 30 segundos |
| Recuento | 2 | | | | |
| Total | | | | | |
| | Proceso.4 | Entrega de receta médica | | | 30 segundos |
| | Proceso.4 | Factura medicamentos | | | 1 minuto |
| | Proceso.4 | Mal despacho de productos | | | 1,5 minutos |
| | Proceso.4 | Despacho de productos | | | 1,5 minutos |
| | Proceso.4 | Registra receta en el sistema | | | 30 segundos |
| Recuento | 4 | | | | |
| Total | | | | | |
| | Terminador | FIN | | | |
| | Terminador | INICIO | | | |
| Recuento | 2 | | | | |
| Total | 3,30 minutos | | | | |
| Total global | 6,30 minutos | | | | |

Análisis FODA

El Objetivo es disminuir los errores que se pueden cometer durante la venta de medicinas con recetas médicas manuales, mediante un sistema de recetas médicas almacenadas en tarjetas inteligentes.

Fortalezas

- Apoyo entre vendedores de mostrador.
- Alto prestigio de las franquicias.
- Excelente atención al cliente.
- Atención rápida.
- Buena capacitación de los vendedores.
- Buena infraestructura.

Debilidades

- Rotación de personal.
- Personal novato.
- Poca cantidad de dependientes.
- Complicado proceso de registrar las recetas en el sistema.

- Mala interpretación de la receta médica y recelo del vendedor al decir que no puede leer el medicamento de la receta.

Oportunidades

- Aprovechamiento de la infraestructura de manejo de tarjetas inteligentes.
- Mejorar los incentivos al empleado por buen desempeño.
- Dar un servicio extra al cliente y que no tiene la competencia.
- Dar la pauta para otras aplicaciones que trabajen con tarjetas inteligentes
- Interconexión con otros sistemas.
- Oportunidad de manejarse independiente del sistema farmacéutico.

Amenazas

- Problemas con las leyes del Ministerio de salud en el ámbito informático.
- Negativa por aprender nuevos procesos por parte de los empleados.
- Poca interés por los clientes.
- Extravío de la tarjeta inteligente.
- Falta de aceptación del sistema por parte de los médicos.

Tabla 2: Tabla de Pruebas de Integración

| | <p style="text-align: center;">FORTALEZAS</p> <ul style="list-style-type: none"> - Apoyo entre vendedores de mostrador - Alto prestigio de las franquicias - Excelente atención al cliente - Atención rápida - Buena capacitación de los vendedores - Buena infraestructura | <p style="text-align: center;">DEBILIDADES</p> <ul style="list-style-type: none"> - Rotación de personal - Personal novato - Poca cantidad de dependientes - Complicado proceso de registrar las recetas en el sistema - Mala interpretación de la receta médica - Recelo del vendedor al decir que no puede leer el medicamento de la receta |
|---|--|--|
| <p style="text-align: center;">OPORTUNIDADES</p> <ul style="list-style-type: none"> - Aprovechamiento de la infraestructura de manejo de tarjetas inteligentes - Mejorar los incentivos al empleado por buen desempeño - Dar un servicio extra al cliente y que no tiene la competencia - Dar la pauta para otras aplicaciones que trabajen con tarjetas inteligentes - Interconexión con otros sistemas - Oportunidad de manejarse independiente del sistema farmacéutico | <ul style="list-style-type: none"> - Fomentar el compañerismo - Seguimiento de los conocimientos y desempeño de los trabajadores - Constante capacitación de los vendedores | <ul style="list-style-type: none"> - Mejor selección de personal |
| <p style="text-align: center;">AMENAZAS</p> <ul style="list-style-type: none"> - Problemas con las leyes del Ministerio de salud en el ámbito informático - Negativa por aprender nuevos procesos por parte de los empleados | <ul style="list-style-type: none"> - El personal conoce muy bien el manejo de recetas en el farmacia - Mediante la capacitación del personal se mostrará que no es difícil de usar la | <ul style="list-style-type: none"> - Capacitarse muy bien sobre las leyes del Ministerio de salud en cuanto a la venta con recetas medicas - Indicar al personal que es una herramienta informática que les |

| | | |
|---|---|--|
| <ul style="list-style-type: none"> - Poco interés por los clientes - Extravío de la tarjeta inteligente | <p>herramienta y que será útil para ellos y los clientes</p> <ul style="list-style-type: none"> - Si se extravía la tarjeta tenemos de hacer la venta normalmente con la receta física | <p>ayudará en el desempeño de ventas y les quitará responsabilidades</p> <ul style="list-style-type: none"> - Hacer propaganda sobre este servicio. |
|---|---|--|

1.3.2 Pronóstico.

Si no se pone atención y no se da una solución a los problemas planteados como consecuencia, se tendrá una mala atención al cliente y la pérdida del mismo y de este modo dando más oportunidades a la competencia.

Tomando en cuenta que la interpretación de una receta con el tiempo se va haciendo más fácil, esto no ocurre en dependientes novatos corriendo el riesgo de una venta de medicamentos equivocados y esto puede generar en problemas muy graves para el cliente, para el dependiente y para la farmacia.

Cuando no se registra una receta del médico, con el cual se tiene un convenio que, no se está cumpliendo esto, puede derivar en la pérdida del médico y así disminuir las ventas del medicamento que esta patrocinando y la posibilidad de que el médico trabaje con la competencia.

La suma de estos problemas deriva en pérdidas económicas para las farmacias, también falta de seriedad, pérdida de posicionamiento en el mercado y una mala imagen empresarial

1.3.3. Control del Pronóstico

Con este proyecto se pretende agilizar eliminar los problemas que conlleva la emisión de recetas médicas manuales, agilizando así el proceso de ventas, la atención al usuario y la captación y retención del cliente.

Cuando se graba la receta en una tarjeta inteligente, no habrá equivocación al interpretar la receta médica y este será un proceso automático lo cual ahorrará tiempo al dependiente ya que no tendrá que tipear los insumos, tampoco correrá el riesgo principalmente en dependientes novatos de vender medicamentos que no son

los que pide el médico, el registro de la receta según el médico también será automático, el proceso de venta al cliente será mas rápido permitiendo así una mejor atención y una mejora en el proceso de ventas. La posología¹ de la receta también será clara y no habrá malas interpretaciones por el tipo de letra no legible.

El registro de una receta digital lo que pretende es evitar la interpretación por el dependiente, haciendo de esto un proceso automático y seguro

1.4 OBJETIVOS

1.4.1 Objetivo general

Almacenar una receta médica en una tarjeta electrónica para agilizar el proceso de ventas.

1.4.2 Objetivos específicos

- Identificar las necesidades tecnológicas para almacenar la receta digital en las tarjetas inteligentes.
- Identificar los procedimientos necesarios para la venta con recetas médicas.
- Analizar como la generación de una receta digital mejoraría el proceso de ventas en la farmacia.
- Identificar los problemas que conlleva las recetas manuales.
- Proponer soluciones ante posibles problemas que puedan obstaculizar el método de generación de recetas digitales.
- Analizar las ventajas y desventajas de entre la receta manual y la receta digital.
- Definir una comparación de costos de receta manual frente a la receta digital.
- Conocer técnicamente como es el funcionamiento y la programación de una tarjeta inteligente.

¹ Es la manera de administrar de la dosificación de los fármacos (dosis terapéutica)

1.5 JUSTIFICACIÓN

1.5.1 JUSTIFICACIÓN PRÁCTICA

Optimizar procesos operativos en una empresa es uno de los objetivos primordiales para un correcto desempeño laboral y competitividad, la automatización de procedimientos brinda seguridad, confiabilidad, datos oportunos y ordenados, reduce tiempos, costos y mala interpretación de datos.

La generación de una receta digital almacenada en una tarjeta electrónica servirá principalmente para corregir la mala interpretación de datos, agilizar el proceso de ventas y mejorar la atención al cliente y debido a esta innovación dar una mejor imagen a la farmacia.

1.5.2 JUSTIFICACIÓN TEÓRICA

Desarrollo de un sistema de acceso a las tarjetas inteligentes de un abonado a una cadena de farmacias, para la grabación de recetas médicas está basado en un lector de tarjetas Smart Card que se comunica a través de una de un puerto serial, o Usb para el intercambio de información entre estos dispositivos.

En las tarjetas se almacenará la información del usuario, los detalles de la receta médica y los datos del doctor en una base de datos con el fin de realizar posteriores consultas

Los lectores de tarjetas Smart Card obtienen la información de los usuarios que desean comprar en las cadenas farmaceuticas, el momento de introducir la tarjeta inteligente el sistema envía un APDU que es una trama que contiene los datos de la tarjeta y la almacena en un servidor central.

Figura 2: Trama APDU

| | | | | |
|------------|-------------|---------------------------|-------|-----------|
| Encabezado | Instrucción | Longitud del datos = N | Datos | *Checksum |
|------------|-------------|---------------------------|-------|-----------|

La comunicación se lleva a cabo a través de intercambios sucesivos, Comando Respuesta. El controlador transmite un comando al lector y recibe una respuesta desde este, después que el comando ha sido ejecutado. Un nuevo comando puede ser transmitido al ACR30 sólo después que la respuesta del comando previo ha sido recibida.

Los datos almacenados por la tarjeta inteligente serán archivos xml con el propósito de ser leído por cualquier sistema de información.

Los archivos xml serán generados por el lenguaje de programación y primero se grabarán en el disco duro del computador para luego transmitirlo a la tarjeta electrónica.

1.6. Alcance

Con el proyecto “**Recetas médicas almacenadas en tarjetas electrónicas**” se logrará almacenar una receta electrónica con una interfaz de usuario que va de cara al médico, que emite la receta de manera que esta devolverá datos (XML). Esta interfaz además de grabar un archivo xml nos permitirá llevar un historial de recetas emitidas, las cuales se podrán visualizar con reportes clasificados por paciente, por fecha o medicamento según se requiera. Los reportes serán configurados con la herramienta Cristal Report, en un formato de receta médica general.

En el momento que el cliente se acerque a la farmacia a realizar una compra el sistema de facturación de la farmacia, es el encargado de extraer los datos en formato XML. Debido a que el sistema usado en el Grupo DIFARE (Sistema Neptuno) goza de derechos de propiedad intelectual, por tanto no se tiene acceso al código fuente y el archivo grabado en la tarjeta electrónica podrá ser grabado en el computador, con la oportunidad de que los desarrolladores del sistema Neptuno lo puedan utilizar este archivo, para la facturación y registro de las recetas médicas y a su vez este archivo pueda ser visualizado por una interfaz gráfica legible y comprensible para el dependiente de la farmacia. Con esta interfaz usada por los dependientes de la farmacia, se podrá visualizar la receta médica así como reportes clasificados por fechas, clientes, médico o buscarla por número de receta en esta interfaz no se imprimirá la receta; pero si se configurará archivos de impresión de los deferentes reportes que podrán ser impresos.

El archivo XML poseerá los datos del médico, fecha de emisión, lista de medicamentos, datos del paciente y administración de los medicamentos.

El módulo de emisión de recetas médicas no será implementado en la cadena farmacéutica.

1.7. Estudios de factibilidad

1.7.1. Factibilidad Técnica

Para el desarrollo de este proyecto ya se cuenta con la infraestructura tanto en hardware como software, para el sistema operativo en la cadena de farmacias todas funcionan con Windows XP profesional, según las especificaciones técnicas valoradas para el sistema farmacéutico con el que trabajan las farmacias.

Las características del sistema operativo XP Profesional son:

- Windows XP Professional ofrece una experiencia informática fiable a todos los usuarios comerciales.
- Windows XP Professional carga más rápido para los controladores de dispositivos.
- Elimina la mayoría de los escenarios que obligaban a los usuarios finales a reiniciar los equipos, además, una gran parte de las instalaciones de software no requieren reiniciar.
- Las estructuras de los datos importantes del núcleo son de sólo lectura, por lo que los controladores y las aplicaciones no pueden corromperlas. Todos los códigos de controladores de dispositivos son de sólo lectura y con protección de página.
- Permite a los usuarios crear sesiones virtuales en sus equipos de escritorio mediante el protocolo de escritorio remoto (RDP, Remote Desktop Protocol) de Microsoft.
- Almacén protegido para la información de contraseñas. Permite que los usuarios escriban una vez sus nombres de usuario y contraseñas, y que a partir de entonces el sistema proporcione esta información en las visitas siguientes.

- Hace posible la visualización sin conexión de páginas Web con gráficos enteras.
- Proporciona lo último en administración de energía y soporte Plug and Play.
- Proporciona acceso protegido, además de mejoras en el rendimiento, para redes inalámbricas.
- Guían al usuario por los pasos de configuración del acceso remoto en una red o una red privada virtual (VPN).
- Proporciona una consola de línea de comandos para que los administradores inicien y detengan servicios, den formato a unidades, lean y escriban datos en una unidad local y realicen otras muchas tareas administrativas.

La base de datos con la que se trabaja el sistema farmacéutico (NEPTUNO) es SQL Express 2005 debido a que es compatible con este sistema desarrollado en Visual Basic Versión 6.0 herramientas que serán las que se usen en este proyecto

Las características de SQL Express 2005 son:

- Una base de datos robusta para crear aplicaciones dinámicas.
- Fuerte soporte de XML.
- Herramientas y características para mejorar la gestión y facilidad de uso.
- Instalación y setup sencillos.
- Escalabilidad y rendimientos.
- Gran integración con Visual Studio 2005.
- Robusta seguridad.
- Procedimientos almacenados.
- Vistas.
- Triggers.
- Cursores.
- Índices ampliados.
- Aislamiento a nivel de snapshot.
- Optimizador avanzado de consultas.
- Compatibilidad con T-SQL.
- Tipo XML nativo.
- Compatibilidad con datos estructurados y semiestructurados.
- Compatibilidad con XQuery.

- Compatibilidad con esquemas XML.

El lenguaje de programación será Visual Basic 6.0 ya que el sistema actual con el que funciona las cadenas farmacéuticas está desarrollado en este lenguaje. Las características de este lenguaje son:

- Es un lenguaje RAD (*Aplicación de Rápido Desarrollo*).
- Posee una curva de aprendizaje muy rápida.
- Integra el diseño e implementación de formularios de Windows.
- Permite usar con suma facilidad la plataforma de los sistemas Windows dado que tiene acceso prácticamente total a la API de Windows incluidas librerías actuales.
- El código en Visual Basic es fácilmente migrable a otros lenguajes.
- Es un lenguaje muy extendido por lo que resulta fácil encontrar información, documentación y fuentes para los proyectos.
- Fácilmente extensible mediante librerías DLL y componentes ActiveX de otros lenguajes.
- Posibilidad de añadir soporte para ejecución de scripts, VBScript o JScript, en las aplicaciones mediante Microsoft Script Control.
- Acceso a la API multimedia de DirectX (versiones 7 y 8). También está disponible, de forma no oficial, un componente para trabajar con OpenGL 1.1
- Existe una versión integrada en las aplicaciones de Office, versiones tanto Windows como Mac, que permite programar macros para extender y automatizar funcionalidades en documentos como por ejemplo una hoja de cálculo de EXCEL o una base de datos ACCESS.
- Es un entorno perfecto para realizar pequeños prototipos rápidos de ideas.

1.7.2 Factibilidad Operativa

Este proyecto es factible operativamente ya que el proceso de recuperación de la receta médica, solo consiste en introducir la tarjeta inteligente del abonado y el sistema es el encargado de procesar la receta médica. El dependiente hará el mismo proceso que lo realiza cuando un cliente viene con su tarjeta de afiliado.

Para recopilar información sobre el proceso de venta con receta médicas se elaborara una encuesta al personal al personal que trabaja en farmacia y a la observación del proceso de venta con tarjeta de descuento todo esto más la experiencia de trabajo como implementador del sistema farmacéutico y como usuario de la tarjeta FARMAVIP.

Cabe mencionar que se cuenta con la autorización del Jefe de Servicios Región Sierra para realizar el proceso de recolección de datos para el desarrollo del proyecto.

1.7.3 Factibilidad Económica

Los costos de este sistema se distribuyen de la siguiente manera:

Tabla 3: Factibilidad Económica

| Descripción | Cantidad | Costo Mensual | Costo | Costo total |
|---------------------------------|----------|---------------|---------|-------------|
| Programación | 1 | n/a | \$ 300 | \$ 300 |
| Lector de tarjetas inteligentes | 40 | n/a | \$86.00 | \$ 3.440 |
| Equipo Nuevo (computadoras) | 1 | n/a | \$ 600 | \$ 600 |
| Tarjetas inteligentes | 5 | n/a | \$ 2.00 | \$ 10 |
| Capacitación (uso de Internet) | 3 meses | n/a | \$ 70 | \$ 210 |

- Programación.- Costo de que conlleva la programación del módulo del sistema
- Lector de tarjetas inteligentes.- Es el dispositivo que permite grabar y leer las recetas de la tarjeta inteligente indispensable para la emisión de recetas médicas por parte del médico.

- Equipo Nuevo – computadoras marca Dell con Windows XP professional SP2, Office 2003 Small Bussines con tres años de garantía.
- Tarjeta inteligente.- Consiste en el dispositivo donde se almacenará la receta médica.
- Capacitación.- Es el costo que conlleva las consultas en Internet para la capacitación de esta tecnología.

Tabla 4: Costo del proyecto

| Costo de: | Valor |
|-----------------------------------|-----------------|
| Desarrollo | \$ 300.00 |
| Lectores de tarjetas inteligentes | \$ 3.440 |
| Equipo Nuevo (computadoras) | \$ 600.00 |
| Tarjetas inteligentes | \$ 10.00 |
| Capacitación (uso de Internet) | \$ 210.00 |
| total | \$ 4.560 |

Costo Fijo = \$ 4474.00

Costo Recurrente = \$ 86.00

Costo del proyecto = \$ 4.560

Costos del sistema actual

En una farmacia promedio la facturación con recetas médicas según los administradores de las farmacias son: de 18% de clientes diariamente y con un valor promedio de \$30 cada receta, por los motivos mencionados anteriormente se pueden perder 5% de las ventas con recetas médicas promedio por día. El valor sugerido por la cadena ECUAFARMACIAS para la ganancia en un producto es el 18%. Otro dato importante es que una farmacia vende diariamente un promedio de \$600.

Tabla 5: Pérdida de Ventas con recetas médicas por día

| Descripción | Ventas diarias | Ventas con recetas Médicas | Ventas diarias con recetas médicas | Pérdida Diaria venta diaria % 5 |
|-------------------------------|-----------------------|-----------------------------------|---|--|
| 1 farmacia | \$ 600 | % 18 | \$ 108 | \$ 30 |
| Cadena Farmacys(30 farmacias) | \$18.000 | % 18 | \$ 3.240 | \$ 900 |
| | | | Total de pérdida = | \$ 162 |

Con estos datos podemos concluir que aproximadamente las pérdidas de ventas por receta son de 162 dólares diariamente en la cadena de farmacias en la ciudad de Quito

Beneficios

El sistema proveerá de una mejor forma de manejar las recetas médicas en una farmacia, al tratar el tema digitalmente se reduce notablemente la mala interpretación de los nombres de los medicamentos principalmente por el personal novato de una farmacia, el apuro de atender a todos los clientes. Al ser guardada esta receta en la tarjeta de afiliación del cliente es mas difícil que se pierda y el proceso de interpretación por el vendedor se obviará pasando así a ser parte del sistema de facturación. Al ser un proceso sistematizado se reducirán los tiempos de atención al cliente brindando así una mejor imagen y comodidad a las personas que compren en las farmacias.

Beneficios Tangibles:

- Evitar las pérdidas y malos despachos de las recetas médicas.
- Reducción de tiempos en la atención al cliente.
- Un mejor y ordenado registro de recetas médicas.

- Reportes reales de emisión de recetas médicas por médico.
- Comodidad al vendedor de mostrador y fácil despacho de medicamentos.

