



Colegio Universitario de Cartago

Tecnologías de la Información

Curso: Programación IV

Profesor: Gabriel Gonzalez Solano

Proyecto Pagos de Servicios

Estudiantes: Dahiana Solano Mena

Deivy Kael Sandi Durán

Emmanuel Arce Cascante

José Daniel Calderón Solano

CARTAGO, COSTA RICA, CUATRIMESTRE II 2024

Introducción

La solución de un software tiene que cada vez mucha importancia en el ámbito de la tecnología, la tendencia mundial es de mejorar y a facilitar el uso que le puede dar al usuario. Para la población en general siempre va a hacer de gran ayuda tener a mano herramientas, que le pueda dar mejores experiencias y favorecer a una mejorar calidad de vida.

El mundo de la tecnología está cada vez más orientado en hacer la vida de las personas más rápida, sencilla y segura en el uso de programas o software, muchas empresas u organizaciones realizan software que implementan seguridad en sus programas. Por ejemplo, en este caso, que se asigna el tema de elaborar una solución de software que permita realizar pagos de diferentes tipos de servicios para ofrecer en distintos comercios afiliados diseñada para que los usuarios puedan permita que los clientes puedan realizar los pagos de luz, teléfono, electricidad entre otros, usando sus tarjetas o bien sus cuentas bancarias, mediante mecanismos de seguridad que le permitirá certificar su identidad, en donde es importante tener a la mano herramientas en donde procesar información con sus debidos métodos de seguridad. Un punto importante es el desarrollo del software, se puede utilizar para múltiples funciones, va a depender del usuario final que es lo que desea.

SUPUESTOS

A continuación, se presentan los posibles supuestos sobre el contexto del proyecto:

1. **Cumplimiento de requisitos:** Se asume que la solución de software de la empresa Pagos Express cumplirá con las regulaciones y estándares de seguridad establecidos en el documento de historias de usuario con las indicaciones dadas.
2. **Aceptación del usuario final:** Se asume que los usuarios finales, es decir, los clientes que utilizarán la solución de software de la empresa Pagos Express,

encontrarán la interfaz fácil de usar y estarán dispuestos a realizar sus pagos de servicios en el momento que desean.

3. **Estándares de comunicación:** Se asume que se establecerán estándares de comunicación efectivos entre los diferentes sistemas de lenguajes de programación y bases de datos utilizados en el software, para garantizar una comunicación fluida y segura. Además, que lo mencionado anteriormente cumpla a cabalidad con los requisitos nombrados en el documento de historias de usuario.

