



Universidad de Antioquia

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

Código: 2508585

Versión: 4

Fecha: Agosto 14

Tiempo: 4 horas

Examen parcial

Nombre: \_\_\_\_\_ código: \_\_\_\_\_

## 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 los 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.

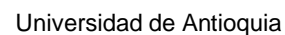
*Usted como arquitecto deberá:*

- 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%)
2. Defina cuales son las entidades (clases) arquitectónicamente más significativas (10%)
3. Diagrama de 4+1
  - a. Diagrama de clases (10%)
  - b. Diagrama de componentes (20%)
  - c. Diagrama de paquetes (10%)
  - d. Diagrama de despliegue (10%)





Código:	2508585
---------	---------

Edici3n:	1
Versi3n:	4

Fecha: Agosto 14

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

Solución:

1.

Requisitos funcionales:

- La aplicación móvil y sitio web deberán tener un módulo de inicio de sesión para que el cliente se identifique debidamente y pueda ver sus productos disponibles, saldrá el mensaje “Usuario y/o contraseña incorrectos” si el usuario no se identifica exitosamente
- La aplicación móvil y sitio web deberán tener un módulo de ventas en el que el usuario solo se autenticará al finalizar la solicitud, es decir, el usuario ingresa al sitio o aplicación y deberá ir al módulo de “Solicitud de servicios”, donde le aparecen los productos disponibles, una vez allí, el cliente debe diligenciar todos los datos y al finalizar, debe presionar el botón “Solicitar”, allí se le pedirán sus datos de contacto para que un asesor se contacte con él por diferentes medios.
- El módulo de ventas deberá garantizar lo siguiente:
  - 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.
- La aplicación debe mostrar en pantalla el resultado de la solicitud o el estado en que se encuentra

Requisitos no funcionales

- Como cliente deseo que el sistema de información tenga los logos de la compañía en los sitios web que se desarrollen y que el color se ajuste a los parámetros de comunicaciones de la compañía
- Yo como cliente deseo que el sistema sea multiplataforma ya que nuestra compañía tiene presencia en varios países

- Yo como cliente deseo que los datos disponibles en los dos CRM de las empresas estén disponibles en un único módulo web, el cual, me ayude en la representación y análisis del comportamiento de mis clientes.
- Yo como cliente deseo el desarrollo de un módulo web de ventas, el cual ofrece a los clientes de todas nuestras sedes los productos de préstamo digital, una vez tomado el producto.
- El sistema debe permitir la venta de productos en todos los países en los cuales tiene presencia NuBank.
- Se debe conocer cuál es el medio de ingreso de la solicitud, es decir, si es por browser o dispositivo móvil
- La aplicación soportará un apartado de simulación para saber si el producto solicitado si puede ser debidamente desembolsado

#### Restricciones técnicas

- Los clientes estarán debidamente registrados en el CRM Dynamics 365
- La información de los clientes estará debidamente cifrada mediante SSL
- El lenguaje usado para las consultas en base de datos será SQL
- Los enlaces de una capa a otra serán por medio del protocolo Https
- El enlace específico para el módulo de fraude a los diferentes medios de reporte se hará por medio de Https
- Todo el Frontend de la aplicación será desarrollado usando JavaScript

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

Cliente: Define cada uno de los atributos que posee la persona que hace uso de los servicios que ofrece el banco, como son su id, nombre, residencia, país, etc

Productos No Cuenta: Define cada uno de los productos que se ofrecen desde el banco y que los clientes toman desde el módulo web, el cual pueden pagar de forma on line, son diferentes de la cuenta bancaria

Cuenta Bancaria: Define el tipo y número de cuenta que poseerá la persona que hace uso de los

servicios del banco, no necesariamente se deberá tener cuenta para hacer uso de los productos que se ofrecen desde el banco.

