
**Arquitectura de software - Facultad de
Ingeniería Universidad de
Antioquia**

Código: 2508585
Versión: 4
Fecha: Agosto 14

Tiempo: 4 horas

Examen parcial

Nombre: Sebastian Londoño Tobon

código: 1035833769

Caso práctico.

Nubank, el banco digital más grande del mundo con operaciones en Brasil, México y ahora en Colombia con 30 millones de clientes, acaba de comprar un Banco en la ciudad de Sao Paulo denominado **Chubb** para ofrecer servicios de banca completamente digital. Actualmente Chubb cuenta **con 10 millones** de clientes. La meta de Nubank es incorporar nuevos clientes y **llegar a 50.000.000** de clientes en los próximos 5 años.

Restricciones

Por políticas de la gerencia, los clientes de NuBank deben quedar registrados en el CRM para dirigir más fácilmente las campañas de mercadeo sobre el CRM Dynamics 365. Uno de los grandes retos de arquitectura consiste en llevar los clientes **de Chubb** a los sistemas de NuBank, garantizando consistencia de la información entre el **CRM Dynamics 365** y el sistema de clientes del banco **Chubb**.

Por restricciones legales se debe garantizar la privacidad en los datos de los clientes, asegurando que la información de los clientes esté protegida mediante SSL.

NuBank está trabajando en un sistema de servicios y bancarización por internet, el cual permitirá a los clientes evaluar entre varios productos de préstamo digital y que puedan realizar el pago en línea de aquel que más se ajuste a sus necesidades.. El sistema debe permitir la venta de productos en todos los países en los cuales tiene presencia NuBank.

Como consideración especial el sistema valida previo a la venta del producto digital de préstamo los siguiente datos:

**Arquitectura de software - Facultad de
Ingeniería Universidad de
Antioquia**

Código: 2508585
Versión: 4
Fecha: Agosto 14

- Valida que el cliente no se encuentre requerido por INTERPOL mediante microservicio de la agencia de investigación INTERPOLws
- Valida que el cliente no se encuentre reportado a centrales de riesgo mediante un microservicio de DATACREDITO denominado DATACREDITOws
- Con el fin de mitigar el riesgo de tener clientes con dudosa reputación se valida que no se presente lavado de activos ni apoyo al terrorismo mediante un servicio denominado SARLAFTws.

Usted como arquitecto deberá:

- Definir un esquema que facilite el gobierno de todas las integraciones que se realizarán
- Soportar el volumen de crecimiento en clientes sin degradar su desempeño
- Especificar como la venta de productos por internet le permitirá de forma flexible cambiar las validaciones y valores sin afectar la disponibilidad de la sucursal virtual.

Entregables:

1. Los drivers arquitectónicos (Requisitos funcionales, requisitos no funcionales y restricciones técnicas (20%))

R/:

Requisitos funcionales

- El sistema deberá permitir registrar a los clientes de NuBank en el CRM.
- El sistema solo debe permitir el acceso a la información de los clientes a usuarios autorizados.
- El sistema debe garantizar la privacidad de los datos de los clientes.
- El sistema debe permitir a los clientes evaluar entre varios productos de préstamo digital.
- El sistema debe permitir que el cliente realice el pago en línea del producto elegido.
- El sistema debe permitir la venta de productos en los países donde está NuBank.
- El sistema debe permitir validar datos sensibles como reportes en centrales de riesgo, Interpol u actos de lavado de activos.
- El sistema deberá soportar el crecimiento en el volumen de clientes sin disminuir su desempeño.
-

Requisitos no funcionales

- El sistema deberá proteger la información de los clientes mediante SSL.
- El sistema deberá validar requerimientos de la interpol antes de una venta mediante el microservicio INTERPOLws.
- El sistema deberá validar reportes en centrales de riesgo antes de una venta mediante el

microservicio DATACREDITOWs.

- El sistema deberá validar la reputacion, lavado de activos o apoyo al terrorismo antes de una venta mediante el microservicio SARLAFTws.
- El sistema deberá de contar con manuales de usuario ordenados y estructurados.
- El tiempo de aprendizaje del sistema deberá de ser menor a 6 horas.
- El sistema deberá de contar con un apartado para ayuda en línea.
- El sistema deberá tener un diseño Responsive que permita ser desplegado en dispositivos móviles, tablets o computadoras.
- El sistema debe ser capaz de soportar transacciones simultáneas de hasta 80000 clientes concurrentes.
- El sistema deberá permitir a los usuarios administradores ver la información de los clientes pero no modificarla.