Beneficios Intangibles:

- Mejora la imagen de la empresa.
- Ofrecer al cliente una mejor calidad de servicio.
- Comodidad para el cliente al evitar estar llevando la receta médica con el riesgo de extraviarla.
- Atención ágil y rápida al cliente.

Relación Costo-Beneficio

Tabla 6: Recuperación de la inversión

| Descripción | Costo del sistema | Pérdida diaria |
|-------------|-------------------|----------------|
| 1 día | \$ 4.474 | \$ 162 |
| 28 días | \$ 0 | \$ 4.536 |

El análisis costo beneficio nos muestra que es totalmente factible la implementación de este proyecto y la inversión inicial se la recuperará en aproximadamente 28 días, aprovechando de este modo el tiempo de atención del cliente y disminuyendo la pérdida de ventas con recetas médicas, se mejorarán los procesos que conlleva la venta de recetas médicas como son la auditoría del CONSEP.

2. MARCO DE REFERENCIA

2.1 MARCO TEÓRICO

Para el desarrollo de este proyecto se utilizará el Modelo Unificado Racional de desarrollo de sistemas en el cual se basa en varias iteraciones para así lograr la identificación concreta y clara del problema.

En la primera, el modelo de negocios y requisitos se elaborará los modelos de casos de uso y el modelo de objetos de negocios. Estos datos serán obtenidos por medio de la observación en el lugar de trabajo así como entrevistas al personal y los cuestionarios, resaltando los puntos más importantes y los requerimientos necesarios para la solución que se plantea.

En la fase de análisis y diseño los procesos que realizará el sistema ya sean la interactividad con el usuario, las funciones del sistema, la clasificación de los objetos, la interactividad entre capas serán modelados para un mejor entendimiento de la funcionalidad y las características que necesita el sistema y será diagramado con el Lenguaje de Modelado para graficar aspectos conceptuales tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

La implementación del sistema se la hará basada en las herramientas de desarrollo elegidas en la factibilidad técnica del proyecto. En esta fase se programará el sistema cumpliendo los requerimientos obtenidos en el diseño del sistema, teniendo en cuenta los diagramas de objetos, base de datos, clases y procesos, presentando al cliente avances a modo de prototipos de manera que nos ayude a llegar a cumplir los requerimientos que el cliente necesita.

Se realizará las pruebas respectivas con el fin de pulir la funcionalidad del sistema y que el producto final sea un sistema de calidad.

Elaborando el modelo de despliegue se emitirá un plan de desarrollo de software y se evaluará si es necesaria otra iteración.

El resultado obtenido por la interfaz de usuario, será la receta médica de una forma digital la cual mediante un lector de tarjetas inteligentes se guardará esta información.

Se utilizarán APIs de programación para que los programas que permitan la interactividad del computador con la tarjeta inteligente, de este modo mediante una interfaz de usuario se programará la tarjeta inteligente. El modelo de programación que utilizan las tarjetas inteligentes está basado en protocolos de petición-respuesta. La tarjeta (su software) expone una serie de comandos que pueden ser invocados. Estos comandos interactúan con los ficheros que subyacen a cada aplicación de la tarjeta y proporcionan un resultado. Desde el terminal se invocan estos comandos a través de cualquiera de las APIs antes descritas componiendo APDUs (*Application Protocol Data Unit* - comandos con parámetros) que son enviados a la tarjeta para que ésta responda.

Para facilitar la programación del sistema se utilizará el método de programación orientada a objetos, que consiste en identificar los objetos que van a ser utilizados por el sistema y programarlos con sus diferentes funciones. Con estas técnicas los objetos tendrán las propiedades de herencia, modularidad, polimorfismo y encapsulamiento de manera de poder reutilizarlos según conveniencia del sistema y la facilidad de programación.

Para la lectura de la tarjeta inteligente se necesitará un lector de contacto de tarjetas inteligentes y este a su vez tiene dos maneras de conectarse al computador, los dispositivos que se usan en las farmacias son lectores USB pudiendo a su vez funcionar también por el puerto serial. Este dispositivo no solo funciona para leer, sino también para escribir en una tarjeta inteligente, de este modo la aplicación que se realizará, podrá manipular los datos grabados en la tarjeta inteligente según sea el caso.

Las tarjetas inteligentes que se van a usar son tarjetas que no poseen un chip incorporado, sino son tarjetas inteligentes de memoria y contienen sólo componentes de memoria no volátil y posiblemente alguna lógica de seguridad. Estas se introducen en el lector de tarjetas y la interfaz de usuario será la encargada de por

medio del protocolo de comunicación APDU gravará el archivo XML, en la memoria de la tarjeta inteligente.

Los archivos XML gracias a ser un metalenguaje universal y comprensible para la mayoría de lenguajes de programación con que se desarrollan los sistemas actuales, de este modo el archivo que generará el sistema desarrollado será capaz de ser leído e interpretado, por el sistema utilizado actualmente en las farmacias. Este archivo constará de etiquetas que identificarán el médico que emite la receta médica, etiquetas que identificarán el listado de medicamentos, etiquetas para saber la fecha de emisión, etiquetas que identifican el paciente que va a tomar la medicina y etiquetas que indicarán la posología de los medicamentos.

2.2 MARCO CONCEPTUAL

2.2.1 APIs de programación

Es un conjunto de especificaciones para la integración de tarjetas inteligentes en ordenadores personales. En particular se define un API de programación que permite a los desarrolladores trabajar de forma uniforme con lectores de tarjetas de distintos fabricantes (que cumplan con la especificación).

El *API* de *PC/SC* está incorporada en sistemas Microsoft Windows 200x/XP y disponible también Microsoft Windows NT/9x. También hay una implementación libre, de código abierto, llamada **PC/SC Lite** (proyecto MUSCLE) para sistemas operativos GNU Linux.

El API de programación es sencillo y proporciona funcionalidades simples. A continuación se enumeran las principales para dar una visión general del modo en que se programa usando este API:

- *SCardEstablishContext* y *SCardReleaseContext* (inicialización y liberación de recursos)
- *SCardConnect*, *SCardReconnect* y *SCardDisconnect* (conexión y desconexión a una tarjeta en un lector)
- *SCardListReaderGroups*, *SCardListReaders*, ... (exploración de lectores conectados)

- *SCardGetAttrib* (obtención de propiedades de los lectores conectados)
- *SCardTransmit* (envío de comandos APDU's a una tarjeta conectada)

La especificación se divide en 9 partes que contienen los requisitos detallados de inter-operabilidad de dispositivos compatibles, información de diseño, interfaces de programación, etc.

- Parte 1. Introducción y visión general de la arquitectura.
- Parte 2. Requisitos de interoperabilidad para las tarjetas y los lectores.
- Parte 3. Requisitos de interoperabilidad para los lectores.
- Parte 4. Consideraciones de diseño de los lectores e información de referencia.
- Parte 5. Definición de la interfaz del *Resource Manager*.
- Parte 6. Definición de la interfaz del *Service Provider*.
- Parte 7. Consideraciones de diseño para el desarrollo de aplicaciones.
- Parte 8. Recomendación para la implementación de servicios de seguridad y privacidad con tarjetas inteligentes.
- Parte 9. Lectores con capacidades extendidas.

2.2.2 RUP

Es el proceso de desarrollo de software diseñado por IBM. RUP divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades. En la Figura muestra cómo varía el esfuerzo asociado a las disciplinas según la fase en la que se encuentre el proyecto RUP.

Las primeras iteraciones (en las fases de Inicio y Elaboración) se enfocan hacia la comprensión del problema y la tecnología, la delimitación del ámbito del proyecto, la eliminación de los riesgos críticos, y al establecimiento de una base de la arquitectura.

Durante la fase de inicio las iteraciones hacen mayor énfasis en actividades de modelado del negocio y de requerimientos.

En la fase de elaboración, las iteraciones se orientan al desarrollo de la arquitectura, abarcan más los flujos de trabajo de requerimientos, modelo de negocios (refinamiento), análisis, diseño y una parte de implementación orientado a la arquitectura.

En la fase de construcción, se lleva a cabo la construcción del producto por medio de una serie de iteraciones.

Para cada iteración se selecciona algunos Casos de Uso, se refina su análisis y diseño y se procede a su implementación y pruebas. Se realiza una pequeña cascada para cada ciclo. Se realizan tantas iteraciones hasta que se termine la implementación de la nueva versión del producto.

En la fase de transición se pretende garantizar que se tiene un producto preparado para su entrega a la comunidad de usuarios.

Como se puede observar en cada fase participan todas las disciplinas, pero que dependiendo de la fase el esfuerzo dedicado a una disciplina varía.

2.2.3. Modelado UML

UML, por sus siglas en inglés, (*Unified Modeling Language*) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (Object Management Group). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales, tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

Es importante resaltar que UML es un "lenguaje de modelado" para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

Se puede aplicar en el desarrollo de software entregando gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como el Proceso Unificado Racional o RUP), pero no especifica en sí mismo qué metodología o proceso usar.

UML no puede compararse con la programación estructurada, pues UML significa Lenguaje Unificado de Modelado, no es programación, solo se diagrama la realidad de una utilización en un requerimiento. Mientras que, programación estructurada, es una forma de programar como lo es la orientación a objetos, sin embargo, la programación orientada a objetos viene siendo un complemento perfecto de UML, pero no por eso se toma UML sólo para lenguajes orientados a objetos.

2.2.4 Programación Orientada a Objetos

La programación orientada a objetos expresa un programa como un conjunto de estos objetos, que colaboran entre ellos para realizar tareas. Esto permite hacer los programas y módulos más fáciles de escribir, mantener, y reutilizar.

De aquella forma, un objeto contiene toda la información que permite definirlo e identificarlo frente a otros objetos pertenecientes a otras clases e incluso frente a objetos de una misma clase, al poder tener valores bien diferenciados en sus atributos. A su vez, los objetos disponen de mecanismos de interacción llamados métodos que favorecen la comunicación entre ellos. Esta comunicación favorece a su vez el cambio de estado en los propios objetos. Esta característica lleva a tratarlos como unidades indivisibles, en las que no se separan ni deben separarse el estado y el comportamiento.

Los métodos (comportamiento) y atributos (estado) están estrechamente relacionados por la propiedad de conjunto. Esta propiedad destaca que una clase requiere de métodos para poder tratar los atributos con los que cuenta. El programador debe pensar indistintamente en ambos conceptos, sin separar ni darle mayor importancia a ninguno de ellos. Hacerlo podría producir el hábito erróneo de crear clases contenedoras de información por un lado y clases con métodos que manejen a las primeras por el otro. De esta manera se estaría realizando una

programación estructurada camuflada en un lenguaje de programación orientado a objetos.

2.2.5 Tarjeta Inteligente

Una **tarjeta inteligente (smart card)**, o tarjeta con circuito integrado (TCI), es cualquier tarjeta del tamaño de un bolsillo con circuitos integrados que permiten la ejecución de cierta lógica programada. Aunque existe un diverso rango de aplicaciones, hay dos categorías principales de TCI. Las Tarjetas de memoria contienen sólo componentes de memoria no volátil y posiblemente alguna lógica de seguridad. Las tarjetas microprocesadoras contienen memoria y microprocesadores.

La percepción estándar de una *tarjeta inteligente* es una tarjeta microprocesadora de las dimensiones de una tarjeta de crédito (o más pequeña, como por ejemplo, tarjetas SIM o GSM) con varias propiedades especiales (ej. un procesador criptográfico seguro, sistema de archivos seguro, características legibles por humanos) y es capaz de proveer servicios de seguridad (ej. confidencialidad de la información en la memoria).

2.3 Marco Temporal/Espacial

El tiempo de vida de este sistema durará según dure el sistema general de administración farmacéutica (NEPTUNO) que usan en las farmacias, ya que esta enfocado a dar un servicio extra a este sistema con la ventaja de ser útil para cualquier otro lenguaje que maneje archivos XML. En ese caso conforme avance la tecnología y los requerimientos del usuario el sistema podrá seguir siendo actualizado.

2.4 Marco legal

El manejo de recetas médicas implica regirse a ciertas normas impuestas por el Ministerio de salud, este documento tiene que ser registrado en la farmacia si el fármaco es un psicotrópico y el sistema debe ser capaz de llevar un histórico por paciente, fecha de venta ya que estos medicamentos en algunos casos son de medicación continua y son prescritos cada cierto tiempo, también debe ser registrado el nombre del médico que emitió la receta. En el caso de los psicotrópicos

una persona representante del CONSEP acude cada mes a la farmacia para revisar la venta de los psicotrópicos eso lo hace comparando el inventario de la farmacia con las recetas médicas físicamente.

Siendo una normativa del CONSEP el documento físico de la receta médica este nos servirá de respaldo a los datos guardados en la tarjeta inteligente ya que al tratarse de medicamentos restringidos, con lleva una mayor responsabilidad el despacho de estos, manejándolo de esta manera habrá menos errores en la interpretación de las recetas médicas.

3. METODOLOGÍA

3.1 Metodología de la Investigación

3.1.1 Unidad de Análisis

Para el desarrollo del tema **“Recetas médicas almacenadas en tarjetas inteligentes”** se tomará como unidad de análisis la cadena de farmacias Farmacys ya que esta cadena usa como tarjeta de crédito y descuentos una tarjeta inteligente en la cual contiene el tipo de cliente y sus datos.

3.1.2 Tipos de Investigación.

Para este proyecto se realizará un tipo de investigación exploratorio, de manera de ir destacando según los sucesos observados, cuales son los de mayor relevancia para el desarrollo del proyecto.

El propósito de identificar los hechos más relevantes en este caso la problemática que conlleva la venta de medicamentos, con recetas médicas hechas a mano, sería con el fin de dar soluciones puntuales e ir definiendo como se pueden solucionar, con el método de emisión de una receta con tarjetas inteligentes.

Explorando los procesos de ventas en las farmacias concluiremos como se llevan a cabo los procesos e identificaremos sus ventajas y desventajas.

El documento final de este proceso será en detalle los requerimientos exactos que se necesitan para solucionar la problemática y también los procesos necesarios que necesitará para solucionar el problema. Este documento será revisado con el cliente para que desde un principio quede la idea clara.

3.2. Metodología Informática

3.2.1 Metodología Orientada a Objetos

El desarrollo de este proyecto utilizará la metodología orientada a objetos para de este modo encapsular las diferentes funciones del sistema de una forma que puedan ser reutilizados y la programación sea mas ordenada y fácil de entender.

Una vez identificados los componentes que formarán estos serán programados en objetos de esta manera facilitará la comprensión del funcionamiento y principalmente las relaciones e interacciones entre ellos.

Con esta metodología orientada a objetos si se quisiera ampliar y aumentar funciones sería mucho más fácil con esta metodología ya que si fuera necesario se aumentaría más objetos o a su vez se aumentaría su funcionabilidad, a diferencia de la programación estructurada que tendría que analizarse nuevamente todo el código y así ir aumentando la funcionabilidad.

Con esta metodología se pretende realizar un modelo del sistema de información considerando que su estructura interna está formada por un conjunto de objetos que interactúan entre sí. Cada objeto tiene unas propiedades y un comportamiento que representan respectivamente la estructura de la información y su procesamiento.

En esta fase se entregaran los diagramas que serán necesarios para construir y programar el sistema, así como el diagrama de base de datos, diagramas de clases, diagramas de flujo y diagramas de actividades.

3.2.2. Procesos de Ingeniería

Para este proyecto se usará el proceso de desarrollo RUP que va de la mano son el lenguaje de modelamiento UML, que es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. UML entrega una forma de modelar cosas conceptuales como lo son procesos de negocio y funciones de sistema, además de cosas concretas como lo son escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables.

Esta metodología va de la mano con la metodología Orientada a Objetos. De este modo todos los objetos obtenidos de la recopilación de información

serán diagramados según la normativa del modelado UML así también las relaciones entre ellos, la clasificación entre objetos y sus métodos.

Con estos diagramas nos ayudarán a interpretar mejor lo que se necesita en este proyecto.

4. PROCESO DE DESARROLLO

En este proyecto se dará la solución informática a los problemas planteados al inicio de manera de optimizar y mejorar los procesos para la venta de medicamentos con recetas médicas aprovechando la tecnología que utilizan las tarjetas inteligentes ya que es el medio que la cadena de farmacias Farmacys de la cadena Ecuafarmacias que utiliza para dar crédito y descuentos a sus clientes.

Este proyecto se basará en la metodología RUP para dar una solución óptima y que cumpla las necesidades del cliente.

4.1 Fase de Inicio

Los documentos generados en esta etapa son: Documento de Visión, Listado de Riesgos, Diagramas de casos de uso del Negocio, diagramas de actividades, los mismos que son distribuidos en los siguientes flujos:

4.1.1 Flujo Modelado del Negocio

El documento que se elaborará para esta fase de trabajo será el Documento de Visión, el cual nos muestra el modo de funcionamiento y los procesos que se realizan actualmente para la venta de medicamentos con receta médica para ver de una manera global y tener una idea de cómo podemos mejorar los procesos.

El propósito de éste documento es recoger, analizar y definir las necesidades de alto nivel y las características del sistema de gestión de venta de medicamentos con recetas médicas grabadas en tarjetas inteligentes. El documento se centra en la funcionalidad requerida por los usuarios finales.

Esta funcionalidad se basa principalmente en la gestión de los procesos que se efectúan en una farmacia para la venta de medicamentos con recetas médicas.

Los detalles de cómo el sistema cubre los requerimientos se pueden observar en la especificación de los casos de uso y otros documentos adicionales.

El documento “Visión” se ocupa, como ya se ha apuntado, del sistema de venta de medicamentos con recetas médicas grabadas en tarjetas inteligentes.

El sistema permitirá a los encargados de las farmacias controlar todo lo relativo a la venta, registro y manejo de los procesos que con lleva la venta con recetas médicas. Además, el registro de la receta médica en la tarjeta del afiliado por parte del médico.

4.1.2 Flujo de Requisitos

El documento para esta fase será el Documento de Listado de Riesgos, aquí identificaremos los riesgos de manera que no se presente en una fase posterior del proyecto para evitar retrasos o inclusive desistir del proyecto por riesgos no controlados o no detectados. Los riesgos identificados oportunamente nos ayudaran a un desempeño rápido sin demoras, al cumplimiento de los objetivos y la satisfacción del cliente.

4.1.3 Flujo Análisis

El entregable en este flujo de trabajo son los Diagramas de Caso de Uso del Negocio, grafica los procesos de negocio del sistema actual así también los actores que interactúan con esos proceso y las relaciones entre ellos.

Con estos gráficos podemos tener una idea mas específica de del funcionamiento del sistema actual de manera de familiarizarnos con las posibles soluciones a los problemas que puedan observarse.

Actores y Casos de Uso del Negocio

Los casos de uso representan los procesos y los actores que intervienen en el modelo del negocio, actualmente en el sistema sin referirse al sistema planteado o a las soluciones que se proponen.