DIAGRAMA DE BASES DE DATOS

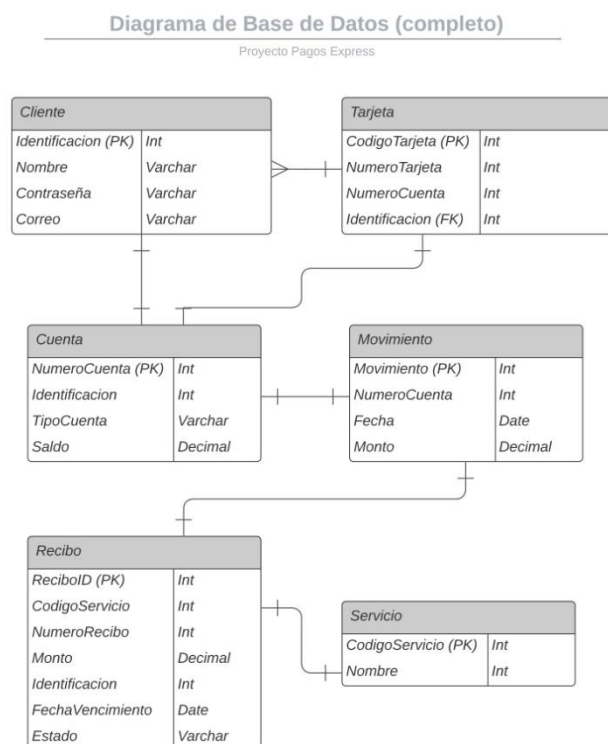


DIAGRAMA DE CASOS DE USO

Diagrama de caso de uso-Historia de usuario 1

June 10, 2024

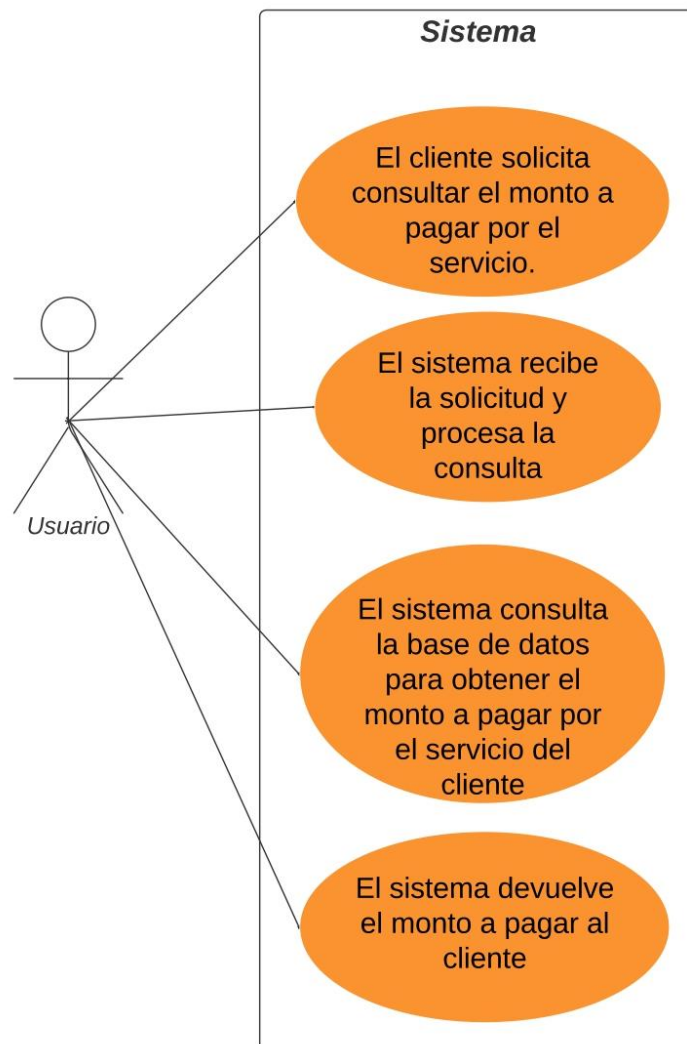