Restricciones técnicas

- El sistema debe usar SSL para cifrar los datos entre el navegador y el sitio web.

-
-
-
-

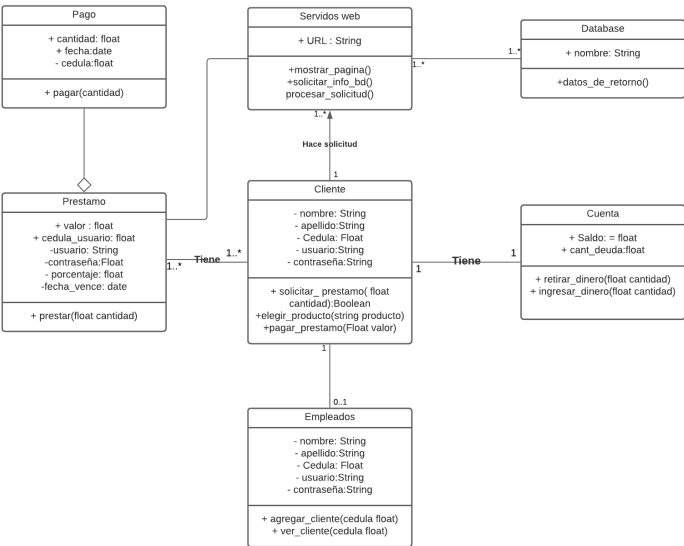
2. Defina cuales son las entidades (clases) arquitectónicamente más significativas (10%)

R/:

- Préstamo
- Cliente

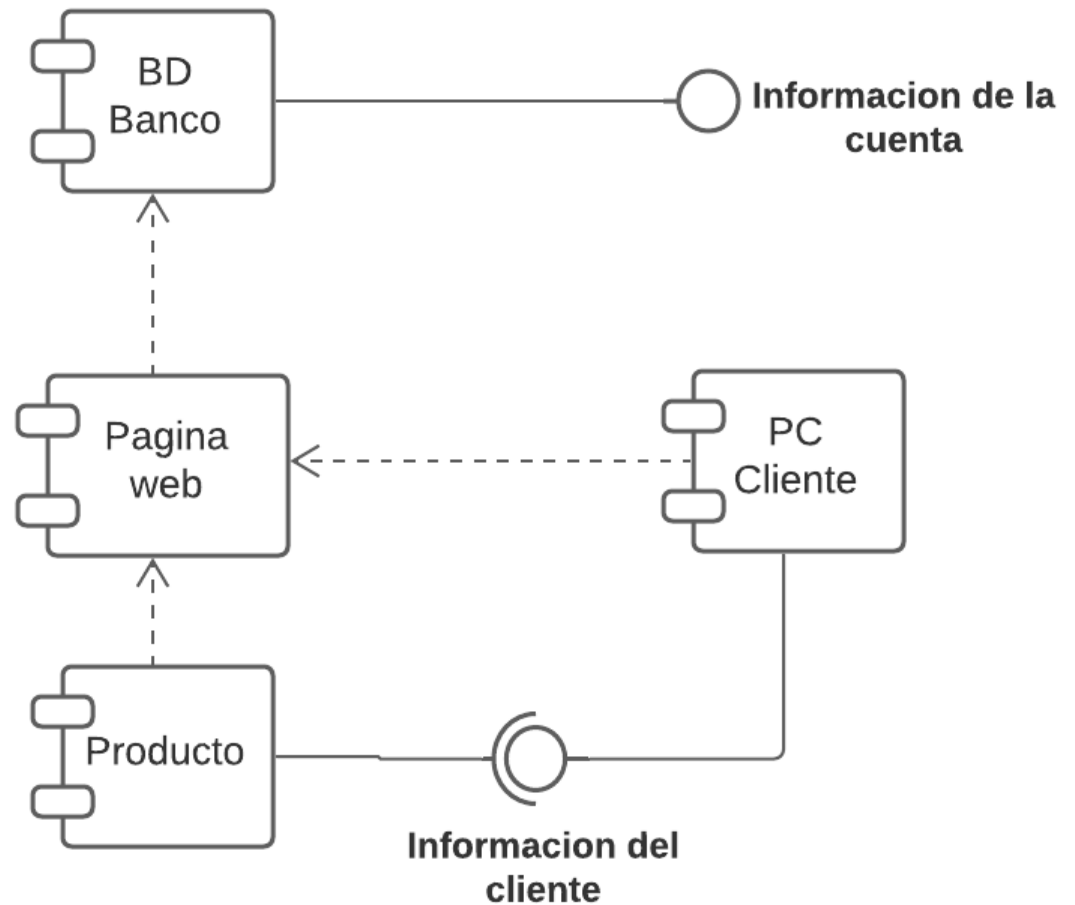
3. Diagrama de 4+1

a. Diagrama de clases (10%)