Actores del Negocio

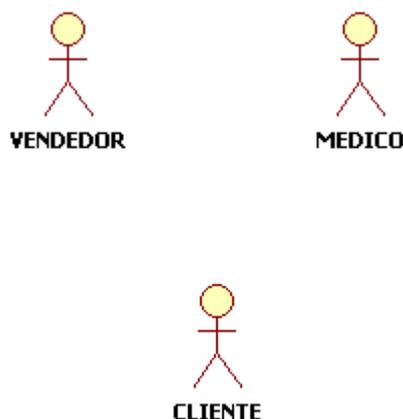


Figura 3: Actores del Modelo del Negocio

Tabla 7 : Descripción de los Actores del negocio

| Nombre | Descripción | Responsabilidades |
|--------------------|---|--|
| Vendedor mostrador | Encargado de vender, despachar medicamentos y registrar la receta médica en el sistema y registrar el médico que emitió la receta en el sistema | Realiza: Despacha los productos de la receta médica Factura medicamentos. Registra la receta en el sistema Registra médico en el sistema |
| Médico | Encargado de emitir la receta médica y guardar en la tarjeta del afiliado | Realiza la receta médica Graba en la tarjeta inteligente |
| Cliente | Interactúa con el vendedor, entrega recela médica, pide anular factura | Entrega receta médica Solicita devoluciones |

Casos de Uso del Negocio

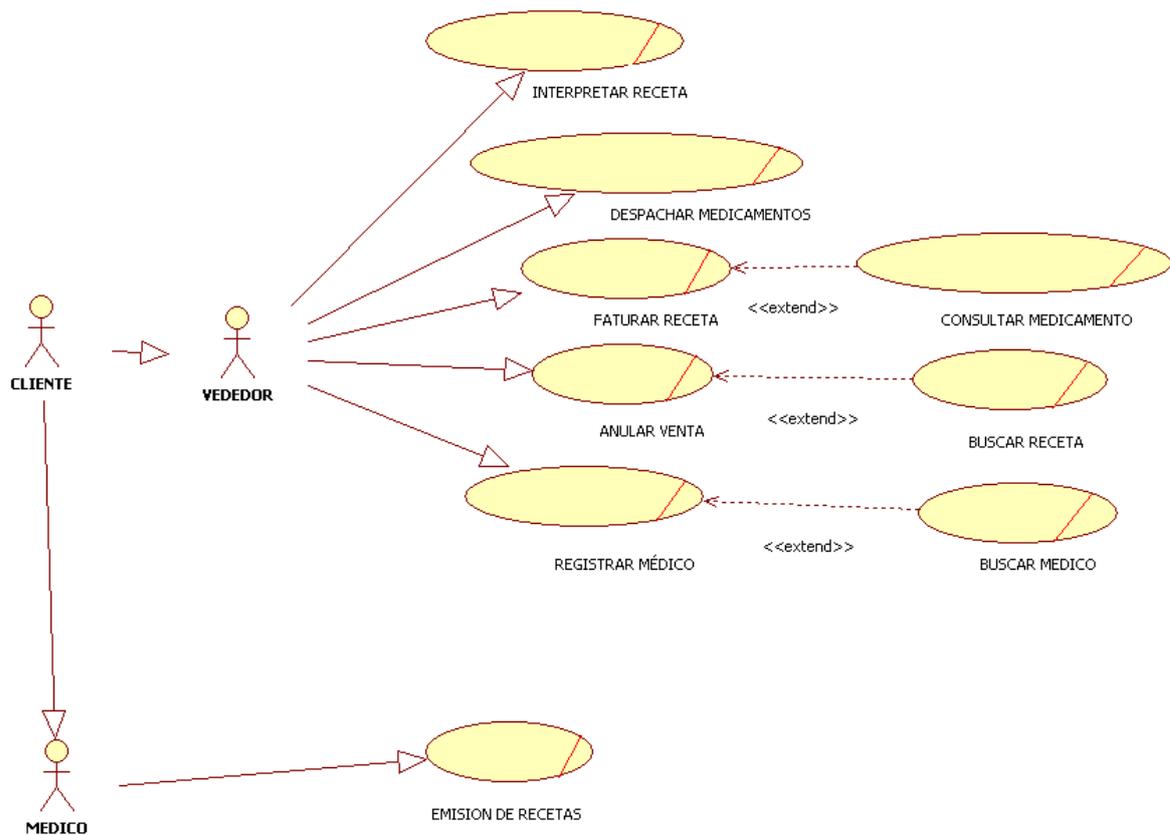


Figura 4 Casos de Uso del Negocio

4.1.4 Flujo Requisitos (Iteración 2)

En este flujo de trabajo se realiza el Listado de Riesgos, en esta fase se identifican nuevos riesgos y posibles planes de contingencia.

4.1.5 Flujo Diseño (Iteración 2)

En esta fase se define el Diagrama de Actividades, mostrando así el flujo de trabajo de cada actor en el negocio de acuerdo al caso de uso; un flujo de trabajo muestra las actividades que tiene cada actor en el sistema actual.

Diagrama de Actividades para el Caso de Uso: Interpretar receta médica

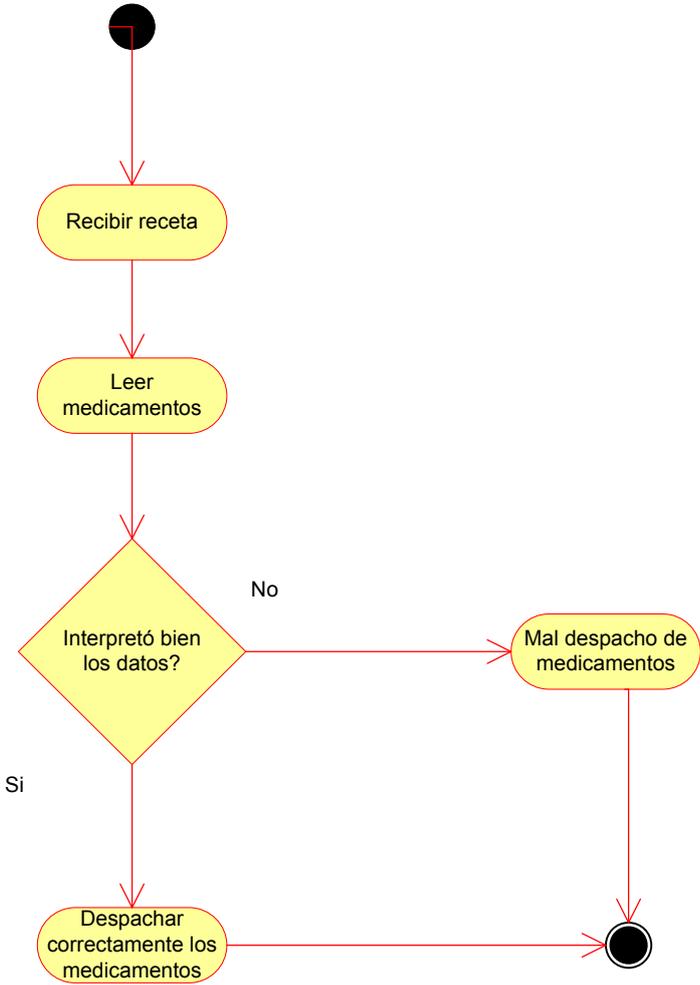


Figura 5: Diagrama de Actividades: Interpretar receta médica

Diagrama de Actividades para el Caso de Uso: Despachar medicamentos

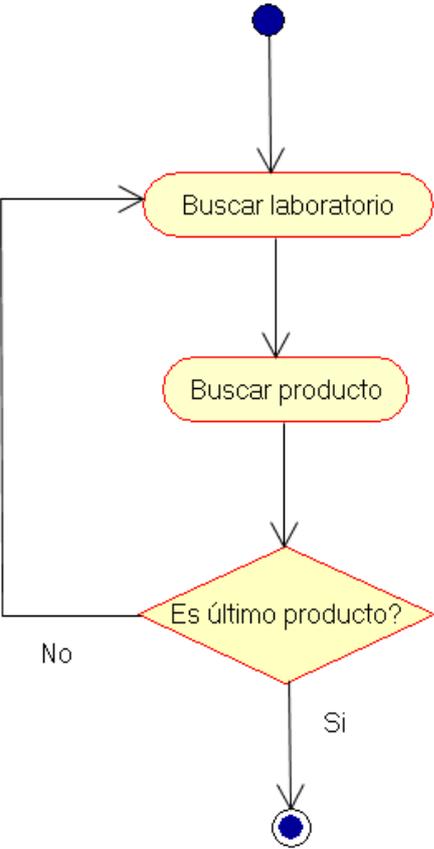


Figura 6: Diagrama de Actividades: Despachar medicamentos

Diagrama de Actividades para el Caso de Uso: Facturar receta



Figura 7: Diagrama de Actividades: Facturar receta

Diagrama de Actividades para el Caso de Uso: Anular venta

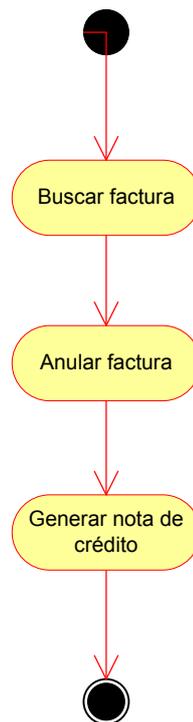


Figura 8: Diagrama de Actividades: Generar Actas de Entrega-Recepción

Diagrama de Actividades para el Caso de Uso: Registrar médico



Figura 9: Diagrama de Actividades: Registrar médico

Diagrama de Actividades para el Caso de Uso: Emitir receta médica



Figura 10: Diagrama de Actividades: Emitir receta médica

4.2 Fase de Elaboración

Los artefactos a obtenerse son: Detalle de los casos de uso del negocio, casos de uso del sistema, especificación de los casos de uso del sistema, diagramas de secuencia, diagrama de despliegue.

4.2.1 Flujo Modelado del Negocio

En esta fase de trabajo es el artefacto Detalle de los Casos de Uso del Negocio, los casos de uso nos permite una mejor visión de las actividades de los procesos, se que se explica de una forma mas específica la lógica del negocio.

Detalle del Caso de Uso del Negocio: Interpretar receta médica

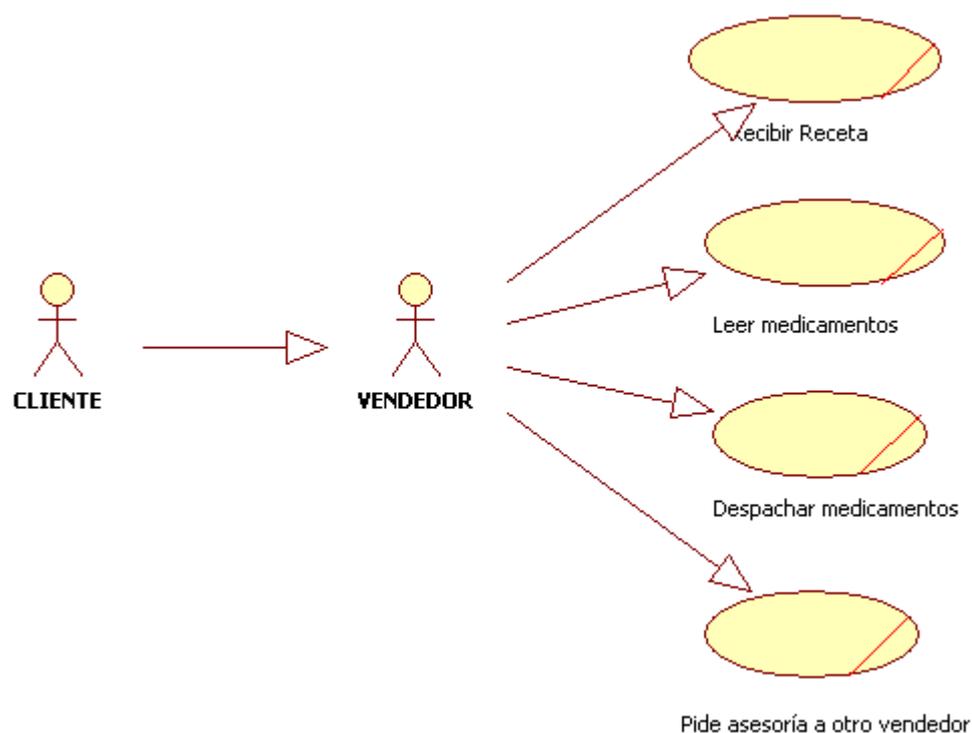


Figura 11: Detalle del Caso de uso del Negocio: Interpretar receta médica

Descripción:

El vendedor de mostrador atiende al cliente recibiendo la receta médica entonces lee cada medicamento, si no lo entiende consulta con un vendedor mas experimentado para luego despachar el medicamento.

Detalle del Caso de Uso del Negocio: Despachar medicamentos

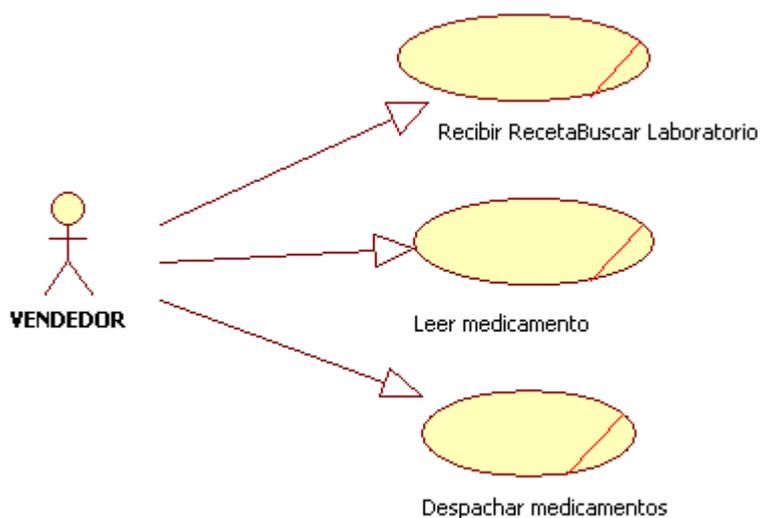


Figura 12: Detalle del Caso de uso del Negocio: Despachar medicamentos

Descripción:

El vendedor de mostrador debe ubicar el laboratorio al cual pertenece el medicamento, para luego ubicarlo de esta manera, completar la receta médica y despachar los medicamentos.

Detalle del Caso de Uso del Negocio: Facturar receta

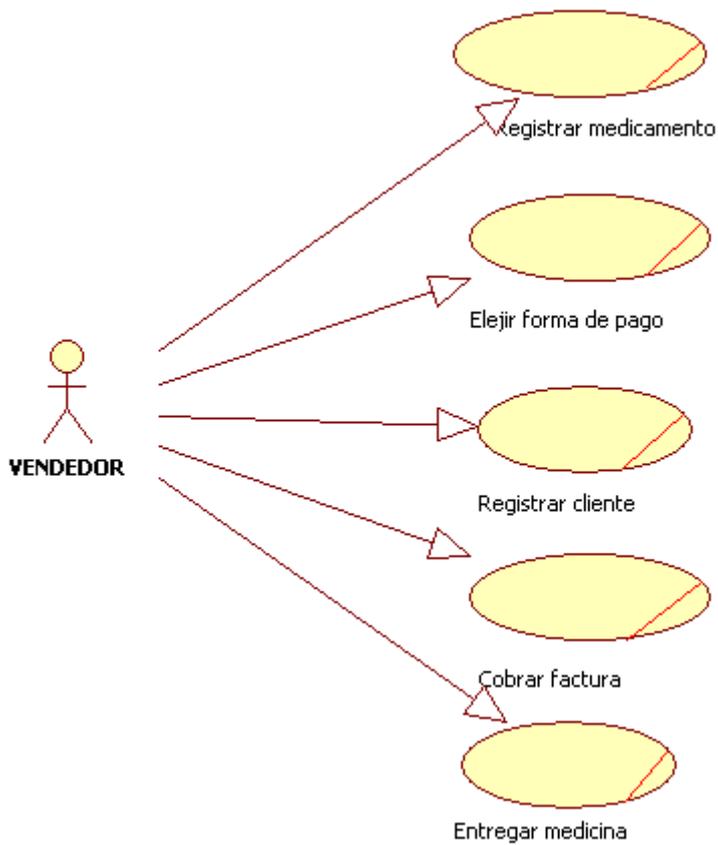


Figura 13: Detalle del Caso de uso del Negocio: Facturar receta

Descripción:

El vendedor de mostrador cuando tiene todos los medicamentos de la receta a facturar los debe registrar al sistema, si el cliente va a pagar a crédito, con tarjeta de crédito, nota de crédito también tiene que registrarse en el sistema.

Detalle del Caso de Uso del Negocio: Anular venta

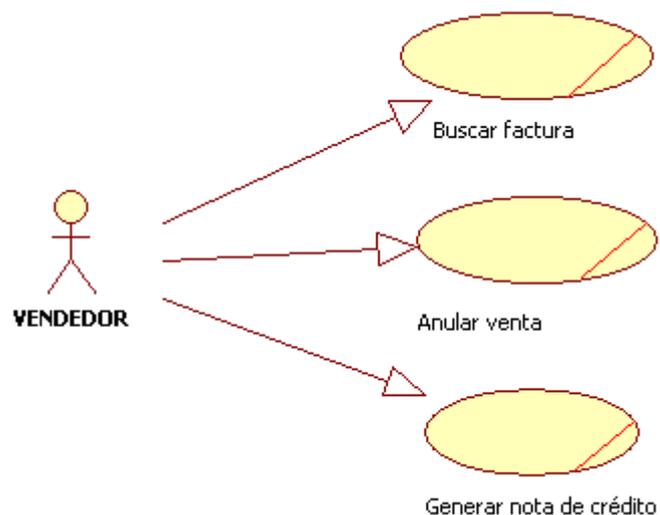


Figura 14: Detalle del Caso de uso del Negocio: Anular venta

Descripción:

El vendedor mostrador debe buscar la receta que se requiere anular ya sea por número de factura o buscar por cliente, cuando se encuentra la factura se procede a hacer la anulación y a generar una nota de crédito la cual se la puede usar para comprar otros productos o cambiarla por efectivo.

Detalle del Caso de Uso del Negocio: Registrar médico

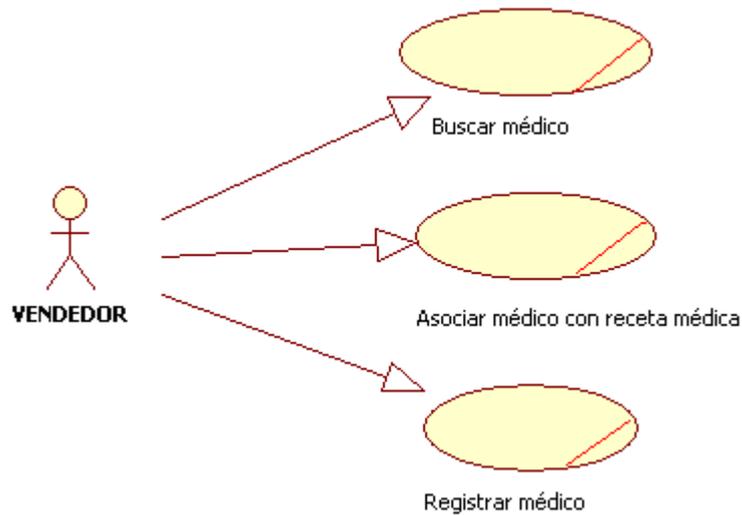


Figura 15: Detalle del Caso de uso del Negocio: Registrar médico

Descripción:

Una vez realizada la factura con receta médica se tiene que registrar el médico, para reconocerle con los incentivos acordados con el médico correspondiente.

Detalle del Caso de Uso del Negocio: Emitir receta médica

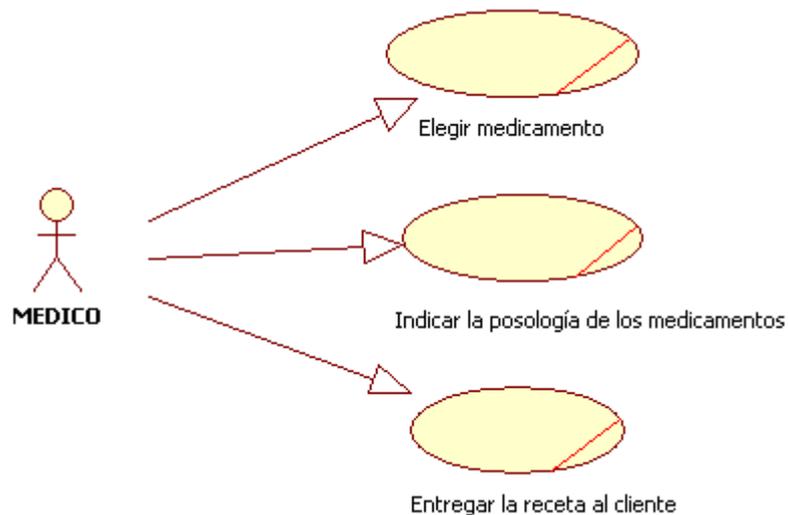


Figura 16: Detalle del Caso de uso del Negocio: Emitir receta médica

Descripción:

El médico una vez de diagnosticar a su paciente elije los medicamentos adecuados para realizar la receta médica.

