**Capítulo 2: Planificación y Diseño del Sistema**

**2.1 Introducción al capitulo**

En este capítulo se abordan detalladamente las fases de planificación y diseño según los módulos a implementar; mediante la puesta en ejecución de la Metodología de Desarrollo Mobile-D que es un enfoque de desarrollo ágil basado en Extreme Programación (XP). Se plantean las reglas del negocio, así como la propuesta del sistema, sus usuarios, roles y responsabilidades. Se declaran las funcionalidades y las historias técnicas que debe cumplir el sistema y se describen las historias de usuario. Se relacionan todas las clases a través de un diagrama de clase generado a través de la herramienta Visual Paradigm. El modelo de datos queda reflejado mediante el diagrama físico de la base de datos.

**2.2 Objetivo a alcanzar**

Una vez instalada la aplicación móvil tanto en dispositivos **Android** como **iOS** permitirá la comercialización de los productos agrícolas en los distintos Mercados de la Ciudad Santiago de Cuba entre el consumidor y el vendedor, de tal forma que el vendedor pueda publicar de forma sencilla toda la información necesaria de sus productos en dichos mercados para que sus clientes sepan que es lo que oferta, a la ves el cliente pueda encontrar el producto de forma sencilla en cualquier mercado de la Ciudad Santiago de Cuba.

**2.3 Características**

La aplicación móvil trabajará en base a los mercados, permitirá la negociación de la venta y compra de productos agrícolas entre los diferentes usuarios tanto aquel usuario que se registra para poder vender sus productos como para aquel usuario que solo desea comprar lo cual no es necesario el registro. Los roles de comprador y vendedor le serán asignados en dependencia de su registro en la aplicación. Un usuario registrado en la aplicación móvil solo podrá desempeñar el rol de vendedor, permitiendo al usuario solo publicar y vender sus productos en caso de que no se registre solo podrá comprar productos.

* 1. **Usuarios del Sistema**En la aplicación móvil se definen dos tipos de usuarios, cada uno con diferentes privilegios. En la Tabla 1.2 que a continuación se muestra, aparecen detallados cada uno de ellos.

**Tabla 2.1:** Usuarios del Sistema

|  |  |
| --- | --- |
| Usuario | Descripción |
| Comprador | Usuario el cual no necesita registrarse cuya característica principal es la capacidad de realizar compras de los productos agrícolas ofrecidos en los mercados de la aplicación móvil. |
| Vendedor | Usuario registrado cuya característica principal es la capacidad de publicar sus productos agrícolas, con el fin de ser ofertados en los mercados de la aplicación móvil. |

**2.5 Requisitos Funcionales del Sistema (RF).**

Una funcionalidad define el comportamiento interno del software: cálculos, detalles técnicos, manipulación de datos y otras características específicas que muestran cómo los procesos serán llevados a la práctica. En la metodología de desarrollo utilizada Mobile-D, estos requisitos van a ser descritos por medio de las Historias de Usuarios (HU).

* RF\_1: Autenticación: ingreso de credenciales de usuario para acceder a la aplicación móvil, el usuario existe en storage(Almacenamiento): [SI] accederá a la aplicación, [NO] le mostrara un mensaje de error por lo que tendrá que ir a la ventana de registro en caso de que no tenga credenciales.
* RF\_2: Registro de nuevos usuarios: Permitir la captura de los datos correspondientes a la información del usuario, con el fin de realizar el registro y permitir su ingreso al contenido de la aplicación móvil.
* RF\_3: Menú Drawer  (Cajón alboroto): Una vez completada la autenticación del usuario, tendrá a disposición el menú de opciones de las diferentes actividades que podrá realizar.
* RF\_4: Insertar datos de usuario: Permitir al usuario vendedor, realizar la inserción de sus datos personales.
* RF\_5: Publicación de nuevos productos: Cada usuario con rol de vendedor podrá crear productos, los cuales tendrán que ser subidos al mercado el cual pertenece.
* RF\_6: Listado de productos: Cada usuario con rol de vendedor podrá acceder al listado de sus productos creados, tendrá un icono de asociación de eliminación (bote de basura) y uno de edición (Lápiz) que al seleccionar el producto permitirá visualizar la información del producto, modificar el producto.
* RF\_7: Mercado, listado de productos publicados: Cada usuario sin exclusión de rol, podrá visualizar los productos publicados mediante una lista con vista previa.