Diagrama de caso de uso-Historia de usuario 2

June 10, 2024

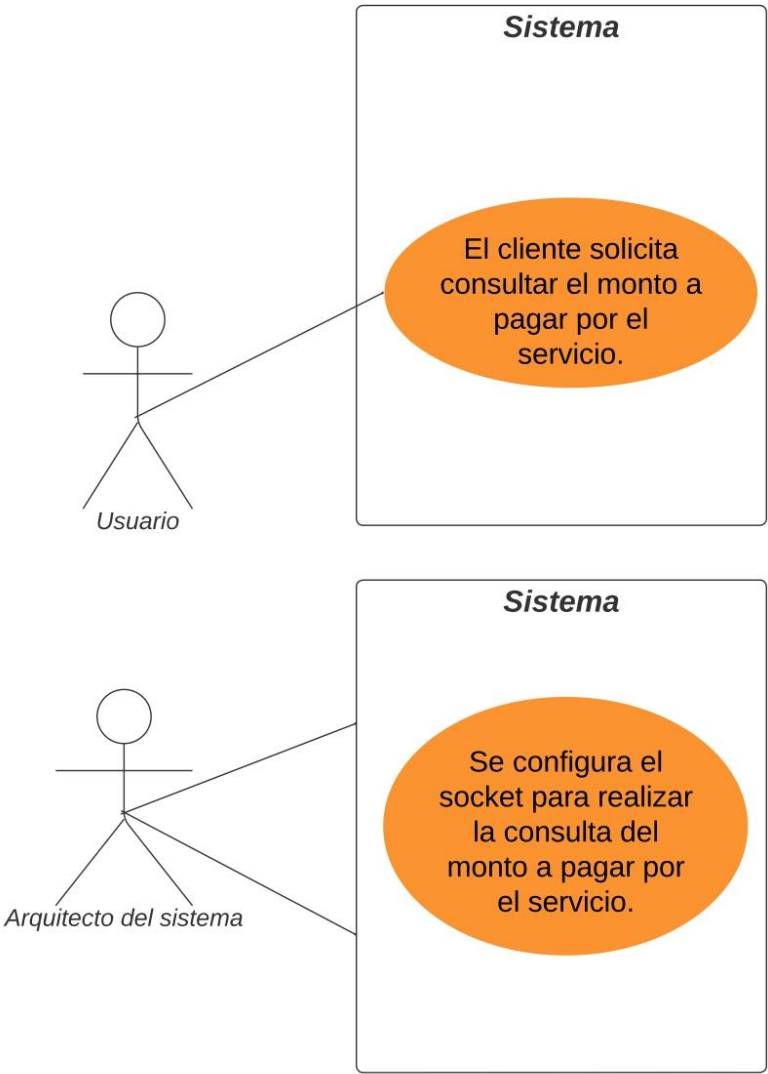


Diagrama de caso de uso-Historia de usuario 3

June 10, 2024