4.2.2 Flujo Requisitos

En este flujo de trabajo se realizarán los diagramas de casos de uso del sistema, como también su descripción.

Actores del Sistema

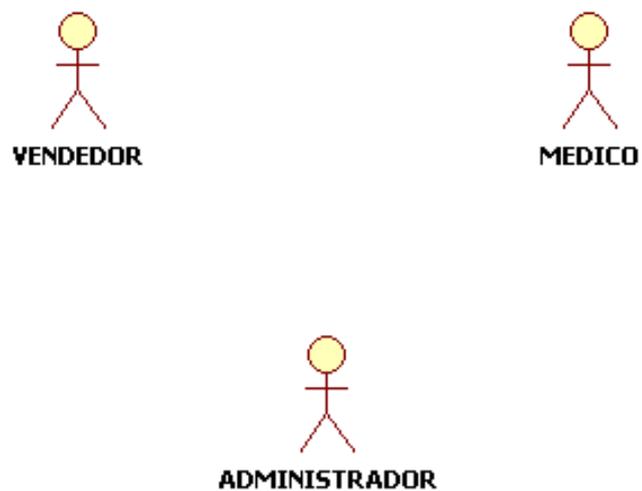


Figura 17: Actores del Sistema

Tabla 8: Descripción de los Actores del Sistema

| Usuarios | Descripción |
|-----------------|--|
| Vendedor | Es el usuario que realizará las ventas y la interpretación de la receta médica. |
| Médico | Este usuario es el encargado de emitir la receta médica y grabar la receta en la tarjeta del afiliado, también hacer consultas de recetas emitidas anteriormente. |
| Administrador | Es el encargado de hacer la diferentes configuraciones para que el sistema sea funcional así como registrar el nombre del médico en el sistema, configurar el número de las recetas médicas e instalar el sistema. |

Casos de Uso del sistema

A continuación se notará todos los casos de uso que involucran al sistema, con sus respectivas descripciones.

Caso de Uso del Sistema: Interpretar receta médica

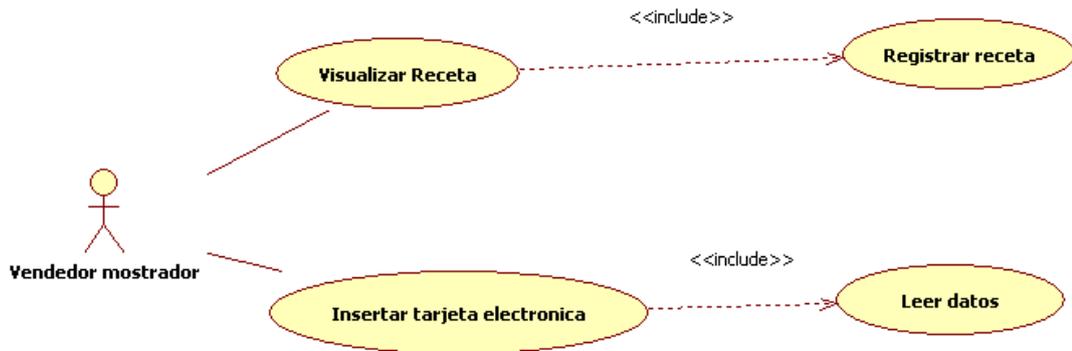


Figura 18: Diagrama de Caso de Uso: Interpretar receta médica

Tabla 9: Descripción del Caso de Uso: Interpretar receta médica

| Acciones | Descripción |
|-------------------------------|---|
| Visualizar receta médica | El vendedor de mostrador insertará la tarjeta del afiliado en el lector de tarjetas electrónicas para leer los datos del cliente que se presentan en pantalla. |
| Visualizar receta electrónica | El vendedor de mostrador será capaz de despachar los medicamentos presentados por el módulo recuperando los medicamentos de la receta médica grabados en la tarjeta del afiliado. |
| Leer datos | Mediante un API programado se obtendrá los datos guardados en la tarjeta del afiliado. |
| Registrar receta médica | Una vez visualizado la receta se generará un archivo xml correspondiendo a la receta el cual puede ser integrado con el sistema de facturación. |

Caso de Uso del Sistema: Generar receta médica

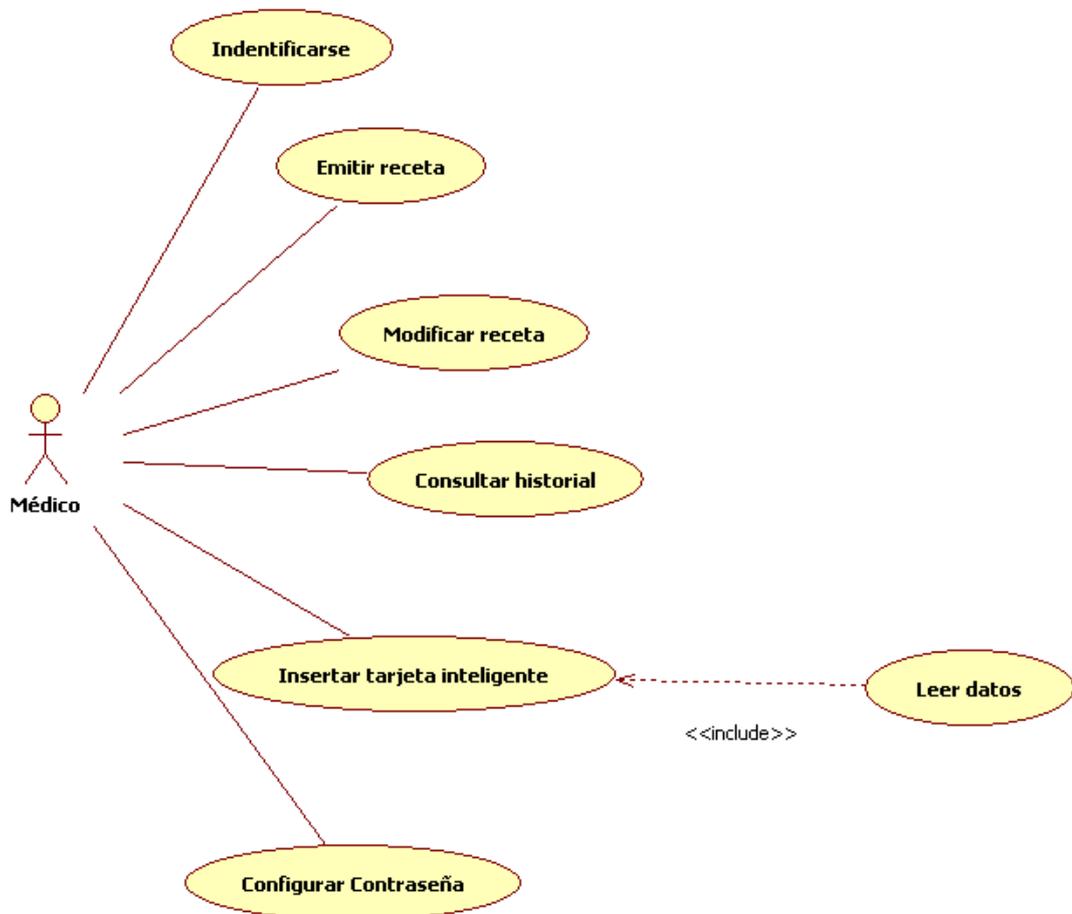


Figura 19: Diagrama del Caso de Uso: Generar receta médica

Tabla 10: Descripción del Caso de Uso: Generar receta médica

| Acciones | Descripción |
|----------------------|---|
| Identificarse | Ingresa al sistema ingresando su usuario y contraseña para poder emitir recetas médicas. |
| Emitir receta médica | El médico ingresa al sistema los medicamentos de la receta conjunto con la patología de cada medicamento para luego grabar en la tarjeta inteligente y en la base de datos. |
| Modificar receta | Si una vez grabada la receta se puede recuperarla para hacer sus respectivas modificaciones. |
| Consultar historial | Con este módulo el doctor podrá recuperar las recetas médicas ya no desde la tarjeta inteligente. |

| | |
|------------------------------|--|
| | sino de la base de datos con el fin de hacer consultas de las recetas emitidas según crea conveniente el médico. |
| Insertar tarjeta inteligente | Una vez insertada la tarjeta inteligente se obtendrán los datos del cliente. |
| Leer datos | Utilizando un API de lectura se obtendrá los datos del cliente para saber a que cliente se va a emitir la receta médica. |

Caso de Uso del Sistema: Administrar usuarios

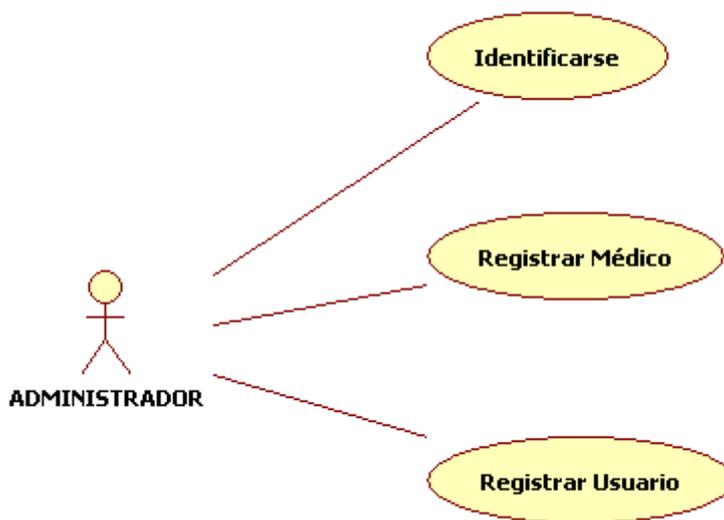


Figura 20: Diagrama del Caso de Uso: Administrar usuarios

Tabla 11: Descripción del Caso de Uso: Administración de Usuario

| Acciones | Descripción |
|-------------------|---|
| Identificarse | Ingresa al sistema ingresando su usuario y contraseña para poder emitir recetas médicas. |
| Registrar médico | El administrador deberá registrar el médico en el sistema. |
| Registrar Usuario | El administrador deberá registrar el nombre de usuario con el que el médico ingresará al sistema. |

4.2.3 Flujo Análisis

En esta fase del proyecto se realizará la Especificación de Casos de Uso, en el cual se hará una breve descripción de los mismos.

4.2.4 Flujo Diseño

En esta fase del proyecto se realizarán los diagramas de secuencia el cual nos indica como va el flujo de información interno en el sistema así la relación entre objetos y el intercambio de datos entre ellos.

Diagrama de secuencia: Interpretar receta

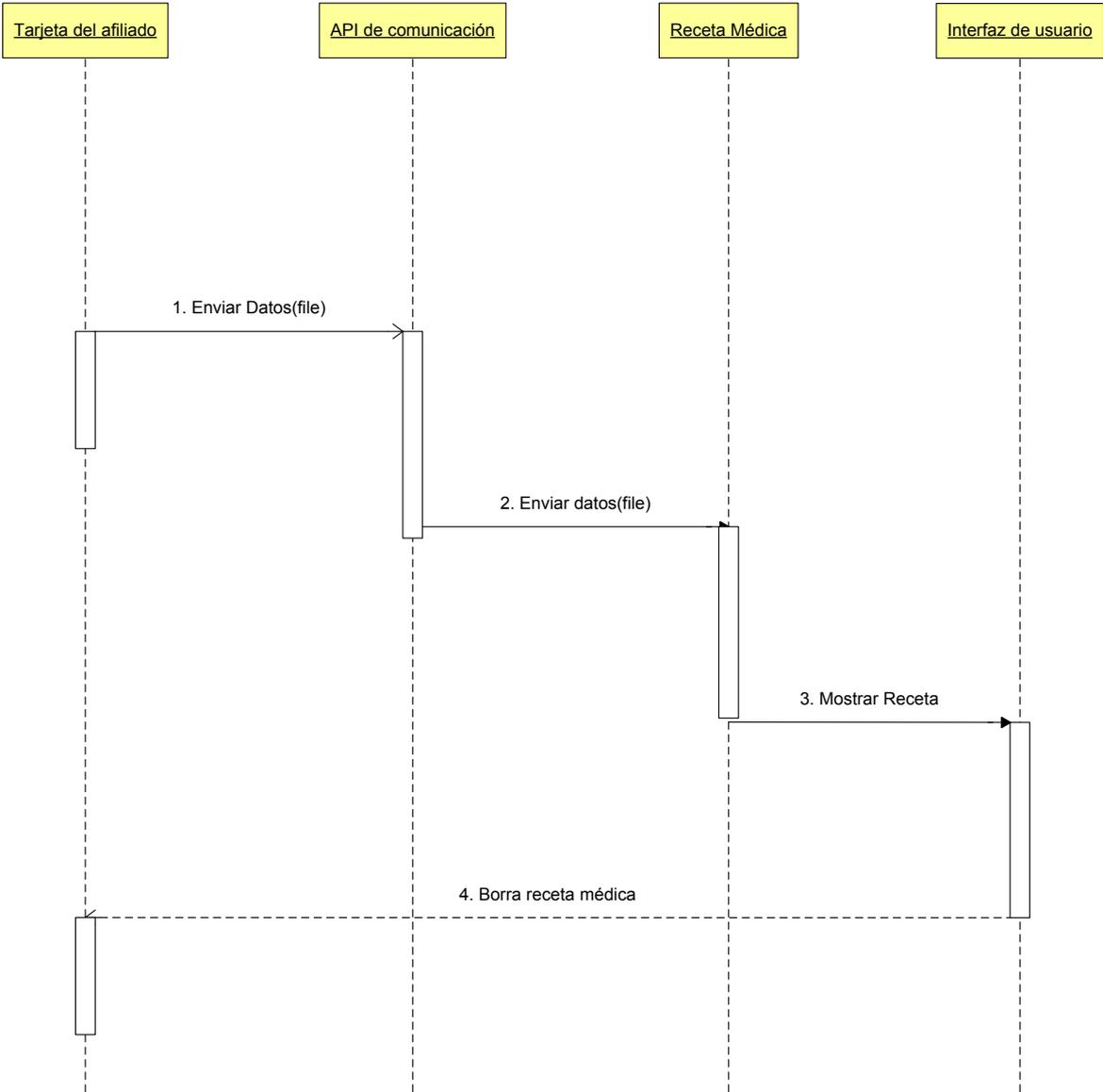


Figura 21: Diagrama de Secuencia: Interpretar receta médica

Diagrama de secuencia: Emitir receta médica

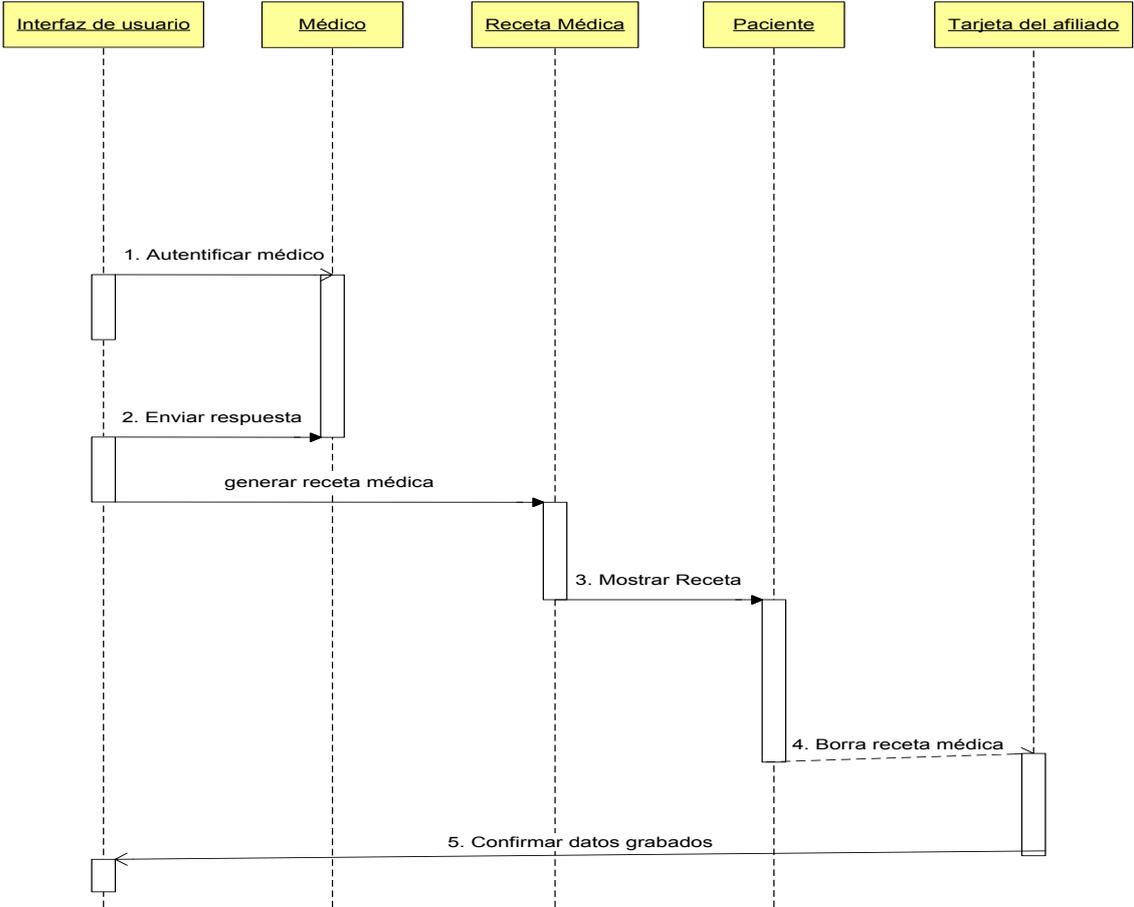


Figura 22: Diagrama de Secuencia: Emitir receta médica

Diagrama de secuencia: Administrar usuarios

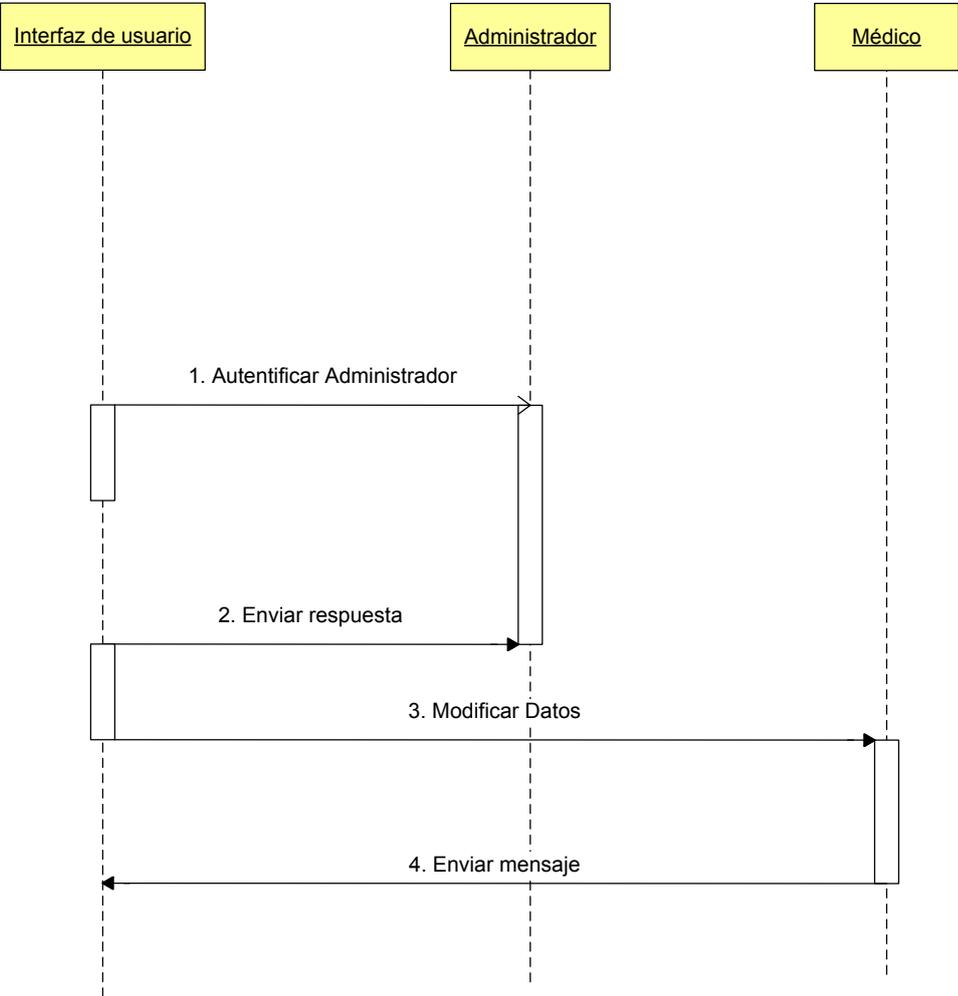


Figura 23: Diagrama de Secuencia: Administrar usuarios

4.2.5 Flujo Despliegue (Iteración 2)

En esta etapa se realizará la arquitectura del sistema acompañado del Diagrama de Despliegue, con la finalidad de que este módulo se acople no solo al sistema ya existente en la cadena Ecu farmacias, sino también a otros sistemas farmacéuticos.