* RF\_8: Mercado, visualización del detalle del producto: Permitirá al usuario con el rol de comprador visualizar la información del producto y posteriormente si lo desea agregarlo al carrito.
* RF\_9: Carrito de compras: Te permite visualizar el listado de productos en el carrito de compra para posterior mente si lo deseas realizar las reservas de dichos productos.
* RF\_10: Listado de compras: Los usuarios con el rol de comprador podrán visualizar el detalle de los productos reservados por WhatsApp.
* RF\_11: Listado de ventas: Los usuarios con el rol de vendedor podrán visualizar el detalle de las ventas realizadas.

**2.6 Historias Técnicas**

Las funcionalidades son complementadas por las historias técnicas que se enfocan en el diseño o la implementación. Las historias técnicas constituyen las propiedades o cualidades que el producto debe tener. Ellas representan las características que deben hacer al producto atractivo, usable, rápido y confiable, lo que hace que el éxito del producto dependa de estas características.

* Eficiencia: Toda funcionalidad y transacción de la aplicación móvil debe responder al usuario en menos de un segundo.
* Seguridad lógica y de datos: Los usuarios externos a la aplicación móvil carecerán de la potestad de leer o escribir datos en la base de datos y del acceso a la aplicación móvil. Los usuarios registrados tendrán limitaciones al leer o escribir datos en la base de datos.
* Usabilidad: El tiempo de aprendizaje del uso de la aplicación móvil deberá ser menor a tres horas, proporcionar mensajes de error informativo y orientado al usuario.
* Disponibilidad: La aplicación móvil tendrá una disponibilidad del 99,99% cuando un usuario desee acceder o ejecutar cualquier tipo de transición.
* Funcionabilidad: La aplicación móvil será compatible con sistema operativo Android y IOS, en Android versión 5 en adelante y en iOS.
* Mantenibilidad: Permitir la escalabilidad de la aplicación móvil.
  1. **Historias de Usuario.**

Las HU son la técnica utilizada en Mobile-D para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales [25]. El tratamiento de las historias de usuario es muy dinámico y flexible, en cualquier momento las HU pueden romperse, reemplazarse por otras más específicas o generales, añadirse nuevas o ser modificadas. Cada HU es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas [26]. A través de las HU (artefacto de la metodología Mobile-D) se pretende describir los requisitos funcionales del sistema. La prioridad de las HU se clasificó de la siguiente manera:

➢ **Alto**: requerimientos que no pueden faltar.

➢ **Medio**: requerimientos que pueden ser implementados en otra iteración.

➢ **Bajo**: requerimientos adicionales.

**Tabla 1** Historia de usuario RF1 “Autenticación”

|  |  |  |  |
| --- | --- | --- | --- |
| HISTORIA DE USUARIO | | | |
| Número:1 | **Nombre de Historia de Usuario:** Autenticación | | |
| Modificación de Historia de Usuario Número: ninguna | | | |
| Usuario: Vendedor | | Iteración Asignada: 1 | |
| Prioridad en Negocio: Alto  (Alto/ Medio/ Bajo) | | Puntos Estimados:2 semanas | |
| Riesgo en Desarrollo: Alto  (Alto/ Medio/ Bajo) | | Puntos Reales: 1 ½ semana | |
| Descripción: Ingreso de credenciales de usuario para acceder a la aplicación móvil, el usuario existe en storage(Almacenamiento): [SI] accederá a la aplicación, [NO] le mostrara un mensaje de error por lo que tendrá que ir a la ventana de registro en caso de que no tenga credenciales. | | | |
| Prototipo de Interfaz Gráfica | | | |
| Observación: En caso de dejar los campos vacíos muestra los mensajes de alerta. | | | |
| Estado | | | Completado satisfactoriamente. |
| Prototipo de Interfaz Gráfica | | | |
| Observación: En caso de ingresar credenciales que no se encuentren en la base de datos, deberá ir al registro | | | |
| Estado | | | En desarrollo |