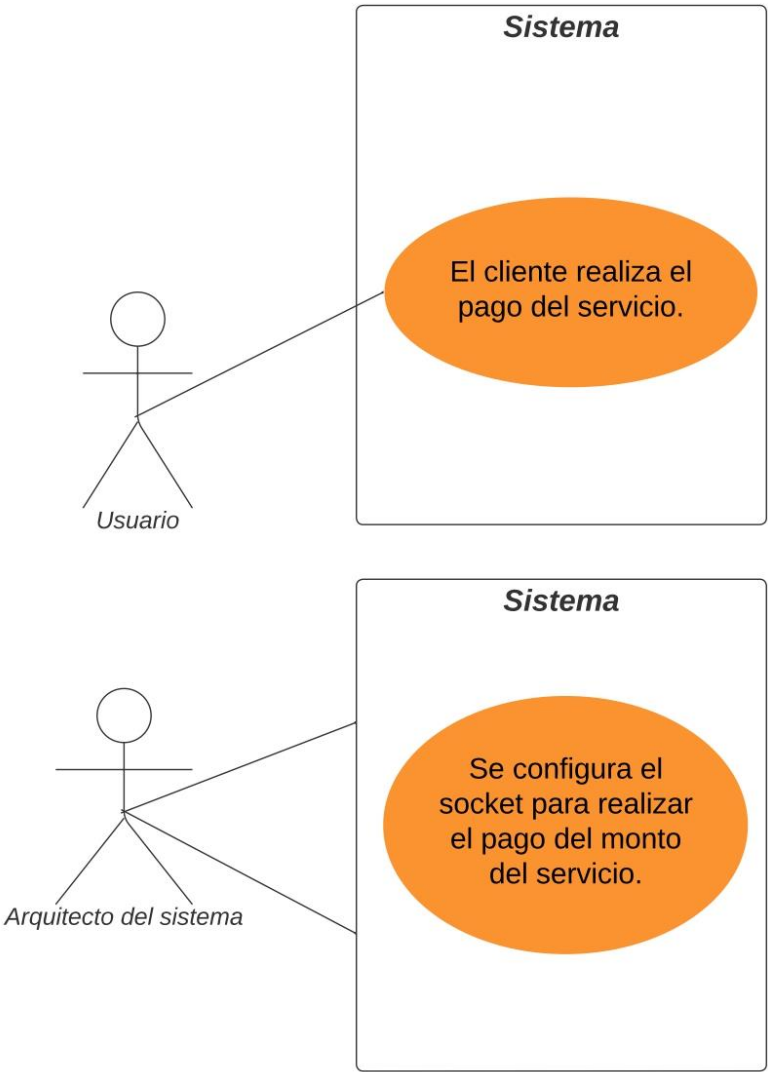


Diagrama de caso de uso-Historia de usuario 4

June 10, 2024

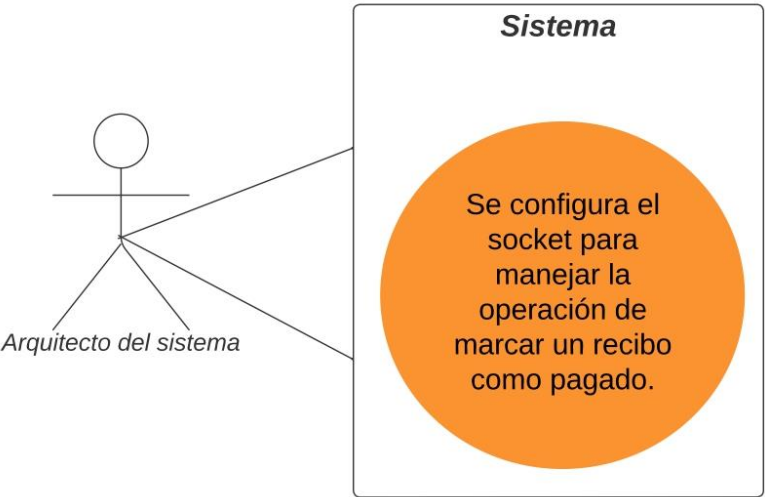
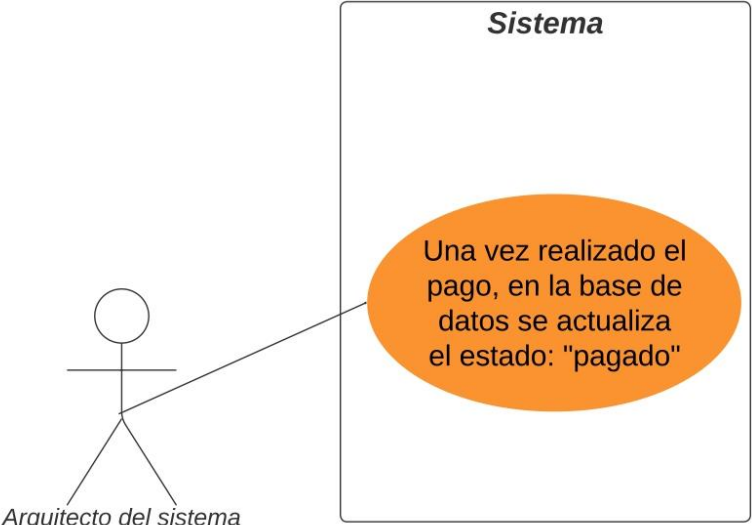