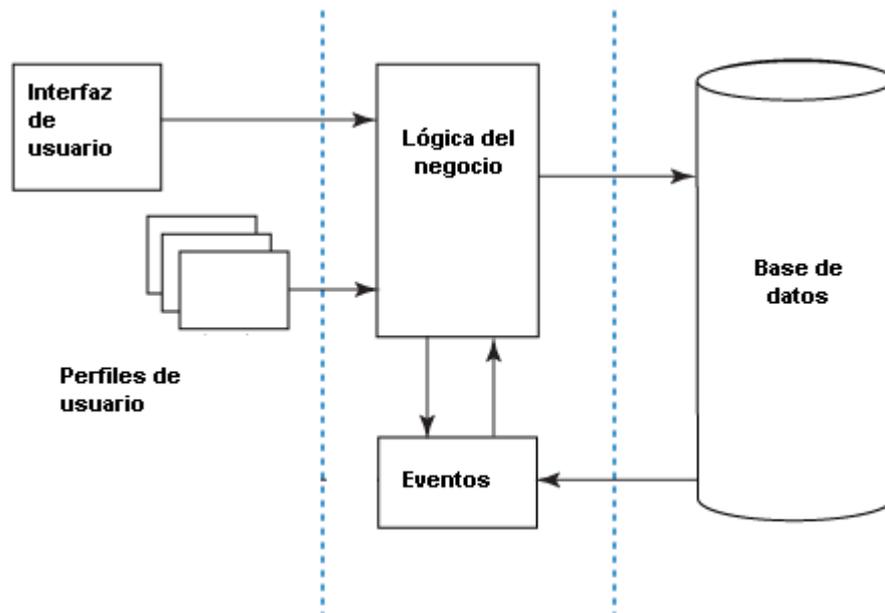


Figura 24: Arquitectura de tres capas

La arquitectura en tres capas separa las interfaces del usuario de la lógica del negocio o de la funcionalidad del sistema y del repositorio de datos con el fin de tener modularidad para una fácil comprensión y modificación del sistema cuando sea necesario.

Al tener separado lógicamente los niveles del sistema la instalación de este se hace mas fácil y se aprovecha mejor los recursos ya que se delegan funciones específicas a los equipos.

1.- Interfaz de usuario. Muestra la información del sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio. También es conocida como interfaz gráfica y debe tener la característica de ser "amigable" (entendible y fácil de usar) para el usuario.

2.- Capa de negocio: Reciben las peticiones del usuario y se envían las respuestas tras un proceso. Aquí se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos.

3.- Capa de datos: es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

4.3. Fase de Construcción

Los documentos a obtenerse en este proceso de desarrollo son: Documento de Especificación de Requerimientos de Software, Diagrama de Clases, Diagrama de Entidad Relación, Diagrama de Componentes.

4.3.1 Flujo Requisitos

El entregable en este flujo de trabajo es el Documento de Especificación de Requerimientos de Software.

4.3.2 Flujo Análisis

El entregable en este flujo de trabajo es el Diagrama de Clases, Un diagrama de clases describe la estructura del sistema mostrando sus clases, atributos y las relaciones entre ellos este diagrama nos muestra la idea para el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre uno y otro.

Diagrama de clases

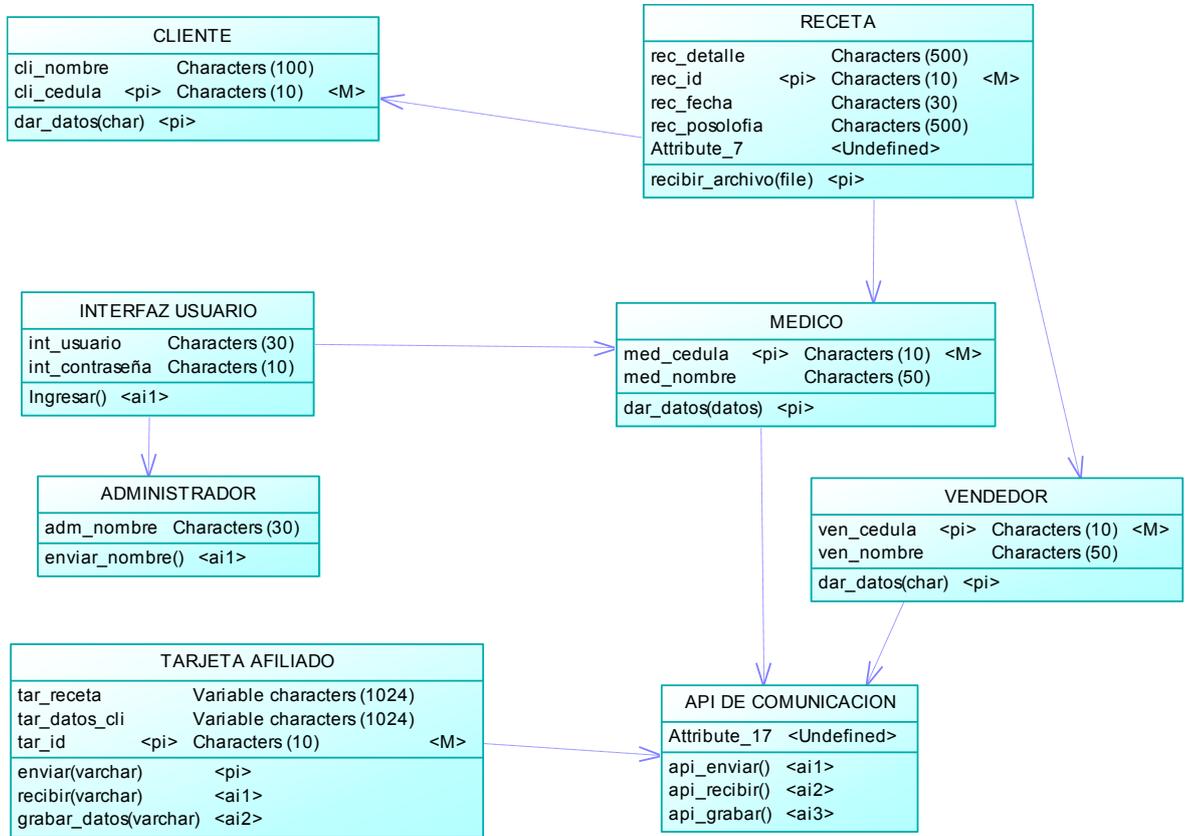


Figura 25: Diagrama de Clases

4.3.3 Flujo Diseño

El entregable en este flujo de trabajo es el Diagrama de Datos Entidad-Relación, nos muestra la estructura de la base de datos para el módulo de emisión de recetas médicas en tarjetas inteligentes.

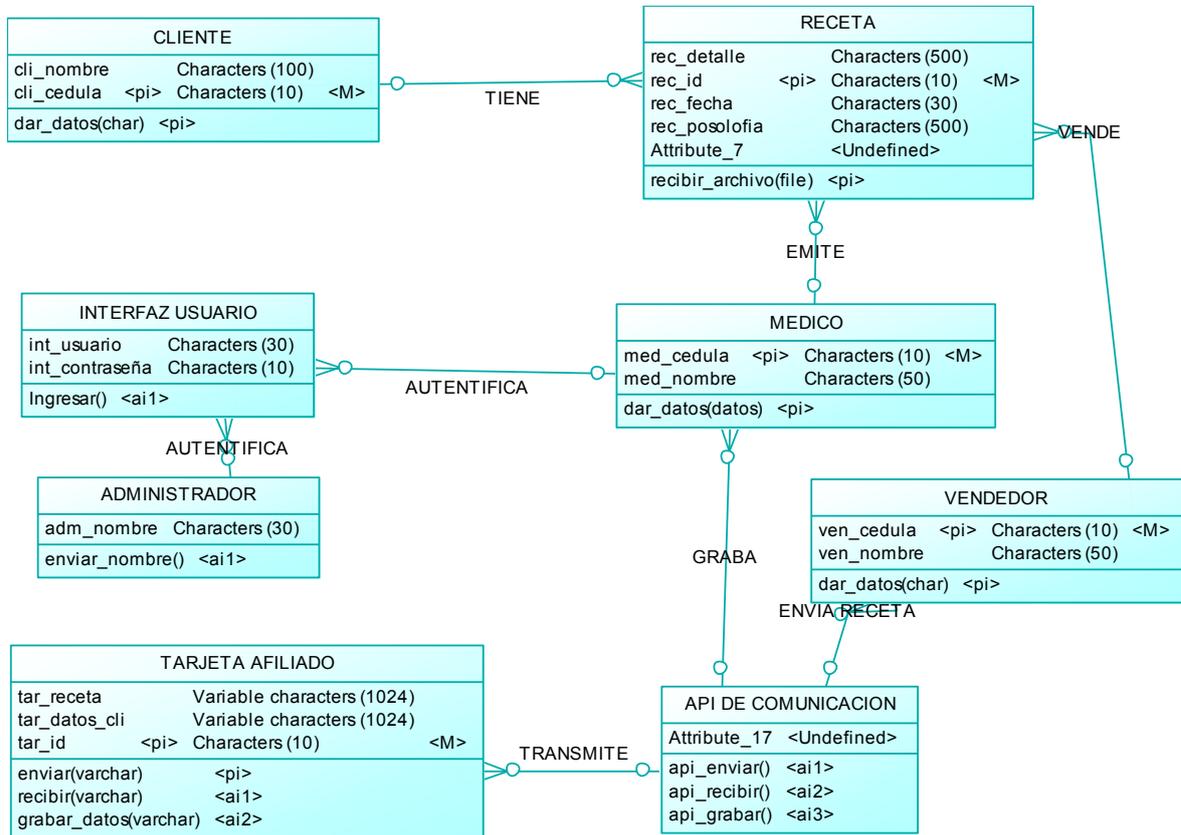


Figura 26: Diagrama de Entidad Relación (Modelo Conceptual)

4.3.4 Flujo Pruebas

En esta fase se realizará las Pruebas de Integración, estas son realizadas en el transcurso de la construcción del producto de software para garantizar que no habrá errores de último momento.

Tabla 12: Pruebas de Integración

| Petición | Respuesta |
|------------------------|--|
| Emitir recetas médicas | Cundo el médico necesita emitir por segunda vez una receta médica al mismo paciente se tendrá que borrar la anterior receta médica grabada ya que la tarjeta inteligente tiene limitaciones en su capacidad de almacenamiento. |

En las Pruebas de Especificaciones, en las que se verifica si las especificaciones establecidas en la etapa de requerimientos son cumplidas por la herramienta, para lo cual se detalla en la siguiente tabla.

Tabla 13: Tabla de Pruebas de Especificaciones

| Requerimiento | Respuesta |
|---|--|
| Aceptación de usuario | <p>Este sistema es similar a cualquier aplicación Windows y por tanto los usuarios estarán familiarizados con su entorno.</p> <p>Los usuarios entrarán al sistema identificándose sobre un ordenador con un sistema operativo Windows XP Profesional y tras este paso entrarán a la parte de aplicación diseñada para cada uno según su papel.</p> |
| Rápida atención al cliente | Al tener la receta digitalmente se ahorra tiempo en la interpretación. |
| Comodidad para el cliente | Al tener la receta médica en la tarjeta de afiliado se evita pérdidas u olvidos de la receta. |
| Confianza | Al tener la receta digital no hay errores humanos en la interpretación y el cliente se sentirá con la seguridad de llevar los medicamentos que necesita. |
| Evitar devoluciones | Al no tener equivocación en los despachos de las recetas médicas se evita la anulación de facturas y una nueva facturación. |
| Oportunidad de interactuar con otros sistemas | Gracias a que el sistema de recetas se maneja con archivos XML este tipo de archivos puede interactuar con otros sistemas sin importar su lenguaje de programación. |

4.3.5 Flujo Requisitos (Iteración II)

En esta fase del proyecto se presenta Interfaces de Usuario para una mejor apreciación del sistema propuesto así como su funcionalidad y el perfil que tendrá cada usuario del sistema.

Interfaz Acceso

Mediante esta interfaz los usuarios ingresarán al aplicativo, dependiendo de su perfil tendrán acceso a los formularios establecidos.



Figura 27: Interfaz de Acceso

Cambiar contraseña

En esta pantalla cuando el médico crea necesario podrá acceder a cambiar su contraseña siempre registrando su contraseña anterior.

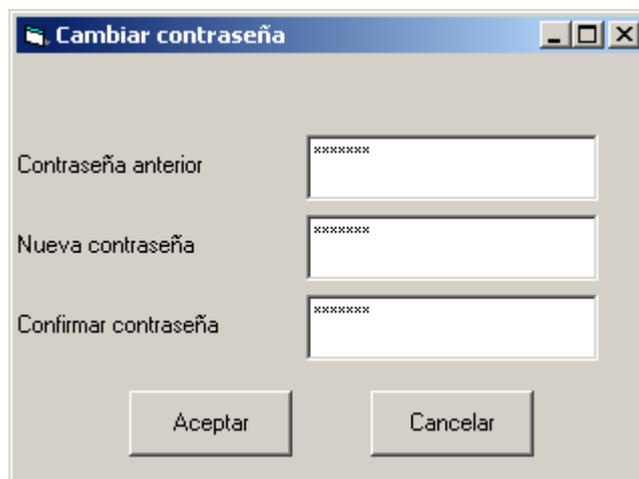


Figura 28: Interfaz Cambio de contraseña

Perfil del Médico

Pantalla Inicial

Este es el menú inicial que aparece después de ingresar el usuario y la contraseña del medico, en esta pantalla el medico elige la tarea que necesite hacer.



Figura 29: Interfaz Menú de Inicio

Emisión de recetas

Esta es la pantalla donde el médico llena la receta médica con los datos del paciente, los medicamentos y las indicaciones. El médico al ir agregando los medicamentos con su indicación y cantidad estos ítems se van agregando a los 2 listbox el primero corresponde a los medicamentos y el segundo corresponde a las indicaciones o Posología. El botón grabar es el encargado de guardar la receta médica en la tarjeta electrónica.

EMISION DE RECETAS MEDICAS

CEDULA: 1707707244 Quito sábado, 06 de marzo de 2010

NOMBRE: JUAN GOMEZ Nombre del doctor

Receta N°:

ITEM: METICORTEN 220 MG 10 Cant.: 10

POSOLOGIA: CADA 12 HORAS

| | | |
|-------------------|---|--|
| APRONAX 550 MG | | |
| VOLTAREN TABLETAS | 5 | |
| PHARMATON JARABE | 1 | |

| | |
|------------------------|--|
| CADA 2 HORAS | |
| 1 VEZ AL DÍA | |
| DESPUES DE CADA COMIDA | |

Figura 30: Interfaz Emisión de recetas médicas

Consulta de recetas

El médico cuando crea necesario podrá acceder a alguna receta emitida.

The screenshot shows a software window titled "MENU - [Consulta de recetas]". The window has a menu bar with "Emisión de Recetas" and "Consulta de Recetas". The main content area is titled "CONSULTA DE RECETAS MEDICAS" in a blue box. Below this, there are three input fields: "CEDULA" with the value "1707707244", "NOMBRE" with "JUAN GOMEZ", and "FECHA" with "17/11/2009". To the right of these fields is a button labeled "CONSULTAR". Below the input fields is a list box containing the text "APRONAX 550 MG 10 2 VECES AL DIA".

| Label | Value |
|--------|------------|
| CEDULA | 1707707244 |
| NOMBRE | JUAN GOMEZ |
| FECHA | 17/11/2009 |

CONSULTAR

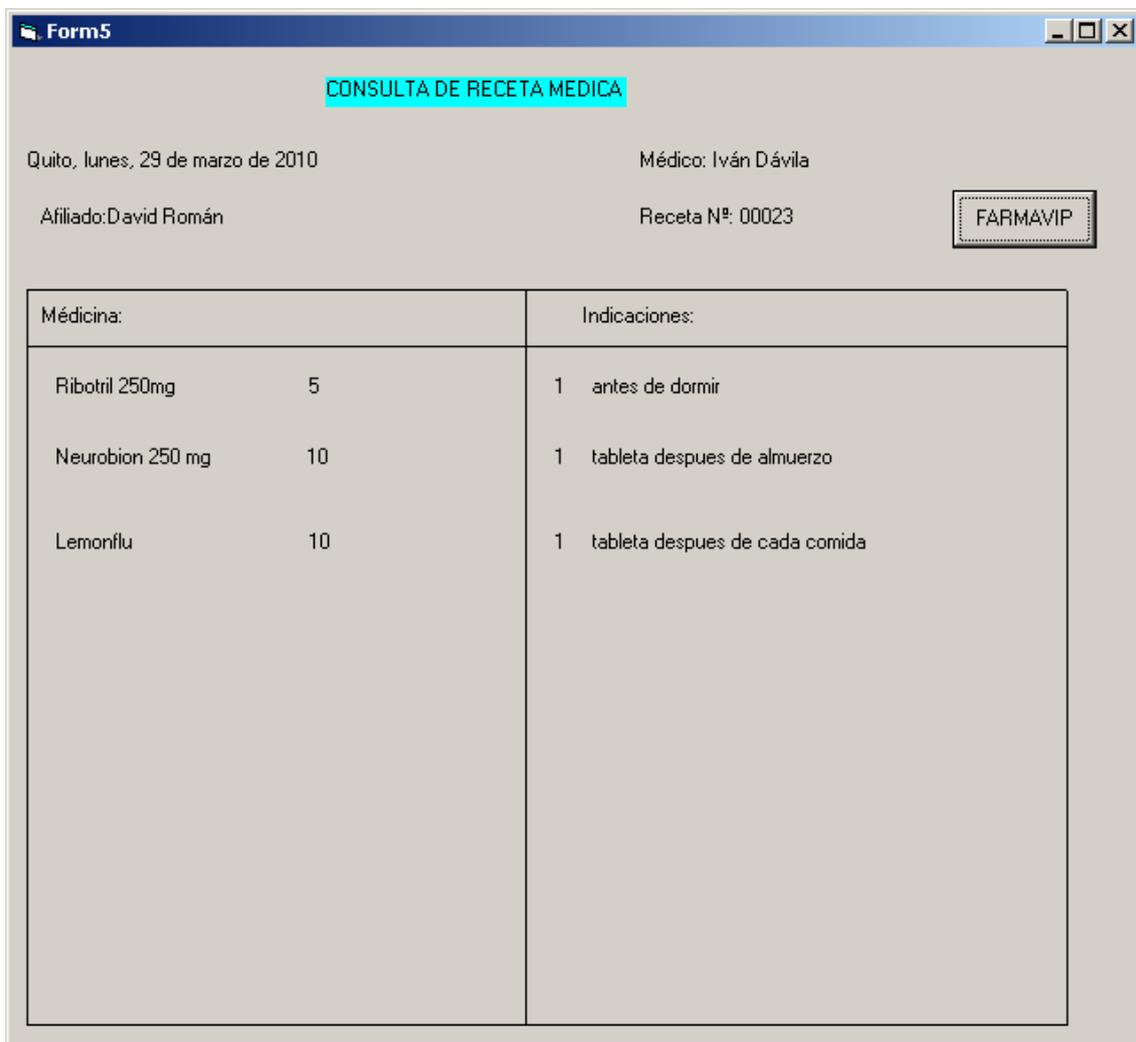
| | | |
|----------------|----|----------------|
| APRONAX 550 MG | 10 | 2 VECES AL DIA |
|----------------|----|----------------|

Figura 31: Consulta de Recetas médicas

Perfil Vendedor mostrador

Consulta de Recetas médicas

En esta pantalla es donde se carga la receta médica para que pueda ser observada por el vendedor mostrador, el vendedor al ingresar la tarjeta del afiliado tiene que aplastar el botón “FARMAVIP” entonces los datos de la receta médica guardados en la tarjeta inteligente serán cargados en la pantalla.