**Tabla 2** Historia de usuario RF2 “Registro de nuevos usuarios”

|  |  |  |
| --- | --- | --- |
| HISTORIA DE USUARIO | | |
| Número:1 | **Nombre de Historia de Usuario:** Registro de nuevos usuarios | |
| Modificación de Historia de Usuario Número: ninguna | | |
| Usuario: Vendedor | | Iteración Asignada: 2 |
| Prioridad en Negocio: Alto  (Alto/ Medio/ Bajo) | | Puntos Estimados:1 semanas |
| Riesgo en Desarrollo: Alto  (Alto/ Medio/ Bajo) | | Puntos Reales: 1 ½ semana |
| Descripción: Permitir la captura de los datos correspondientes a la información del usuario, con el fin de realizar el registro y permitir su ingreso al contenido de la aplicación móvil. | | |
| Prototipo de Interfaz Gráfica | | |
| Estado | | En desarrollo |
|  | | |
| Observaciones: En caso de cumplir con los datos obligatorios con el formato requerido, se registrara usuario y posteriormente puedes ingresar en el login para confirmar registro para así acceder a la aplicación. | | |
| Estado | | En desarrollo |

**Tabla 3** Historia de usuario RF3 “Menú Drawer  (Cajón alboroto)”

|  |  |  |  |
| --- | --- | --- | --- |
| HISTORIA DE USUARIO | | | |
| Número:1 | **Nombre de Historia de Usuario:** Menú Drawer  (Cajón alboroto) | | |
| Modificación de Historia de Usuario Número: ninguna | | | |
| Usuario: Vendedor | | Iteración Asignada: 1 | |
| Prioridad en Negocio: Alto  (Alto/ Medio/ Bajo) | | Puntos Estimados:1 semanas | |
| Riesgo en Desarrollo: Alto  (Alto/ Medio/ Bajo) | | Puntos Reales: 1 ½ semana | |
| Descripción: Una vez completada la autenticación del usuario, tendrá a disposición el menú de opciones de las diferentes actividades que podrá realizar. | | | |
| Prototipo de Interfaz Gráfica | | | |
| Observaciones: En caso de que los datos ingresados a través del formulario no tener el formato requerido, la aplicación muestra mensajes de alerta. | | | |
| Estado | | | Completado satisfactoriamente |

**Tabla 4** Historia de usuario RF4 “Insertar de datos de usuario”

|  |  |  |
| --- | --- | --- |
| HISTORIA DE USUARIO | | |
| Número:1 | **Nombre de Historia de Usuario:** Insertar de datos de usuario | |
| Modificación de Historia de Usuario Número: ninguna | | |
| Usuario: Vendedor | | Iteración Asignada: 1 |
| Prioridad en Negocio: Alto  (Alto/ Medio/ Bajo) | | Puntos Estimados:1 semanas |
| Riesgo en Desarrollo: Alto  (Alto/ Medio/ Bajo) | | Puntos Reales: 1 ½ semana |
| Descripción: Permitir al usuario vendedor, realizar la inserción de sus datos personales. | | |
| Prototipo de Interfaz Gráfica | | |
| Observación | | Insertar foto de perfil, nombre o apodo, número de teléfono |
| Estado | | En desarrollo |

**Tabla 5** Historia de usuario RF5 “Publicación de nuevos productos”

|  |  |  |
| --- | --- | --- |
| HISTORIA DE USUARIO | | |
| Número:1 | **Nombre de Historia de Usuario:** Publicación de nuevos productos | |
| Modificación de Historia de Usuario Número: ninguna | | |
| Usuario: Vendedor | | Iteración Asignada: 1 |
| Prioridad en Negocio: Alto  (Alto/ Medio/ Bajo) | | Puntos Estimados:1 semanas |
| Riesgo en Desarrollo: Alto  (Alto/ Medio/ Bajo) | | Puntos Reales: 1 ½ semana |
| Descripción: Cada usuario con rol de vendedor podrá crear productos, los cuales tendrán que ser subidos al mercado el cual pertenece. | | |
| Prototipo de Interfaz Gráfica | | |
| Observación | | En caso de dejar los campos vacíos muestra el mensaje de alerta |
| Estado | | Completado |
| Prototipo de Interfaz Gráfica | | |
| Observación | | En caso de cumplir con los campos obligatorios y el formato requerido, se guarda la información en la base de datos SQFLITE que se encuentra en el dispositivo móvil. |
| Estado | | Completado |