Diagrama de caso de uso-Historia de usuario 5

June 10, 2024

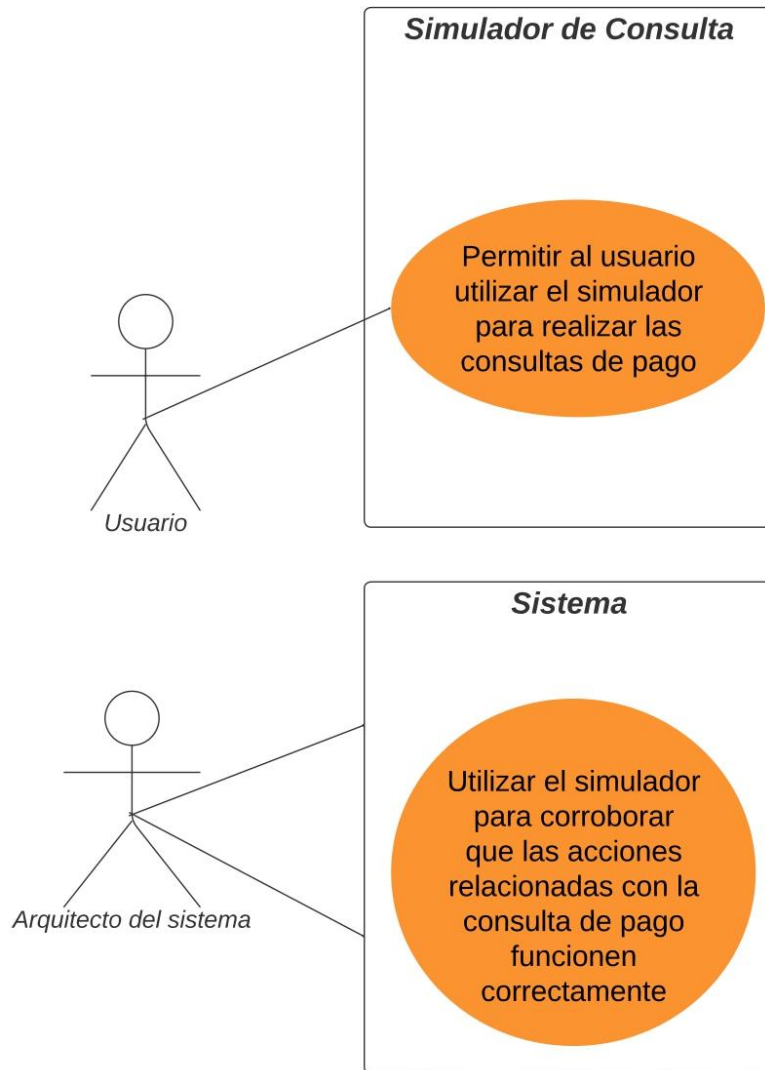
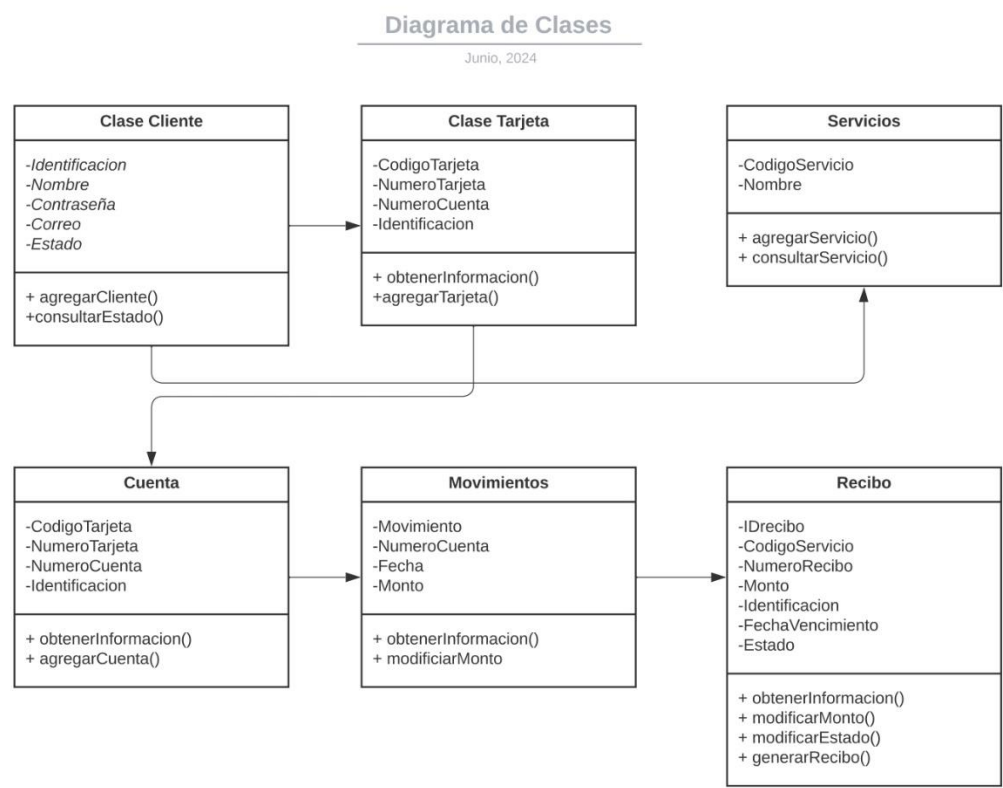
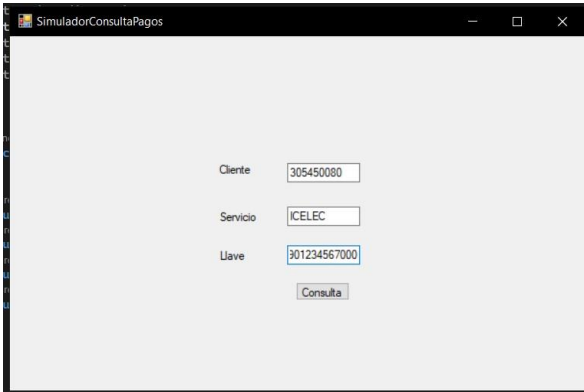
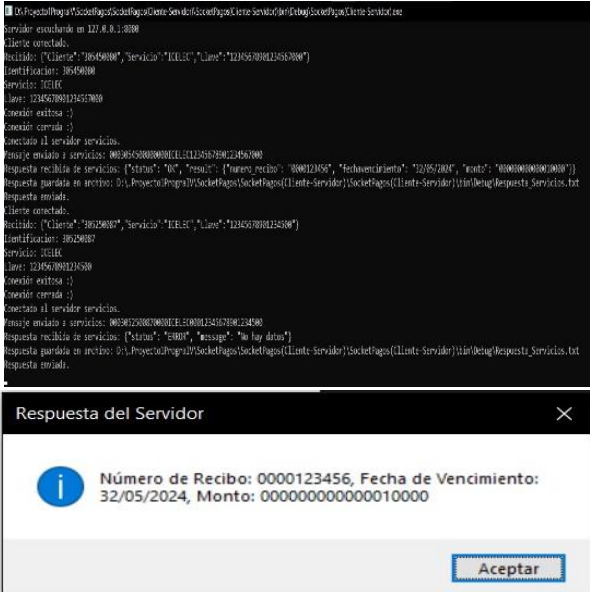