Form5

CONSULTA DE RECETA MEDICA

Quito, lunes, 29 de marzo de 2010 Médico: Iván Dávila

Afiliado: David Román Receta N°: 00023

| Medicina: | | Indicaciones: |
|------------------|----|----------------------------------|
| Ribotril 250mg | 5 | 1 antes de dormir |
| Neurobion 250 mg | 10 | 1 tableta despues de almuerzo |
| Lemonflu | 10 | 1 tableta despues de cada comida |

Figura 32: Consulta de recetas médicas

Perfil Administrador

Pantalla registrar nuevo usuario

El administrador del sistema tiene la capacidad de crear los usuarios para que el doctor pueda ingresar al sistema con un nombre de usuario.



The image shows a screenshot of a software window titled "Registrar Usuario". The window has a title bar with standard minimize, maximize, and close buttons. The main content area has a light gray background. At the top center, the text "REGISTRAR USUARIO" is displayed in a cyan-colored box. Below this, there are three input fields: "Usuario:" with the text "Medico_1", "Contraseña:" with masked characters "*****", and "Confirmar contraseña:" with masked characters "*****". At the bottom of the window, there are two buttons: "ACEPTAR" on the left and "CANCELAR" on the right.

Figura 33: Interfaz nuevo usuario

4.3.6 Flujo Pruebas (Iteración 2)

En esta fase de trabajo se realiza el artefacto Pruebas de Caja Blanca, las mismas que se ejecutaron sobre la lógica de programación del sistema. Estas pruebas garantizan la estabilidad de datos cuando se ponen a prueba los valores ingresados al módulo de emisión de recetas médicas.

Tabla 14: Tabla de Pruebas de Caja Blanca

| No. Pruebas | NOMBRE DE LA PRUEBA | ACCION | RESULTADO |
|------------------------|------------------------------------|---|---|
| 4 | Validación de Usuario y contraseña | Se ejecuta la aplicación y se ingresa el usuario y la contraseña para acceder al sistema. | Al momento de ingresar erróneamente los datos, si el usuario es incorrecto o la contraseña se muestra un mensaje pidiendo la corrección de los datos. |
| 4 | Emisión de receta médica | En la internas de emisión de recetas médicas se llena la receta con los diferentes datos. | Si no esta ingresada la tarjeta del afiliado en el lector aparece un mensaje de error pidiendo el el ingreso de la tarjeta. |
| 4 | Consultar receta médica | Se ingresan los datos del afiliado para buscar la receta médica. | Si no se encuentra la receta médica se muestra un mensaje de error diciendo que no existe receta médica. |

4.3.7 Flujo Despliegue (Iteración 2)

El entregable en este flujo se muestra el Diagrama de Despliegue, el cual indica cada uno de los nodos que son parte del producto de software y el diagrama de componentes que es la parte física y reemplazable de un sistema que conforma un conjunto de interfaces y proporciona la realización de esas interfaces.

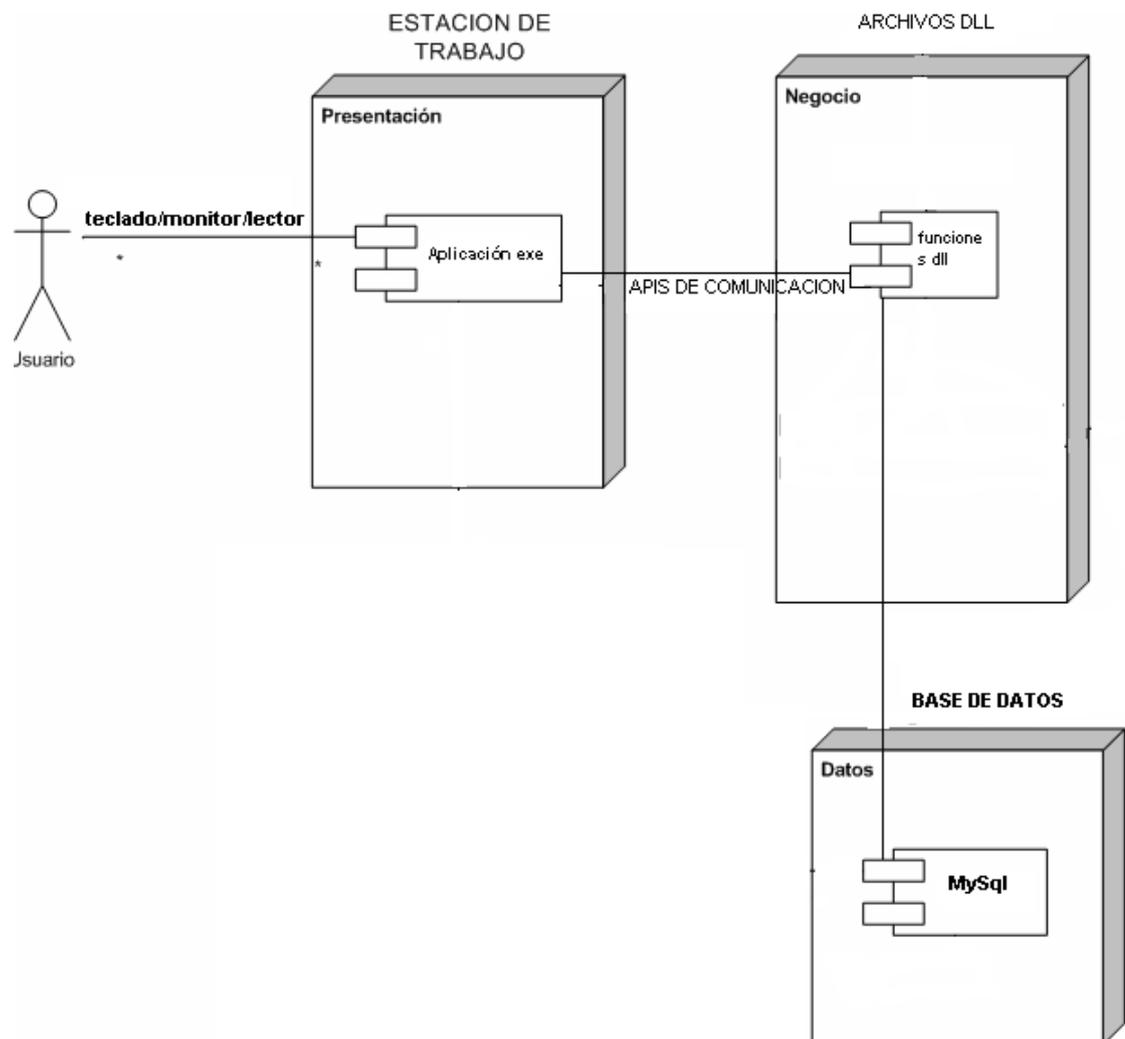


Figura 34: Diagrama de despliegue del producto de software

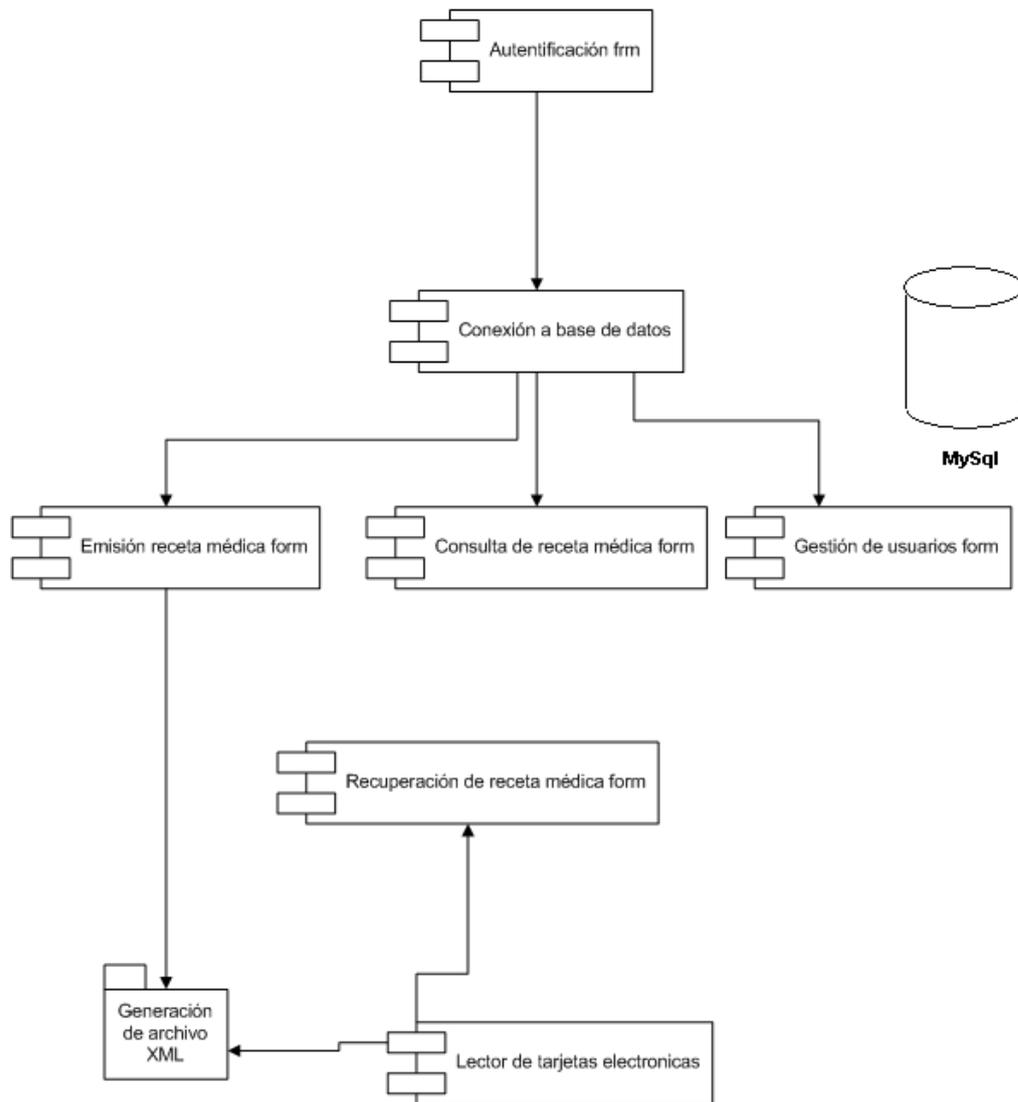


Figura 35: Diagrama de componentes

4.4 Fase de Transición

El objetivo en esta fase es entregar el producto, en esta fase será evaluado el proyecto por el usuario final.

Esta fase se asegura al cliente que el software listo el usuario final; posterior a las pruebas se afirma que el sistema funcione correctamente.

4.4.1 Flujo Pruebas

En este flujo se realiza el artefacto Pruebas de Caja Negra, estas pruebas la hacen los usuarios con el fin de probar la correcta funcionalidad del sistema.

Tabla 15: Tabla de Pruebas de Caja Negra

| No. Pruebas | NOMBRE DE LA PRUEBA | ACCION | RESULTADO |
|--------------------|----------------------------|---|---|
| 3 | Emitir receta médica | Se elije en el menú la pantalla de emisión de receta, se ingresa los datos. | En el campo cantidad se puede ingresar letras debido a que para grabar el archivo xml se toma todos los datos como texto. |
| 3 | Consultar receta médica | Se ingresan los datos del afiliado para buscar la receta médica. | Si no se encuentra la receta médica se muestra un mensaje de error diciendo que no existe receta médica. |
| 3 | Interpretar receta médica | Ingresar la tarjeta del afiliado al lector y abrir la ventana de consulta. | Si no se ingresa la tarjeta del afiliado sale un mensaje pidiendo insertar la tarjeta. |

De las pruebas realizadas se ha determinado que el módulo funcionará correctamente y es amigable y fácil de usar para el usuario.

Tabla 16: Versiones del Sistema

| Versiones de la Aplicación “Emisión de recetas médicas en tarjetas inteligentes” | |
|---|---|
| Versión 1 | En esta versión se desarrolló la primera versión del módulo con el fin de mostrar al usuario la funcionalidad y recoger sugerencias que se podrían implementar para mejorar este módulo, las pruebas se realizaron con el personal de farmacia. |
| Versión Final | Una vez realizado los cambios necesarios al módulo en la versión 1, con los usuarios se las pruebas para mostrar el correcto funcionamiento del sistema. |

5. CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

Después del desarrollo del producto de software para la emisión de recetas médicas en tarjetas inteligentes, se concluye que:

- Para el desarrollo de este proyecto se utilizó Ingeniería de Software y el Proceso de desarrollo RUP por la facilidad de hacer comprender el funcionamiento del negocio para mejorar los procesos, obtener beneficio y aprovechar oportunidades.
- Con la tecnología que nos brinda las tarjetas inteligentes permite dar un mejor servicio al cliente optimizando recursos, agilizar y disminuir el trabajo al vendedor de la farmacia.
- Con el módulo de consulta de recetas médicas electrónicamente se disminuye errores humanos que pueden cometer los vendedores de mostrador al no interpretar bien lo escrito por los médicos.
- El afiliado a la cadena farmacéutica al tener la receta guardada en la tarjeta de afiliación le brinda mas comodidad y evita la pérdida de las recetas físicas manuales que se da frecuentemente.
- Este módulo permite al médico ver un historial de recetas emitidas según la crea conveniente.
- Al generarse un archivo xml permite la interactividad con cualquier sistema incluido, el farmacéutico usado en la cadena de farmacias de ese modo se puede registrar en el sistema agilizando así las revisiones de CONSEP y la emisión de bonificaciones al médico.

- Al aprovechar la infraestructura de las tarjetas inteligentes permite optimizar recursos de la cadena y atender a la mayor cantidad de clientes, mostrar una imagen de una cadena de prestigio y un servicio extra frente a otras cadenas farmacéuticas.
- Permite al vendedor de mostrador mayor agilidad en la atención al cliente.
- Al ser un módulo aparte del sistema le permite total funcionalidad si el sistema farmacéutico deja de funcionar.

5.2 Recomendaciones

Después de la realización del proyecto, se recomienda:

- El trabajo en conjunto con el sistema farmacéutico para el intercambio de datos y la facturación automática con solo insertar la tarjeta del afiliado.
- Se podría realizar un portal de administración de recetas médicas en un portal Web ya que los archivos xml son completamente compatibles al trabajar con cualquier browser.
- Con este proyecto se pretende dar paso no solo a limitarse a pocos usos de la tarjeta inteligente sino a más opciones para su uso y aprovechamiento de la infraestructura que posee la cadena farmacéutica.
- Realizar periódicamente el mantenimiento de la y la base de datos para el correcto funcionamiento del módulo.
- Es necesario que exista un usuario que se encargue de la administración del módulo para corregir cualquier problema que se presente ya sea por mantenimiento del aplicativo o por la necesidad de aumentar su funcionalidad.

- Promover este servicio a sus clientes para que tenga una mejor acogida.

BIBLIOGRAFIA

- <http://c3po.es/>
- <http://support.microsoft.com/kb/106553/es>
- <http://www.mediavida.com/foro/9/multiples-errores-compilar-c-362414>
- <http://www.mediavida.com/foro/9/multiples-errores-compilar-c-362414#14>
- <http://www.functionx.com/visualc/libraries/staticdll.htm>.
- <http://www.scribd.com/doc/395783/RUP-etapa-diseno>.
- <http://www.recursosvisualbasic.com.ar/htm/tutoriales/tutorial-basico7.htm>.
- <http://tony-valderrama.blogspot.com/2007/09/creando-mens-con-visual-basic-60.html>
- <http://www.recursosvisualbasic.com.ar/htm/trucos-codigofuente-visual-basic/84.htm>
- http://sisab.lce.org/~hybrux/html/analisis_costo_beneficio.html
- http://es.wikipedia.org/wiki/Proceso_Unificado_de_Rational
- http://users.dsic.upv.es/asignaturas/facultad/lsi/ejemplorup/Gestion_Proyecto.html
- <http://www.recursosvisualbasic.com.ar/htm/vb-net/3-ejemplos-con-archivos-en-vb-net.htm>

- <http://www.monografias.com/trabajos2/guiavb/guiavb.shtml>
- http://www.c3po.cat/kit_ltc31.html
- http://manuelgross.bligoo.com/content/view/114756/Use_el_analisis_FODA_para_generar_estrategias.html
- <http://bioquimicosfarmaceuticosdelazuay.com.ec/descarga/archivo2.pdf>
- http://jcs.mobile-utopia.com/jcs/5631_ToolkitException.java

ANEXO 1

Cuestionario para dependientes

- 1.- ¿Encuentra usted dificultad en vender con recetas médica?
- 2.- ¿Cuál es el principal problema de vender con receta médicas?
- 3.- ¿Cuánto tiempo trabaja como vendedor de mostrador?
- 4.- ¿Alguna vez se a equivocado al interpretar un producto de una receta médica?
- 5.- ¿Qué hace usted cuando el cliente se olvida la receta médica?
- 6.- ¿Qué hace usted cuando no es legible la receta médica?
- 7.- ¿Cómo es el proceso de registro de una receta medica en el sistema?
- 8.- ¿Cuándo hay mucha gente en la farmacia y el cliente esta apurado, registra usted la receta en el sistema?

Este cuestionario se lo realizó vía correo electrónico a diferentes 10 personas que trabajan en la cadena de farmacias con los siguientes resultados:

Para la pregunta N° 1:

De las 10 personas encuestadas 6 presentan dificultad en interpretar la letra de los médicos ya que no es muy legible.

Para la pregunta N° 2:

De las 10 personas encuestadas 6 presentan se les dificulta leer la receta médica lo cual para ellos es el principal inconveniente, otro inconveniente que se observó es que a veces la farmacia no esta muy surtida y no pueden vender la receta completa.

Para la pregunta N° 3:

De las 10 personas encuestadas el promedio de trabajo es de 9 meses lo cual nos permite observar que hay mucha rotación del personal por lo tanto nos indica que hay muchos vendedores novatos.

Para la pregunta N° 4:

De las 10 personas encuestadas todas se han equivocado alguna vez en la interpretación de una receta médica en algún momento.

Para la pregunta N° 5:

De las 10 personas encuestadas todas dicen que es muy frecuente el olvido de la receta médica por parte del cliente y no se puede realizar la venta de los medicamentos.