**Tabla 6** Historia de usuario RF6 “Listado de productos”

|  |  |  |
| --- | --- | --- |
| HISTORIA DE USUARIO | | |
| Número:1 | **Nombre de Historia de Usuario:** Listado de productos | |
| Modificación de Historia de Usuario Número: ninguna | | |
| Usuario: Vendedor | | Iteración Asignada: 1 |
| Prioridad en Negocio: Alto  (Alto/ Medio/ Bajo) | | Puntos Estimados:1 semanas |
| Riesgo en Desarrollo: Alto  (Alto/ Medio/ Bajo) | | Puntos Reales: 1 ½ semana |
| Descripción: Cada usuario con rol de vendedor podrá acceder al listado de sus productos creados, tendrá un icono de asociación de eliminación (bote de basura) y uno de edición (Lápiz) que al seleccionar el producto permitirá visualizar la información del producto, modificar el producto en caso de que lo requiera. | | |
| Prototipo de Interfaz Gráfica | | |
| Observación | | En caso de dejar los campos vacíos muestra el mensaje de alerta |
| Estado | | Completado |
| Prototipo de Interfaz Gráfica | | |
| Observación | | En caso de cumplir con los campos obligatorios y el formato requerido, se actualiza la información en la base de datos SQFLITE que se encuentra en el dispositivo móvil. |
| Estado | | Completado |

**Tabla 7** Historia de usuario RF7 “Mercado, listado de productos publicados”

|  |  |  |
| --- | --- | --- |
| HISTORIA DE USUARIO | | |
| Número:1 | **Nombre de Historia de Usuario:** Mercado, listado de productos publicados | |
| Modificación de Historia de Usuario Número: ninguna | | |
| Usuario: Vendedor | | Iteración Asignada: 1 |
| Prioridad en Negocio: Alto  (Alto/ Medio/ Bajo) | | Puntos Estimados:1 semanas |
| Riesgo en Desarrollo: Alto  (Alto/ Medio/ Bajo) | | Puntos Reales: 1 ½ semana |
| Descripción: Cada usuario sin exclusión de rol, podrá visualizar los productos publicados mediante una lista con vista previa. | | |
| Prototipo de Interfaz Gráfica | | |
| Estado | | Completado |

**Tabla 8** Historia de usuario RF8 “Mercado, visualización del detalle del producto”

|  |  |  |
| --- | --- | --- |
| HISTORIA DE USUARIO | | |
| Número:1 | **Nombre de Historia de Usuario:** Mercado, listado de productos publicados | |
| Modificación de Historia de Usuario Número: ninguna | | |
| Usuario: Vendedor | | Iteración Asignada: 1 |
| Prioridad en Negocio: Alto  (Alto/ Medio/ Bajo) | | Puntos Estimados:1 semanas |
| Riesgo en Desarrollo: Alto  (Alto/ Medio/ Bajo) | | Puntos Reales: 1 ½ semana |
| Descripción: Permitirá al usuario con el rol de comprador visualizar la información del producto y posteriormente si lo desea agregarlo al carrito. | | |
| Prototipo de Interfaz Gráfica | | |
| Estado | | Completado |