DIAGRAMA DE CLASES



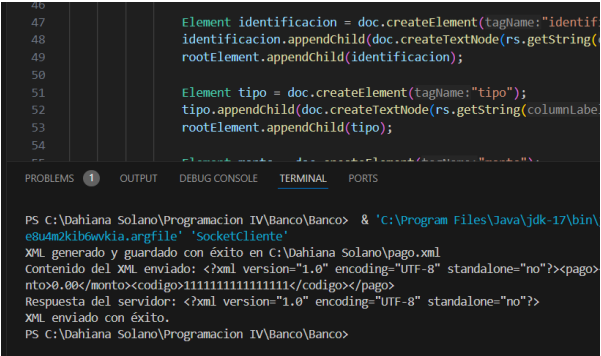
PRUEBAS TÉCNICAS

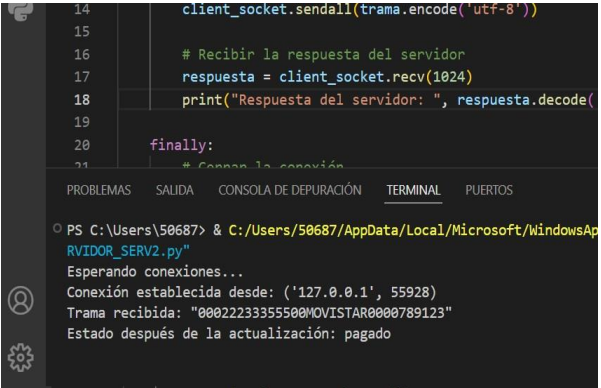
# HU	Criterio	Evidencia
SIM1	de aceptación	
	Un simulador de consulta y pago para corroborar que las acciones funcionan	

# HU	Criterio	Evidencia
PAGOS1	de aceptación	
	Un socket para permitir la consulta del monto a pagar de un servicio de un cliente.	

# HU	Criterio	Evidencia
SRV1	de aceptación	
	Un socket para permitir la consulta del monto a pagar de un servicio de un cliente.	<pre> 19 20 finally: 21 # Cerrar la conexión 22 client_socket.close() 23 </pre> <p>PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN <u>TERMINAL</u> PUERTOS</p> <p>PS C:\Users\50687> & C:/Users/50687/AppData/Local/Microsoft/WindowsApps/python3.12.exe RVIDOR.py"</p> <p>Esperando conexiones...</p> <p>Conexión establecida desde: ('127.0.0.1', 55131)</p> <p>Trama recibida: 00022233355500MOVISTAR10203040506070809555</p> <p>"OK[000078912310/07/20250000000000000000000000000000]"</p>

# HU	Criterio	Evidencia
PAGOS2	de aceptación	
	Una operación en el socket de pagos para permitir el pago de un recibo de un recibo.	<p>Servidor de servicios iniciado. Esperando solicitudes...</p> <p>Cliente conectado al servidor de servicios!</p> <p>Mensaje recibido del cliente: 0001011201330000ICELEC00000000000000000123465</p> <p>Respuesta del servidor de servicios: OK[00012345678901/01/202400000000000000000123456][000012345631/01/202400000000000000000721428][000012345631/01/202400000000000000000721426][000012345631/01/202400000000000000000721427][000012345631/01/202400000000000000000721429][000012345631/01/202400000000000000000721429]</p> <p>Monto a cancelar para el recibo 1234567890: 00123456</p> <p>Trama XML creada:</p> <pre> <pago> <identificacion>000101120133</identificacion> <tipo>CTA</tipo> <monto>00123456</monto> <codigo>45678912</codigo> </pago> </pre> <p>Trama XML enviada al Servidor Banco</p> <p>Respuesta del Servidor Banco: <ok/></p> <p>Trama para el servidor de servicios: 0001011201330000ICELEC1234567890</p> <p>Trama enviada al servidor de servicios</p>