Para la pregunta N° 6:

De las 10 personas encuestadas cuando no entienden la receta médica piden ayuda a algún compañero y en el último de los casos le piden al cliente que consulte el medicamento al médico que emitió la receta.

Para la pregunta N° 7:

Para registrar la receta médica los vendedores después de haber facturado la receta, eligen la ventana datos de cliente en la cual hay un botón donde dice registrar médico, este médico se lo busca por nombre, se elige el médico y se procede a guardar la factura.

Para la pregunta N° 8:

De las 10 personas encuestadas todas las personas cuando hay muchos clientes por atender no registran el médico.

Tabla 12: Pruebas de Integración

| Pregunta | Ventas con recetas Médicas | Porcent aje |
|---|---|------------------------|
| 1.- ¿Encuentra usted dificultad en vender con recetas médica? | De las 10 personas encuestadas 6 presentan dificultad en interpretar la letra de los médicos ya que no es muy legible | 60% |
| 2.- ¿Cuál es el principal problema de vender con receta médicas? | De las 10 personas encuestadas 6 presentan se les dificulta leer la receta médica lo cual para ellos es el principal inconveniente, otro inconveniente que se observó es que a veces la farmacia no esta muy surtida y no pueden vender la receta completa. | 60% |
| 3.- ¿Cuánto tiempo trabaja como vendedor de mostrador? | De las 10 personas encuestadas el promedio de trabajo es de 9 meses lo cual nos permite observar que hay mucha rotación del personal por lo tanto nos indica que hay muchos vendedores novatos. | 90% |
| 4.- ¿Alguna vez se a equivocado al interpretar un producto de una receta médica? | De las 10 personas encuestadas todas se han equivocado alguna vez en la interpretación de una receta médica en algún momento. | 100% |
| 5.- ¿Qué hace usted cuando el cliente se olvida la receta médica? | De las 10 personas encuestadas todas dicen que es muy frecuente el olvido de la receta médica por parte del cliente y no se puede realizar la venta de los medicamentos. | 100% |
| 6.- ¿Qué hace usted cuando no es legible la receta médica? | De las 10 personas encuestadas cuando no entienden la receta médica piden ayuda a algún compañero y en el último de los casos le piden al cliente que consulte el medicamento al médico que emitió la receta. | 100% |

| | | |
|---|--|-------------|
| <p>7.- ¿Cómo es el proceso de registro de una receta medica en el sistema?</p> | <p>Para registrar la receta médica los vendedores después de haber facturado la receta, elijen la ventana datos de cliente en la cual hay un botón donde dice registrar médico, este médico se lo busca por nombre, se elije el médico y se procede a guardar la factura</p> | |
| <p>8.- ¿Cuándo hay mucha gente en la farmacia y el cliente esta apurado, registra usted la receta en el sistema?</p> | <p>De las 10 personas encuestadas todas las personas cuando hay muchos clientes por atender no registran el médico.</p> | <p>100%</p> |

Conclusiones:

Con los datos obtenidos por la encuesta nos damos cuenta que el principal problema es la interpretación de la receta medica por parte de los empleados, en su mayoría los dependientes de las farmacias son vendedores novatos.

Otro problema identificado es el olvido o la pérdida de la receta médica por parte del cliente.

Para el desarrollo de este proyecto debemos tomar en cuenta estos factores para eliminar estos inconvenientes o reducirlos de la mejor manera.

ANEXO 2

INSTRUCTIVO PARA EL USO DE RECETA MÉDICA

EL MINISTRO DE SALUD PÚBLICA

CONSIDERANDO:

Que la Constitución Política de la República en el Art. 22 numeral 15 garantiza el derecho a un nivel de vida que asegure la salud, la asistencia médica y los servicios sociales necesarios;

Que por Ley sin número promulgada en el Registro Oficial No. 124 del 6 de agosto de 1997 se reformaron los Arts. 129 y 155 del Código de la Salud ampliando sus contenidos respecto a la forma como debe ser presente la receta médica; Que el Decreto No. 2007, publicado en el Registro Oficial No. 570 del 26 de noviembre de 1990, estableció el Sistema Nacional de Atención Farmacoterapéutica (SINAF) uno de cuyos componentes es el Subsistema de Dispensación y Expendio Institucional del Ministerio de Salud; Que mediante Acuerdo Ministerial No. 4639 de 3 de junio de 1994 se establece el Reglamento Interno para el funcionamiento del Subsistema de Dispensación y Expendio Institucional de Medicamentos del Ministerio de Salud Pública; Que el Ministerio de Salud Pública debe tener capacidad de gestión planificada, descentralizada, rectora del sector salud; Que es necesario promover el uso racional de medicamentos por parte de la comunidad y miembros del equipo de salud; Que es necesario arbitrar medidas que conduzcan a mejorar la gestión administrativa y calidad de los servicios farmacéuticos mediante la información actualizada; Que la receta médica constituye un vínculo de comunicación asistencial entre el prescriptor dispensador y paciente; En ejercicio a las atribuciones que le confiere el Art. 53 del Código de la Salud. Acuerda:

Art. 1.- El objeto del presente instructivo es regular y controlar la Prescripción y Dispensación de los medicamentos en las unidades del Ministerio de Salud Pública.

Art. 2.- Los médicos, odontólogos, y obstetras, están autorizados a prescribir, estrictamente los Medicamentos de su Especialidad.

Todos los profesionales autorizados para prescribir tienen la obligación de

sujetarse al Cuadro Nacional de medicamentos esenciales y niveles de complejidad.

Art. 3.- Para la comercialización de medicamentos es indispensable el Registro Sanitario Ecuatoriano.

Art. 4.- Las recetas médicas, además del nombre comercial obligatoriamente deberán llevar la denominación común Internacional DCI o el nombre genérico como lo señala el Art. 129 del Código de la Salud reformado por la Ley s/n publicada en el R.O. No. 124 del 6 de agosto de 1997.

Art. 5.- Deberá consignarse en la historia clínica respectiva toda prescripción de medicamentos con las mismas especificaciones que la receta a excepción de las emitidas por urgencia o emergencia.

El servicio de urgencias y emergencias llevará un registro donde se anotará el nombre del paciente, medicamentos y dosis prescritas.

Art. 6.- Las recetas en clave o que no contengan el nombre genérico del medicamento no serán despachadas, excepto en casos de emergencia de conformidad con el Art. 155 de la Ley Reformatoria indicada en el artículo 4 de éste Instructivo.

Art. 7.- Las recetas en las que se prescriban estupefacientes y psicotrópicos o elementos que las contengan, serán prescritas de conformidad con lo establecido en la ley y respectivos reglamentos.

Art. 8.- La receta para uso hospitalario y ambulatorio preferentemente será para un solo medicamento, en ella se escribirá en forma clara y legible el nombre genérico, concentración, forma farmacéutica y cantidad, además se señalará el número de días, y tipo de tratamiento.

Art. 9.- Los formatos de receta que se adjuntan se utilizarán de acuerdo al tipo de atención que se brinde además de diferenciarlo por colores, blanca para atención ambulatoria y rosada para hospitalización.

La receta de especialidades farmacéuticas de uso ambulatorio tiene pictogramas con el fin de ayudar a recordar la frecuencia de la administración.

Art. 10.- La persona encargada de entregar los recetarios, llevará un control en el que se registre el número de recetarios entregados, la numeración y los datos generales de cada profesional.

Art. 11.- Cuando se terminen los recetarios el profesional tiene la obligación de canjearlos. En caso de pérdida debe indicarlo inmediatamente.

Art. 12.- Las recetas se clasifican de acuerdo a las clases de prescripción:

- a) Recetas magistrales;
- b) Recetas de especialidades farmacéuticas para uso ambulatorio;
- c) Recetas de especialidades farmacéuticas de uso hospitalario; y,
- d) Recetas de especialidades farmacéuticas de uso restringido.

Art. 13.- Se conoce como especialidades farmacéuticas a los medicamentos de composición e información definidas, forma farmacéutica y dosificación determinadas, dispuesto y acondicionado para su dispensación al público, con denominación, embalaje, envase y etiquetado uniformes al que el estado a través del Ministerio de Salud autorice.

Art. 14.- Las recetas magistrales, son aquellas en las que el facultativo formula las sustancias medicinales con sus respectivas cantidades para que lo prepare el farmacéutico, destinado a un paciente individualizado.

Art. 15.- Las recetas de especialidades farmacéuticas para uso ambulatorio son aquellas en las que el paciente puede administrarse siguiendo las normas e información del facultativo. Para la dispensación se requiere de una receta médica de color blanco.

Art. 16.- Las recetas de especialidades farmacéuticas de uso hospitalario deben ser administradas a pacientes bajo tratamiento en un hospital. Para la dispensación es indispensable la receta de uso hospitalario. Para distinguirse será de color rosado.

Art. 17.- Las recetas de especialidades farmacéuticas de uso restringido, son aquellas que el Comité de Farmacoterapia o cada servicio impone ciertos requisitos para su uso. Para la dispensación se requerirá la presentación de una receta rosada con la firma de autorización del Jefe de Servicio.

Art. 18.- Para la dispensación de todo medicamento es indispensable la presentación de la receta médica, correctamente cumplimentada, a excepción de los medicamentos de venta libre.

Art. 19.- No se dispensarán las recetas médicas que contengan cantidades superiores de medicamentos para su tratamiento completo o que haya incompatibilidad entre ellos. El farmacéutico está en la obligación de pedir al prescriptor la rectificación, salvo el caso que la cantidad esté escrita en letras y subrayada.

Art. 20.- Para los pacientes crónicos, el médico determinará la cantidad a prescribir, pero en ningún caso la medicación superará la necesaria para dos meses.

Art. 21.- Para la dispensación de las recetas de especialidades farmacéuticas para uso ambulatorio, se considerará el tiempo de validez de la receta, pasado este período se requiere actualizarla.

a) La receta de urgencia es válida máximo 8 horas después de emitida;

b) La receta para paciente agudo es válida hasta 24 horas después de emitida;
y,

c) La receta para pacientes crónicos es válida hasta 5 días después de emitida.

Art. 22.- La dispensación de las recetas de especialidades farmacéuticas para uso hospitalario, cada día debe distribuirse la dosis respectiva de medicamentos prescritos, sin requerir cada vez de receta. El tiempo de validez de ésta receta, está determinada por el tipo de paciente, así: para pacientes estables la prescripción debe ser para tres días y para pacientes críticos la prescripción debe ser cada día.

Art. 23.- De conformidad al Art. 47 de la Ley de Substancias Estupefacientes y Psicotrópicas las recetas en que se ordene el despacho de sustancias estupefacientes o psicotrópicas o preparados que los contengan, caducan a las 72 horas de su expedición.

Art. 24.- Es obligatorio para la farmacia archivar todas las recetas originales, dispensadas con las respectivas fechas de prescripción y dispensación, a fin de proteger a sus autores de acciones legales posteriores que pudieran presentarse.

Art. 25.- La supervisión en cuanto a prescripción con el nombre genérico y que corresponda a los medicamentos incluidos en el cuadro Nacional de Medicamentos Básicos, se realizará con el control de los talonarios, recetas dispensadas e historias clínicas.

Art. 26.- El incumplimiento de las disposiciones emitidas en este instructivo será sancionado por las autoridades de la institución de acuerdo a lo establecido en la ley.

DISPOSICION TRANSITORIA

Se establece como plazo máximo hasta marzo de 1999, para que todas las unidades de salud faciliten a los prescriptores las recetas pre impresas señaladas en el artículo anterior.

ANEXO 3

Modo de comunicación con la Tarjeta Inteligente

Descripción de la tarjeta inteligente

Tarjeta de 256 bytes de 8 bits de memoria principal EEPROM y 32 bits de memoria de protección funcional PROM.

La memoria principal se borra y escribe byte a byte. Al ser borrado, los 8 bits del byte se colocan en la posición lógica 1. La escritura y el borrado tienen un tiempo de 2.5 ms cada uno.

Los primeros 32 bits pueden ser irreversiblemente protegidos contra cambios mediante la escritura de Aplicaciones tipo correspondiente bit en la memoria protegida.

Características Técnicas

- Tarjeta de control industrial
- Tarjeta para carga de parámetros 256 x 8-bit de organización de EEPROM.
- Tarjeta publicitaria
- Tarjeta para almacén de datos

Introducción

El lector - grabador LTC31 es un dispositivo diseñado para realizar la comunicación entre el sistema donde se ha programado la aplicación (llamado PC) y la tarjeta chip. El panel frontal contiene:

Ranura de entrada para la tarjeta.

Dos LEDs, rojo y verde.

En la parte lateral dispone de un cable con los siguientes conectores en su extremo:

Un conector de 9 pins para conexión al puerto RS232 del PC.

Un conector hembra miniDin de 6 contactos para conectar el teclado o ratón (se aconseja hacerlo en el ratón).

Un conector macho miniDin de 6 contactos para conectar al espacio destinado para la conexión del teclado o ratón del PC.

El LTC31 se controla mediante comandos suministrados por el PC. Las principales funciones del LTC31 son controlar que la tarjeta está bien insertada, se alimente correctamente y converger los comandos y respuestas entre el PC y la tarjeta chip.

El LTC31 cumple con las especificaciones Plug and Play (PnP) para dispositivos COM (comunicación serie) Versión 1.0 de Microsoft.

El LTC31, a nivel de hardware, es compatible con las especificaciones PC/SC de Microsoft. Utilizando el driver adecuado, que suministra C3PO, S.L. junto con el LTC31, el producto es 100% compatible con las especificaciones PC/SC de Microsoft Versión 1.0 (Ver Apéndice C del presente documento referido al driver PC/SC).

Indicadores

El LTC31 tiene dos LEDs la combinación de los cuales se usa para indicar los distintos estados del lector. La Tabla 2-01 nos muestra dichos estados.

Tabla 2-01 Estado de los led

| ROJO | VERDE | SIGNIFICADO |
|--------------|--------------|---------------------------------|
| OFF | OFF | Lector en reposo, no ha tarjeta |
| ON | OFF | Tarjeta insertada |
| OFF | ON | Tarjeta insertada y alimentada |
| PARPA DEO | OFF | El PC está detectando el LTC31 |

En el caso de que haya una tarjeta insertada y el led rojo no se encienda, es debido a que el LTC31 no ha sido reconocido por el driver.

Librerías

Se trata de una librería en formato DLL (Dinamic Link Library) para la comunicación con el lector LTC31.

La estructura de este apartado para la descripción de las distintas funciones contenidas en la librería es la siguiente:

| | |
|--------------------|--|
| Descripción | Aquí se describe en líneas generales la función a la que se hace referencia. Explicando todo aquello que será necesario para su implementación. |
| Sintaxis | Aquí se detalla la sintaxis que se aplica para el uso de la función a la que se hace referencia. |
| Respuesta | Aquí se indican las posibles respuestas que suministra la librería tras la ejecución de la función. |
| Notas | Se añaden notas explicativas sobre algún tema en concreto que pueda ser necesario para el buen funcionamiento de la librería en relación con la función a la que hacemos referencia. |
| Ejemplos | Se especifican uno o varios ejemplos para el buen entendimiento de la función, tanto a nivel de sintaxis como de las posibles respuestas que nos puede suministrar la función a la que se hace referencia. |

Para hacer uso de la librería deberemos trabajar con los siguientes ficheros:

Nombre de Fichero

LTC2x.H

LTC2x.LIB

Funciones de comunicación:

C3OPEN

Descripción Abre el puerto serie de comunicaciones e inicializa dicho puerto para poder comunicar con el LTC31.

Sintaxis #include "LTC2x.H"

HANDLE **C3OPEN**(LPSTR Port)

Port Nombre del puerto que se desea abrir:

COM1 COM2 COM3 COM4

Respuesta Si el valor devuelto es < 0 Error

Si el valor devuelto es ? 0 Identificador del puerto

Notas Esta función debe ser la primera en ejecutarse. El identificador del puerto devuelto debe utilizarse con todas las demás funciones.

Ejemplos #include "LTC2x.H"

HANDLE hCommDev;

void main()

{hCommDev = C3OPEN("COM2");

if(hCommDev < 0) ExitMessage("Error al abrir el puerto");

...

C3CLOSE(hCommDev);}

C3CLOSE

Descripción Cierra el puerto serie de comunicaciones previamente abierto.

Sintaxis #include "LTC2x.H"

BOOL **C3CLOSE**(HANDLE hCommDev)

hCommDev Identificador del puerto.

Respuesta Si el valor devuelto es = 0 (False) Error

Si el valor devuelto es ? 0 (True) Ejecución correcta

Notas Esta función debe ser la última en ejecutarse. El identificador del puerto devuelto por C3OPEN debe utilizarse con ésta y todas las demás funciones.

Ejemplos `#include "LTC2x.H"`

```
HANDLE hCommDev;
```

```
void main()
```

```
{hCommDev = C3OPEN( "COM2" );
```

```
if( hCommDev < 0 ) ExitMessage( "Error al abrir el puerto" );
```

```
...
```

```
C3CLOSE( hCommDev );}
```

C3RESET

Descripción Inicializa el lector desactivando los contactos de la tarjeta y preestableciéndose todos los parámetros internos a su valor por defecto. Devuelve la respuesta al reset del lector.

Sintaxis `#include "LTC2x.H"`

```
LONG C3RESET( HANDLE hCommDev, LPINT ATRLen, LPSTR ATR)
```

hCommDev Identificador del puerto.

ATRLen Longitud de la cadena devuelta en ATR.

(Longitud máxima 80 caracteres).

ATR Cadena de caracteres correspondiente a la respuesta al reset del lector

Respuesta Si el valor devuelto es < 0 Error de comunicaciones

Si el valor devuelto es = 0 Ejecución correcta

Si el valor devuelto es > 0 Error detectado en el lector

Notas Esta función inicializa el lector a sus valores por defecto, interrumpiendo cualquier proceso de comunicación con la tarjeta y desactivando los contactos de la tarjeta.

Ejemplos `#include "LTC2x.H"`

```
HANDLE hCommDev;
```

```
int nLen;
```

```
char ATR[80];
```

```

void main()
{long IRet;
hCommDev = C3OPEN( "COM2" );
if( hCommDev < 0 ) ExitMessage( "Error al abrir el puerto");
IRet = C3RESET( hCommDev, &nLen, ATR );
if( IRet <> 0 ) ExitMessage( "Error del lector" );
...
C3CLOSE( hCommDev );}

```

C3INSERT

Descripción Espera a que se inserte una tarjeta en el contactor de tarjeta especificado del lector.

Sintaxis #include "LTC2x.H"

LONG **C3INSERT**(HANDLE hCommDev, int nCCA, LONG TimeOut)

hCommDev Identificador del puerto.

nCCA Contactor de tarjeta. (0 = Primero)

TimeOut Tiempo máximo de espera en milisegundos

Respuesta Si el valor devuelto es < 0 Error de comunicación

Si el valor devuelto es = 0 Ejecución correcta

Notas En el caso del LTC31 al disponer de un solo contactor de tarjeta nCCA siempre será 0.

Ejemplos #include "LTC2x.H"

HANDLE hCommDev;

void main()

long IRet;

hCommDev = C3OPEN("COM2");

if(hCommDev < 0) ExitMessage("Error al abrir el puerto");

IRet = C3INSERT(hCommDev, 0, 30000);

if(IRet = 0) {

/ Se ha insertado una tarjeta */...*

C3REMOVE(hCommDev, 0, 30000);}

C3CLOSE(hCommDev);}

C3REMOVE

Descripción Espera a que se extraiga la tarjeta del contactor de tarjeta especificado del lector.

Sintaxis #include "LTC2x.H"

LONG **C3REMOVE**(HANDLE hCommDev, int nCCA, LONG TimeOut)

hCommDev Identificador del puerto.

nCCA Contactor de tarjeta. (0 = Primero)

TimeOut Tiempo máximo de espera en milisegundos

Respuesta Si el valor devuelto es < 0 Error de comunicación

Si el valor devuelto es = 0 Ejecución correcta

Notas En el caso del LTC31 al disponer de un solo contactor de tarjeta nCCA siempre será 0.