**Tabla 9** Historia de usuario RF9 “Carrito de compras”

|  |  |  |
| --- | --- | --- |
| HISTORIA DE USUARIO | | |
| Número:1 | **Nombre de Historia de Usuario:** Carrito de compras | |
| Modificación de Historia de Usuario Número: ninguna | | |
| Usuario: Vendedor | | Iteración Asignada: 1 |
| Prioridad en Negocio: Alto  (Alto/ Medio/ Bajo) | | Puntos Estimados:1 semanas |
| Riesgo en Desarrollo: Alto  (Alto/ Medio/ Bajo) | | Puntos Reales: 1 ½ semana |
| Descripción: Te permite visualizar el listado de productos en el carrito de compra para posterior mente si lo deseas realizar las reservas de dichos productos. | | |
| Prototipo de Interfaz Gráfica | | |
| Estado | | En desarrollo |

**Tabla 10** Historia de usuario RF10 “Listado de compras”

|  |  |  |
| --- | --- | --- |
| HISTORIA DE USUARIO | | |
| Número:1 | **Nombre de Historia de Usuario:** Listado de compras | |
| Modificación de Historia de Usuario Número: ninguna | | |
| Usuario: Vendedor | | Iteración Asignada: 1 |
| Prioridad en Negocio: Alto  (Alto/ Medio/ Bajo) | | Puntos Estimados:1 semanas |
| Riesgo en Desarrollo: Alto  (Alto/ Medio/ Bajo) | | Puntos Reales: 1 ½ semana |
| Descripción: Los usuarios con el rol de comprador podrán visualizar el detalle de los productos reservados por WhatsApp. | | |
| Prototipo de Interfaz Gráfica | | |
| Estado | | En desarrollo |

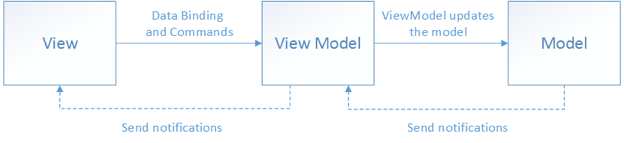
**Tabla 11** Historia de usuario RF11 “Listado de Ventas”

|  |  |  |
| --- | --- | --- |
| HISTORIA DE USUARIO | | |
| Número:1 | **Nombre de Historia de Usuario:** Listado de ventas | |
| Modificación de Historia de Usuario Número: ninguna | | |
| Usuario: Vendedor | | Iteración Asignada: 1 |
| Prioridad en Negocio: Alto  (Alto/ Medio/ Bajo) | | Puntos Estimados:1 semanas |
| Riesgo en Desarrollo: Alto  (Alto/ Medio/ Bajo) | | Puntos Reales: 1 ½ semana |
| Descripción: Los usuarios con el rol de vendedor podrán visualizar el detalle de las ventas realizadas. | | |
| Prototipo de Interfaz Gráfica | | |
| Estado | | En desarrollo |

* 1. **Patrón Arquitectónico empleado: MVVM o Modelo-Vista-Vista Modelo.**

El patrón MVVM o Modelo-Vista-VistaModelo (Model-View-ViewModel en inglés) es un patrón que ayuda a desacoplar o separar al máximo la interfaz de usuario de la lógica de la aplicación. Esto facilita tanto el mantenimiento como las pruebas y la escalabilidad del proyecto. Sus tres componentes principales definen el nombre del patrón: el modelo, la vista y el modelo de vista. Vamos a detallar un poco más cada uno de ellos.

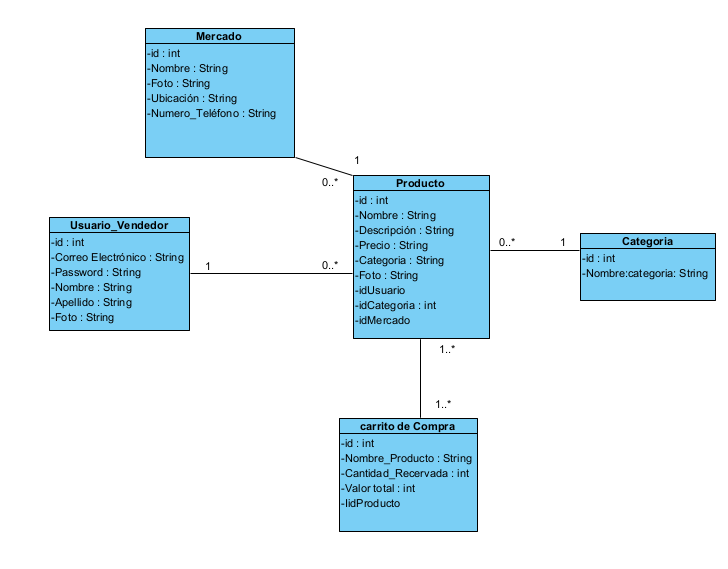
* **Model**: representa la capa de los datos. Contiene la información pero no las acciones o servicios que lo manipulan.
* **View**: representa la interfaz de usuario. Contiene la parte visual y las interacciones con el usuario a través de animaciones, eventos, etc.
* **ViewModel**: será el intermediario entre los datos y la interfaz. Contiene la lógica de presentación y las operaciones con los datos del modelo.