# HU	Criterio	Evidencia
BANK1	de aceptación	
	Un socket para registrar el pago de un servicio de un servicio de un cliente.	 <pre> 46 47 48 49 50 51 52 53 54 55 Element identificacion = doc.createElement(tagName:"identif identificacion.appendChild(doc.createTextNode(rs.getString(rootElement.appendChild(identificacion); Element tipo = doc.createElement(tagName:"tipo"); tipo.appendChild(doc.createTextNode(rs.getString(columnLabe rootElement.appendChild(tipo); PROBLEMAS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS PS C:\Dahiana Solano\Programacion IV\Banco\Banco> & 'C:\Program Files\Java\jdk-17\bin\ e8b4a026b6w\kia.argfile' 'SocketCliente' XML generado y guardado con éxito en C:\Dahiana Solano\pago.xml Contenido del XML enviado: <?xml version="1.0" encoding="UTF-8" standalone="no"?><pago> nto>0.00</monto><codigo>1111111111111111</codigo></pago> Respuesta del servidor: <?xml version="1.0" encoding="UTF-8" standalone="no"?> XML enviado con éxito. PS C:\Dahiana Solano\Programacion IV\Banco\Banco> </pre>

# HU	Criterio	Evidencia
SRV2	de aceptación	
	Una operación de un socket para marcar un recibo un recibo como pagado.	 <pre> 14 15 16 17 18 19 20 21 client_socket.sendall(trama.encode('utf-8')) # Recibir la respuesta del servidor respuesta = client_socket.recv(1024) print("Respuesta del servidor: ", respuesta.decode(finally: # Cerrar la conexión PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL PUERTOS PS C:\Users\50687> & C:/Users/50687/AppData/Local/Microsoft/WindowsAp RVIDOR_SERV2.py" Esperando conexiones... Conexión establecida desde: ('127.0.0.1', 55928) Trama recibida: "00022233335500MOVISTAR0000789123" Estado después de la actualización: pagado </pre>

Conclusión

El desarrollo y diseño de un software de gestión de Pagos de servicios es diseñada para que los usuarios puedan realizar el pago de algún servicio de forma segura, ha sido proceso complicado para muchos desarrolladores a lo largo de la historia para llegar a crear mejores algoritmos que faciliten la calidad de vida de las personas, esto se ha convertido en una labor ardua, pero a su vez satisfactoria que brindará un mejor servicio para la empresa u organización que lo desarrolla y a su vez para el publico en general que utiliza la herramienta tecnológica, en este caso un sistema de información.

Actualmente muchas de las empresas, ya sean, grandes, medianas o pequeñas adquieren softwares para poder tener herramientas necesarias y agilizar sus labores cotidianas con respecto a la comercialización de su producto o servicio.

Recomendaciones

- Sugerir la incorporación en interfaz gráfica del servicio pagado y el monto del saldo restante en la cuenta del cliente.
- Recomendar la implementación de notificaciones o recordatorios periódicos para que los usuarios actualicen su información de contacto y siguiendo las mejores prácticas de seguridad cibernética.
- Proponer la inclusión de características adicionales, como la capacidad de agregar más cuentas del cliente y tener varias opciones para realizar el pago.

Bibliografía

Microsoft. (s/f). Socket services. .NET Fundamentals. Recuperado el 29 de mayo de 2024, de <https://learn.microsoft.com/es-es/dotnet/fundamentals/networking/sockets/socket-services>

Programarya. (s/f). Sockets. En Cursos Avanzados. Recuperado el 27 de mayo de 2024, de https://www.programarya.com/Cursos-Avanzados/Java/Socket#google_vignette

Python Software Foundation. (s/f). Howto de Python: Sockets. Recuperado el 26 de mayo de 2024, de <https://docs.python.org/es/3/howto/sockets.html>

Aprender BigData, "Introducción a MongoDB", aprenderbigdata.com, Recuperado el 23 de mayo. Disponible en: <https://aprenderbigdata.com/mongodb/>