Ejemplos #include "LTC2x.H"

```
HANDLE hCommDev;
```

```
void main()
```

```
{long IRet;
```

```
hCommDev = C3OPEN( "COM2" );
```

```
if( hCommDev < 0 ) ExitMessage( "Error al abrir el puerto");
```

```
IRet = C3INSERT( hCommDev, 0, 30000 );
```

```
if( IRet = 0 ) /* Se ha insertado una tarjeta */...
```

```
C3REMOVE( hCommDev, 0, 30000 );}
```

```
C3CLOSE( hCommDev );}
```

C3POWERON

Descripción Alimenta e inicializa la tarjeta insertada en el contador correspondiente y devuelve la respuesta al reset de la tarjeta.

Sintaxis #include "LTC2x.H"

LONG **C3POWERON**(HANDLE hCommDev, int nCCA, int CardType,

LPINT ATRLen, LPSTR ATR, LPBYTE SW1, LPBYTE SW2)

hCommDev Identificador del puerto.

nCCA Contactor de tarjeta. (0 = Primero)
 CardType Tipo de tarjeta (0 = Microprocesada T=0)
 ATRLen Longitud de la cadena correspondiente a la respuesta al reset.
 ATR Cadena correspondiente a la respuesta al reset por parte de la tarjeta.
 SW1 Status Word 1.
 SW2 Status Word 2.

Respuesta Si el valor devuelto es < 0 Error de comunicación
 Si el valor devuelto es = 0 Ejecución correcta

Notas En el caso del LTC31 al disponer de un solo contactor de tarjeta nCCA siempre será 0.

Ejemplos *#include "LTC2x.H"*

```
HANDLE hCommDev;
int nATRLen;
char ATR[80];
char SW[2];
void main()
{long IRet;
hCommDev = C3OPEN( "COM2" );
if( hCommDev < 0 ) ExitMessage( "Error al abrir el puerto");
IRet = C3INSERT( hCommDev, 0, 30000 );
if( IRet = 0 ) /* Se ha insertado una tarjeta */
if( C3POWERON( hCommDev, 0, 0, &nATRLen, ATR,
&SW[0], &SW[1] ) == 0 ) {
/* Tarjeta inicializada */...
C3POWEROFF( hCommDev, 0 );}
C3REMOVE( hCommDev, 0, 30000 );}
C3CLOSE( hCommDev );}
```

C3POWEROFF

Descripción Desactiva la tarjeta insertada en el contactor correspondiente, terminando así el proceso de interacción con ella.

Sintaxis #include "LTC2x.H"

LONG **C3POWEROFF**(HANDLE hCommDev, int nCCA)

hCommDev Identificador del puerto.

nCCA Contactor de tarjeta. (0 = Primero)

Respuesta Si el valor devuelto es < 0 Error de comunicación

Si el valor devuelto es = 0 Ejecución correcta

Notas En el caso del LTC31 al disponer de un solo contactor de tarjeta nCCA siempre será 0.

Ejemplos #include "LTC2x.H"

```
HANDLE hCommDev;
```

```
int nATRLen;
```

```
char ATR[80];
```

```
char SW[2];
```

```
void main()
```

```
{long lRet;
```

```
hCommDev = C3OPEN( "COM2" );
```

```
if( hCommDev < 0 ) ExitMessage( "Error al abrir el puerto");
```

```
lRet = C3INSERT( hCommDev, 0, 30000 );
```

```
if( lRet = 0 ) /* Se ha insertado una tarjeta */
```

```
if( C3POWERON( hCommDev, 0, 0, &nATRLen, ATR,
```

```
&SW[0], &SW[1] ) == 0 ) /* Tarjeta inicializada */...
```

```
C3POWEROFF( hCommDev, 0 );}
```

```
C3REMOVE( hCommDev, 0, 30000 );}
```

```
C3CLOSE( hCommDev );}
```

C3ISOIN

Descripción Ejecución de un comando ISO 7816 de escritura en la tarjeta introducida y previamente inicializada en el contactor de tarjeta correspondiente.

Sintaxis #include "LTC2x.H"

LONG **C3ISOIN**(HANDLE hCommDev, int nCCA, BYTE Cla, BYTE

Ins, BYTE P1, BYTE P2, BYTE P3, LPSTR Data, LPBYTE SW1,

LPBYTE SW2, LONG TimeOut)

hCommDev Identificador del puerto.

nCCA Contactor de tarjeta. (0 = Primero)

Cla

Ins

P1

P2

P3

Campos del comando según ISO7816

Ver la documentación de la tarjeta pertinente para la correcta interpretación de estos comandos.

Data Cadena correspondiente a los datos a enviar a la tarjeta. Ver la documentación de la tarjeta pertinente.

SW1 Status Word 1.

SW2 Status Word 2.

TimeOut Tiempo máximo de espera en milisegundos

Respuesta Si el valor devuelto es < 0 Error de comunicación

Si el valor devuelto es = 0 Ejecución correcta

Notas En el caso del LTC31 al disponer de un solo contactor de tarjeta nCCA siempre será 0.

Para la correcta interpretación de esta función consultar el comando correspondiente en la documentación de la tarjeta utilizada.

SW1 y SW2 son devueltos por la tarjeta y deberán ser analizados en el caso de una respuesta = 0 para determinar la ejecución de dicho comando en la tarjeta.

Ejemplos *#include "LTC2x.H"*

HANDLE hCommDev;

int nATRLen;

char ATR[80];

char SW[2];

void main()

```

{long IRet;
hCommDev = C3OPEN( "COM2" );
if( hCommDev < 0 ) ExitMessage( "Error al abrir el puerto" );
IRet = C3INSERT( hCommDev, 0, 30000 );
if( IRet = 0 ) /* Se ha insertado una tarjeta */
if( C3POWERON( hCommDev, 0, 0, &nATRLen, ATR,
&SW[0], &SW[1] ) == 0 ) /* Tarjeta inicializada */
if( C3ISOIN( hCommDev, 0, 0x00, 0xE4, 0x01,
0x00, 0x00, NULL, &SW[0], &SW[1], 10000 ) == 0 && SW[0]
== 0x90 } /* Abierto el fichero 0x01 de C3M2K */...}
C3POWEROFF( hCommDev, 0 );}
C3REMOVE( hCommDev, 0, 30000 );}
C3CLOSE( hCommDev );}

```

C3ISOOUT

Descripción Ejecución de un comando ISO 7816 de lectura en la tarjeta introducida y previamente inicializada en el contactor de tarjeta correspondiente.

Sintaxis #include "LTC2x.H"

LONG **C3ISOOUT**(HANDLE hCommDev, int nCCA, BYTE Cla, BYTE Ins, BYTE P1, BYTE P2, BYTE P3, LPSTR Data, LPBYTE SW1, LPBYTE SW2, LONG TimeOut)

hCommDev Identificador del puerto.

nCCA Contactor de tarjeta. (0 = Primero)

Cla

Ins

P1

P2

P3

Campos del comando según ISO7816

Ver la documentación de la tarjeta pertinente para la correcta interpretación de estos

comandos.

Data Cadena correspondiente a los datos recibidos de la tarjeta. Ver la documentación de la tarjeta pertinente.

SW1 Status Word 1.

SW2 Status Word 2.

TimeOut Tiempo máximo de espera en milisegundos

Respuesta Si el valor devuelto es < 0 Error de comunicación

Si el valor devuelto es = 0 Ejecución correcta

Notas En el caso del LTC31 al disponer de un solo contactor de tarjeta nCCA siempre será 0.

Para la correcta interpretación de esta función consultar el comando correspondiente en la documentación de la tarjeta utilizada.

SW1 y SW2 son devueltos por la tarjeta y deberán ser analizados en el caso de una respuesta = 0 para determinar la ejecución de dicho comando en la tarjeta.

Ejemplos *#include "LTC2x.H"*

```
HANDLE hCommDev;
```

```
int nATRLen;
```

```
char ATR[80];
```

```
char SW[2];
```

```
void main()
```

```
{long lRet;
```

```
hCommDev = C3OPEN( "COM2" );
```

```
if( hCommDev < 0 ) ExitMessage( "Error al abrir el puerto" );
```

```
lRet = C3INSERT( hCommDev, 0, 30000 );
```

```
if( lRet = 0 ) /* Se ha insertado una tarjeta */
```

```
if( C3POWERON( hCommDev, 0, 0, &nATRLen, ATR, &SW[0], &SW[1] ) == 0 ) /* Tarjeta inicializada */
```

```
if( C3ISOIN( hCommDev, 0, 0x00, 0xE4, 0x01, 0x00, 0x00, NULL, &SW[0], &SW[1], 10000 ) == 0 && SW[0] == 0x90 ) /* Abierto el fichero 0x01 de C3M2K */
```

```
if( C3ISOOUT( hCommDev, 0, 0x00, 0xC6,
```

```

0x01, 0x00, 0x10, sATR, &SW[0], &SW[1], 10000 ) == 0 &&
SW[0] == 0x90 ) /* Leídos 16 bytes del primer registro del
fichero */...}
C3POWEROFF( hCommDev, 0 );}
C3REMOVE( hCommDev, 0, 30000 );}
C3CLOSE( hCommDev );}

```

C3SENDCOMMAND

Descripción Envía un comando al lector a bajo nivel. Esta función puede usarse para analizar el comportamiento del lector a los comandos.

Sintaxis #include "LTC2x.H"

```

LONG C3SENDCOMMAND(HANDLE hCommDev, BYTE RsCode,
BYTE RsCla, BYTE RsIns, BYTE RsP1, BYTE RsP2, BYTE RsP3,
LPSTR RsData, LPBYTE RsSW1, LPBYTE RsSW2, LPINT RsLen,
LONG TimeOut )

```

hCommDev Identificador del puerto.

RsCode

RsCla

RsIns

RsP1

RsP2

RsP3

RsData

RsSW1

RsSW2

Campos del comando

(máximo 256 caracteres)

RsLen Longitud de los datos pasados a RsData

TimeOut Tiempo máximo de espera en milisegundos

Respuesta Si el valor devuelto es < 0 Error de comunicación

Si el valor devuelto es ? 0 RsStat

Ejemplos #include "LTC2x.H"

```
HANDLE hCommDev;
int nATRLen;
char ATR[80];
char SW[2];
void main()
{long IRet;
hCommDev = C3OPEN( "COM2" );
if( hCommDev < 0 ) ExitMessage( "Error al abrir el puerto");
/* Insertar tarjeta */
IRet = C3SENDCOMMAND( hCommDev, 0x20, 0x00, 0x20,
0x00, 0x00, 0x00, NULL, &SW[0], &SW[1], NULL, 10000 );
C3CLOSE( hCommDev );}
```

C3ICCSTATUS

Descripción Devuelve el estado actual del contactor de tarjeta especificado.

Sintaxis #include "LTC2x.H"

LONG **C3ICCSTATUS**(HANDLE hCommDev, int nCCA)

hCommDev Identificador del puerto.

nCCA Contactor de tarjeta (0 = primero)

Respuesta Si el valor devuelto es < 0 Error de comunicación

Si el valor devuelto es = 0 Hay tarjeta y está alimentada

Si el valor devuelto es = 192 Hay tarjeta no alimentada

Si el valor devuelto es = 80 No hay tarjeta

Notas En el caso del LTC31 al disponer de un solo contactor de tarjeta nCCA siempre será 0.

Ejemplos #include "LTC2X.H"

```
HANDLE hCommDev;
void main()
{hCommDev = C3OPEN("COM2");
if( hCommDev >= 0 ) {
switch( C3ICCSTATUS( hCommDev, 0 ) {case 0:
```

```

printf( "Hay una tarjeta en el lector y está
alimentada\n" );
break;
case 192:
printf( "Hay una tarjeta en el lector y no está
alimentada\n" );
break;
case 80:
printf( "No hay tarjeta en el lector\n" );
break;}
C3CLOSE(hCommDev);}}

```

C3CARDINSERTED

Descripción Devuelve el estado del indicador de presencia de tarjeta del contactor de tarjeta especificado.

Sintaxis #include "LTC2x.H"

BOOL **C3CARDINSERTED**(HANDLE hCommDev, int nCCA)

hCommDev Identificador del puerto.

nCCA Contactor de tarjeta (0 = primero)

Respuesta Si el valor devuelto es ? 0 (True) Hay tarjeta insertada

Si el valor devuelto es = 0 (False) No hay tarjeta insertada

Notas En el caso del LTC31 al disponer de un solo contactor de tarjeta

nCCA siempre será 0.

Esta función tan sólo consulta el estado de la línea indicadora de presencia de tarjeta del puerto serie, minimizando el uso de recursos del ordenador y su respuesta es inmediata.

C3READERPOWERED

Descripción Devuelve el estado del indicador de presencia y alimentación del lector de tarjeta conectado al puerto serie propietario del HANDLE especificado.

Sintaxis #include "LTC2x.H"

BOOL **C3READERPOWERED**(HANDLE hCommDev)

hCommDev Identificador del puerto.

Respuesta Si el valor devuelto es? 0 (True) se ha detectado un lector conectado y alimentado al puerto.

Si el valor devuelto es = 0 (False) no se ha detectado ningún lector en el puerto

Notas Con esta función puede darse el caso de que hubiera algún dispositivo conectado al puerto serie, distinto de lector de tarjeta chip, y que lo identificara como tal.

Características físicas

Dimensiones

Altura: 12.5 mm

Ancho: 60.0 mm

Largo: 50.0 mm

Peso: 90 gr.

Material Carcasa

Frontal: Plástico ABS UL94 V0

Cuerpo: Plástico ABS UL94 V0

Comando de comunicación

Escritura de la memoria principal

SINTAXIS

00 D6 00 P2 P3 d0 d1...d(P3-1)

DESCRIPCIÓN

Escribe los P3 bytes d0 a d(P3-1) en la Memoria Principal de la tarjeta desde la posición determinada por P2

RESPUESTA

00 90 00 00

ACCESO

Presentación previa de la Clave. Si no se ha presentado la clave correcta con anterioridad la respuesta será la misma por lo que sólo podremos comprobar la operación tras una lectura posterior y su comprobación.

NOTA

El tamaño de la memoria principal de esta tarjeta es de 256 bytes. Los primeros 32 bytes normalmente se emplean para la identificación de la tarjeta y su aplicación. Puede protegerse su escritura con la memoria de protección.

Lectura de memoria de protección

SINTAXIS

00 CA 01 00 04

DESCRIPCIÓN

Devuelve los 4 bytes que componen la memoria de protección

RESPUESTA

00 90 00 04 d0 d1 d2 d3

ACCESO

Libre

NOTA

El tamaño de la memoria de protección de esta tarjeta es de 4 bytes. Estos 32 bits corresponden a los primeros 32 bytes de la memoria principal. Si el bit está a 0, el byte correspondiente de la memoria principal está permanentemente protegido.

Lectura de la memoria principal

SINTAXIS

00 B0 00 P2 P3

DESCRIPCIÓN

Devuelve P3 bytes del contenido de la Memoria Principal de la tarjeta desde la posición determinada por P2

RESPUESTA

00 90 00 LEN d0 d1...d(len-1)

ACCESO

Libre

NOTA

El tamaño de la memoria principal de esta tarjeta es de 256 bytes. Los primeros 32 bytes normalmente se emplean para la identificación de la tarjeta y su aplicación. Puede protegerse su escritura con la memoria de protección.

Escritura de memoria de protección

SINTAXIS

00 DA 01 00 20 d0 d1 d2 d3

DESCRIPCIÓN

Escribe los 32 bits en la memoria de protección protegiendo así la memoria principal. Si el bit 1 de la memoria de protección está a 1, el byte 1 de la memoria principal y así sucesivamente.

RESPUESTA

00 90 00 00

ACCESO

Presentación previa de la Clave

NOTA

El tamaño de la memoria principal que puede ser protegida de esta tarjeta es de 32 bytes.

ESTRUCTURA DE UNA TARJETA INTELIGENTE

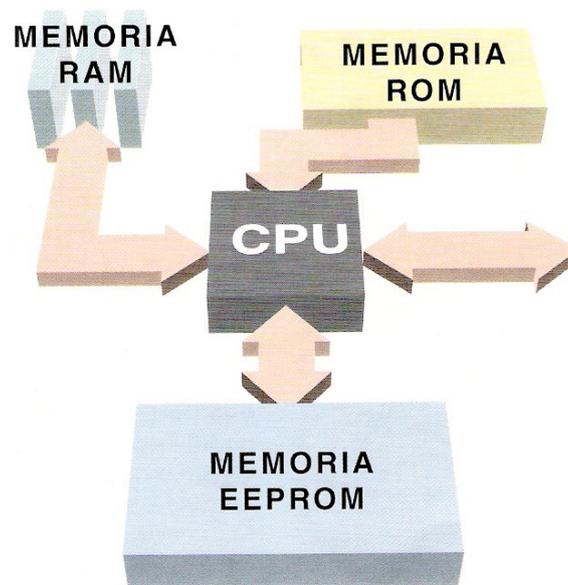
Internamente, el chip de una tarjeta inteligente microcontrolada se compone de:

- **CPU** (*Unidad de procesamiento central*): el procesador de la tarjeta; suele ser de 8 bits, a 5 MHz. y 5 voltios. Pueden tener opcionalmente módulos hardware para operaciones criptográficas.
- **ROM** (*Memoria de sólo lectura*): memoria interna (normalmente entre 12 y 30 KB) en la que se incrusta el sistema operativo de la tarjeta, las rutinas del protocolo de comunicaciones y los algoritmos de seguridad 28 de alto nivel por

software. Esta memoria, como su nombre indica, no se puede reescribir y se inicializa durante el proceso de fabricación.

- **EEPROM**: memoria de almacenamiento en el que está grabado el sistema de ficheros, los datos usados por las aplicaciones, claves de seguridad y las propias aplicaciones que se ejecutan en la tarjeta. El acceso a esta memoria está protegido en distintos niveles por el sistema operativo de la tarjeta.

- **RAM** (*Memoria de acceso aleatorio*): memoria volátil para el trabajo en el procesador.



Contacto Designación Uso

C1 VCC

Conector de energía a través del cual se provee energía al chip microprocesador en la tarjeta.

C2 RST

Línea a través de la cual se le puede enviar una señal al chip microprocesador de la tarjeta inteligente para iniciar la secuencia de instrucciones de reinicialización (reset).

C3 CLK

Línea de señal de reloj. Esta línea controla la velocidad de operación y provee un marco de trabajo común para la comunicación de datos entre el circuito integrado de la tarjeta y el lector.

C4 RFU

Reservado para uso futuro.

C5 GND

Línea que provee un punto eléctrico común entre el circuito integrado de la tarjeta y el lector.

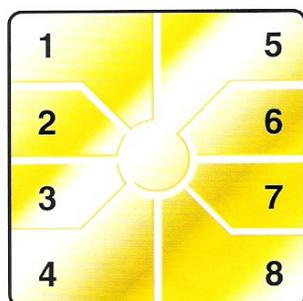
C6 VPP

Conector de energía de programación usado para programar la memoria **EEPROM del circuito** integrado de la tarjeta.

C7 I/O

Línea de entrada / salida que provee un canal de comunicación bidireccional no simultánea entre el lector y la tarjeta inteligente.

C8 RFU Reservado para uso futuro.



| | | | |
|---|-------|---|-------|
| 1 | Vcc | 1 | Vcc |
| 2 | Reset | 2 | In |
| 3 | CLK | 3 | CLK |
| 4 | Nc | 4 | Reset |
| 5 | GND | 5 | GND |
| 6 | Nc | 6 | Vpp |
| 7 | I/O | 7 | Out |
| 8 | Nc | 8 | Nc |