Existen dos reglas fundamentales de este patrón:

* Cada vista vendrá siempre acompañada por su modelo de vista.
* La vista sólo conoce a su modelo de vista y este al modelo, la vista no conoce al modelo.

**2.9 Diagrama de clases del diseño.**



**2.10. Diagrama de despliegue**

El diagrama de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. Se utiliza como entrada fundamental en las actividades de diseño e implementación debido a que la distribución del sistema tiene una influencia principal en su diseño[27].

Un diagrama de despliegue es un tipo de diagrama UML que muestra la arquitectura de ejecución de un sistema, incluyendo nodos como entornos de ejecución de hardware o software, y el middleware que los conecta. Los diagramas de despliegue ayudan a modelar la topología de hardware de un sistema en comparación con otros tipos de diagramas UML, que en su mayoría esbozan los componentes lógicos de un sistema.[28]

**Nodos**: Un nodo, representado como un cubo, es una entidad física que ejecuta uno o más componentes, subsistemas o ejecutables. Un nodo podría ser un elemento de hardware o software[27].

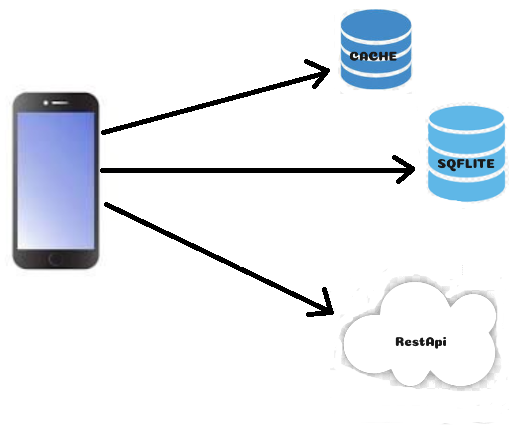
**Artefactos**: Los artefactos son elementos concretos que son causados por un proceso de desarrollo. Ejemplos de artefactos son las bibliotecas, archivos, archivos de configuración, archivos ejecutables, etc[27].

**Asociación de Comunicación**: Esto está representado por una línea sólida entre dos nodos. Muestra el camino de la comunicación entre los nodos[27].

**Dispositivos**: Un dispositivo es un nodo que se utiliza para representar un recurso físico computacional en un sistema. Un ejemplo de un dispositivo es un servidor de aplicaciones[27].

**Especificaciones de despliegue**: Especificación despliegue Las especificaciones de despliegue son un archivo de configuración, como un archivo de texto o un documento XML. Describe cómo se despliega un artefacto en un nodo[27].

A partir de lo referenciado anteriormente se ha confeccionado el siguiente diagrama de despliegue para la Aplicaciónn móvil de compra y venta en mercados agropecuarios en la Ciudad de Santiago de Cuba. Figura 2.12.1 Diagrama de Despliegue.



**2.11. Servicios Web.**

En términos técnicos, un servicio web es una aplicación diseñada para establecer comunicación con algún otro programa, para eso se establece reglas de comunicación como las direcciones para encontrar los recursos, qué acciones se pueden realizar con dichos recursos, y cómo se va a estructurar el texto que se intercambia entre ambos programas.[14]

Un servicio web, no hace asunciones acerca de la tecnología que usarán los clientes para consumir la información, así, bien el servidor puede estar programado con JAVA y el cliente usar JavaScript, o cualquier otro tipo de tecnología. Es por esto que se establece un formato para el texto que sea estándar, este puede ser una estructura JSON, XML, entre otros.[15]

En términos prácticos, un servicio web está diseñado para intercambiar mensajes con otra aplicación, permitiéndonos enviar y recibir información.