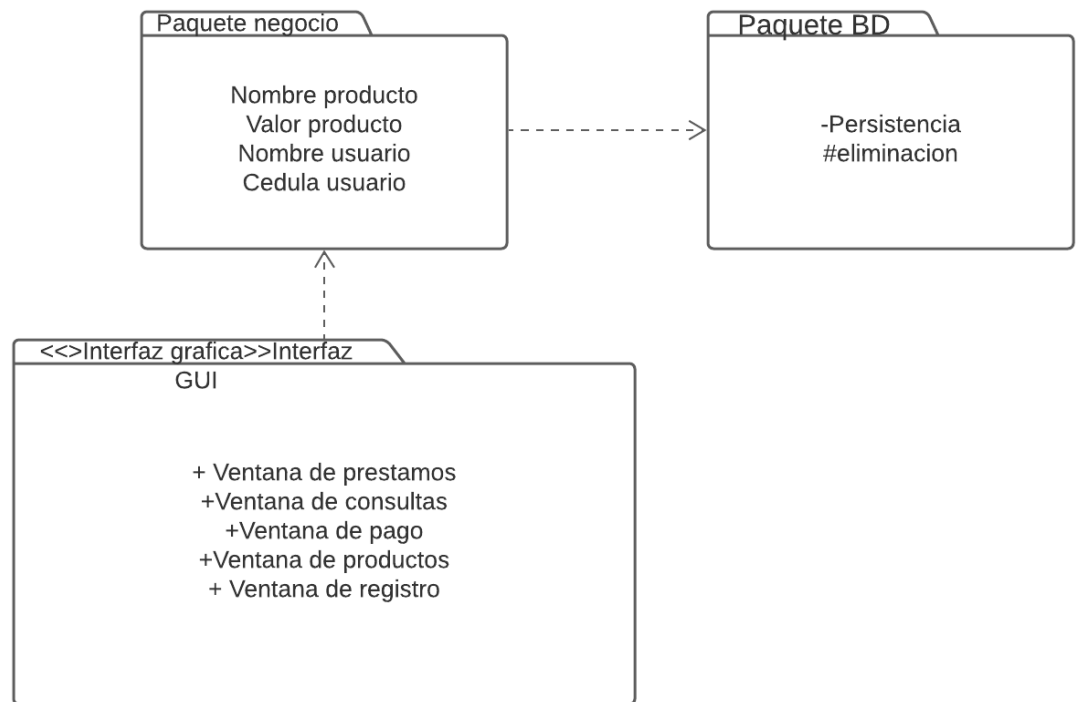
Enlace: https://lucid.app/lucidchart/invitations/accept/inv_c58ba450-7f98-409a-95b4-ee25541f0025?viewport_loc=-2383%2C-1391%2C3330%2C1461%2C0_0

b. Diagrama de componentes (20%)



Enlace: https://lucid.app/lucidchart/invitations/accept/inv_c58ba450-7f98-409a-95b4-ee25541f0025?viewport_loc=-2383%2C-1391%2C3330%2C1461%2C0_0

c. Diagrama de paquetes (10%)



Enlace:

https://lucid.app/lucidchart/invitations/accept/inv_c58ba450-7f98-409a-95b4-ee25541f0025?viewport_loc=-2383%2C-1391%2C3330%2C1461%2C0_0

d. Diagrama de despliegue (10%)



1. Cuáles serían las entradas para la definición de una arquitectura de un proyecto software (10%)

2. Cuáles son los pasos lógicos para definir una arquitectura de software (5%)

- Escribir los requisitos y restricciones del sistema.
- Elegir las tecnologías, protocolos y lenguajes que se van a utilizar.
- Diseñar los diagramas necesarios basándose en los componentes y tecnologías elegidos anteriormente.
- Probar la arquitectura creada con casos de la vida real y con el problema que se trató de resolver.
- Implementar la arquitectura de ser exitosas las pruebas, en otro caso volver al ítem A y empezar de nuevo.

3. Defina la diferencia entre un patrón de diseño y un patrón arquitectónico (5%)

R/: El patrón de diseño ayuda a estructurar u organizar nuestro código en términos de las relaciones que tienen los elementos que lo componen como las clases etc, y el patrón arquitectónico se encarga no de esas relaciones a nivel de código sino más bien a un nivel más general y sus posibles formas de relacionarse e interactuar.