Para establecer las reglas de comunicación, un servicio web puede usar algún protocolo de comunicación como graphql, REST: (en inglés Representational State Transfer; en español Transferencia de Estado Representacional) y SOAP: (siglas en inglés de Simple Object Access Protocol; traducido como Protocolo Simple de Acceso a Objetos), cada uno con sus respectivas ventajas y desventajas. Estos protocolos establecen una normativa de cómo intercambiarán mensajes el cliente y el servidor. Una vez que se ha elegido el protocolo, los y las desarrolladoras implementan sus aplicaciones siguiendo estas reglas para el envío y la lectura de mensajes.

Graphql: es un lenguaje de consulta y manipulación de datos para APIs, y un entorno de ejecución para realizar consultas con datos existentes

REST es un estilo de arquitectura de software para sistemas hipermedia distribuidos como la World Wide Web, empleado para describir cualquier interfaz entre sistemas que utilice directamente HTTP para obtener datos o indicar la ejecución de operaciones sobre los datos, en cualquier formato (XML, JSON.

SOAP constituye un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. Es un paradigma de mensajería de una dirección sin estado, que puede ser utilizado para formar protocolos más complejos y completos según las necesidades de las aplicaciones que lo implementan.

**JSON (Notación de objetos de JavaScript)** es un formato ligero de intercambio de datos y es completamente independiente del lenguaje. Está basado en el lenguaje de programación JavaScript y es fácil de entender y generar.

**Ejemplo :**

|  |
| --- |
| {"Producto":[      { "Boniato":"vianda", "Buena para acompañar la comida":"50" },      { "Mango":"fruta", "Buena para jugo":"30" },          { "Arroz":"grano", "Indispensable en la comida del cubano":"80" },  ]} |

**XML (lenguaje de marcado extensible)** fue diseñado para transportar datos, no para mostrar datos. Es una recomendación del W3C. El lenguaje de marcado extensible (XML) es un lenguaje de marcado que define un conjunto de reglas para codificar documentos en un formato que es tanto legible por humanos como por máquinas. Los objetivos de diseño de XML se centran en la simplicidad, la generalidad y la facilidad de uso en Internet. Es un formato de datos textuales con fuerte soporte a través de Unicode para diferentes idiomas humanos. Aunque el diseño de XML se centra en los documentos, el lenguaje se utiliza ampliamente para la representación de estructuras de datos arbitrarias, como las que se utilizan en los servicios web.  
**Ejemplo :**

|  |
| --- |
| <Geeks>      <Geek>          <firstName>Vivek</firstName> <lastName>Kothari</lastName>      </Geek>      <Geek>          <firstName>Suraj</firstName> <lastName>Kumar</lastName>      </Geek>    </Geeks>  **Servicios web requeridos por la aplicación** |

**Productos:** Crear producto, actualizar producto, eliminar producto, listar producto.

Ejemplo de Json admitidos.

{  
    "ArrProductos":[{  
            "nombre”: “Boniato",  
            "precio":"50lb"

"descripción": “Bueno para acompañar con los alimentos"

"categoría”: "vianda"

        },  
        {  
           "nombre": "Mango",  
            "precio":"20lb"

"descripción": “Bueno para jugo"

"categoría": "fruta"

        },  
        {  
            "nombre": "Arroz",  
            "precio":"80lb"

"descripción”: “Indispensable en la comida cubana"

"categoría”: “Grano"

        },  
        {  
 "nombre":"Papa",  
            "precio":"100lb"

"descripción":"Bueno para acompañar con los alimentos"

"categoria":"vianda"

        },  
         
  
    ]  
}

**Mercados**: Listar mercados y productos y productos existente en dicho mercado con sus datos nombre, precio, descripción, categoría, foto.