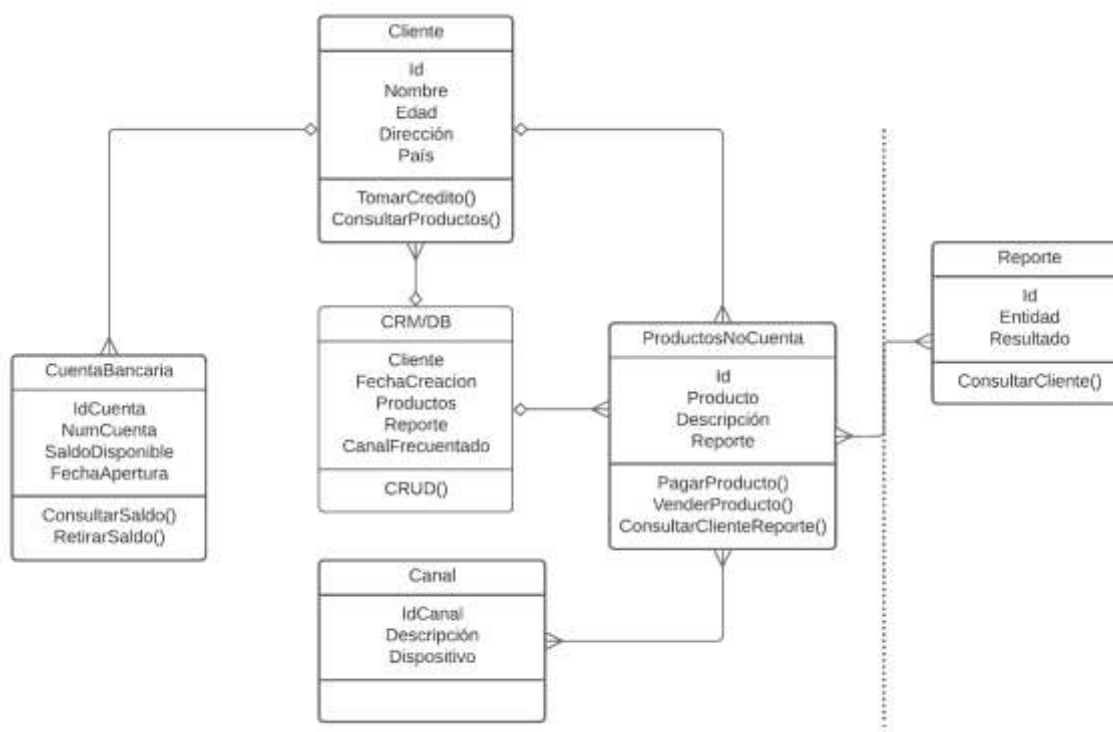
Reporte: Esta clase define si una persona está debidamente reportada en las centrales de riesgo como son Interpol, Datacrédito o Sarlaft

Canal: Define el medio (Dispositivo) en el que el cliente realiza la consulta y toma de productos, es decir si es desde el canal de aplicación para smartphone o web.

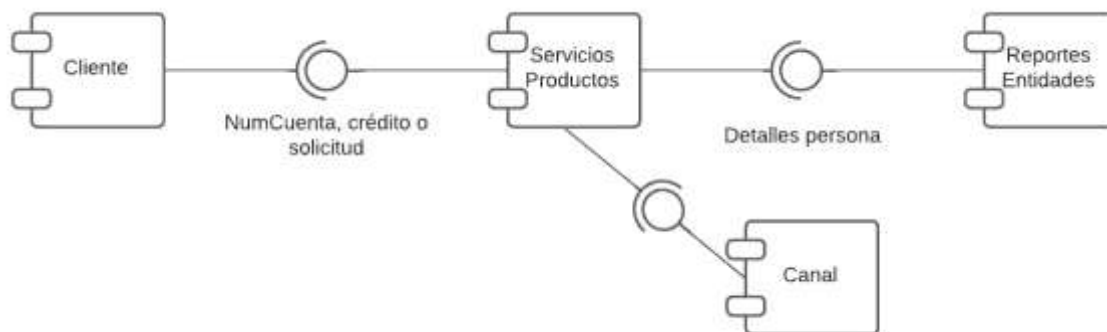
CRM/DB: Establece la base de clientes en el sistema, aquí están registradas todas las personas que tienen relación con el banco.

### 3. Diagrama de 4+1

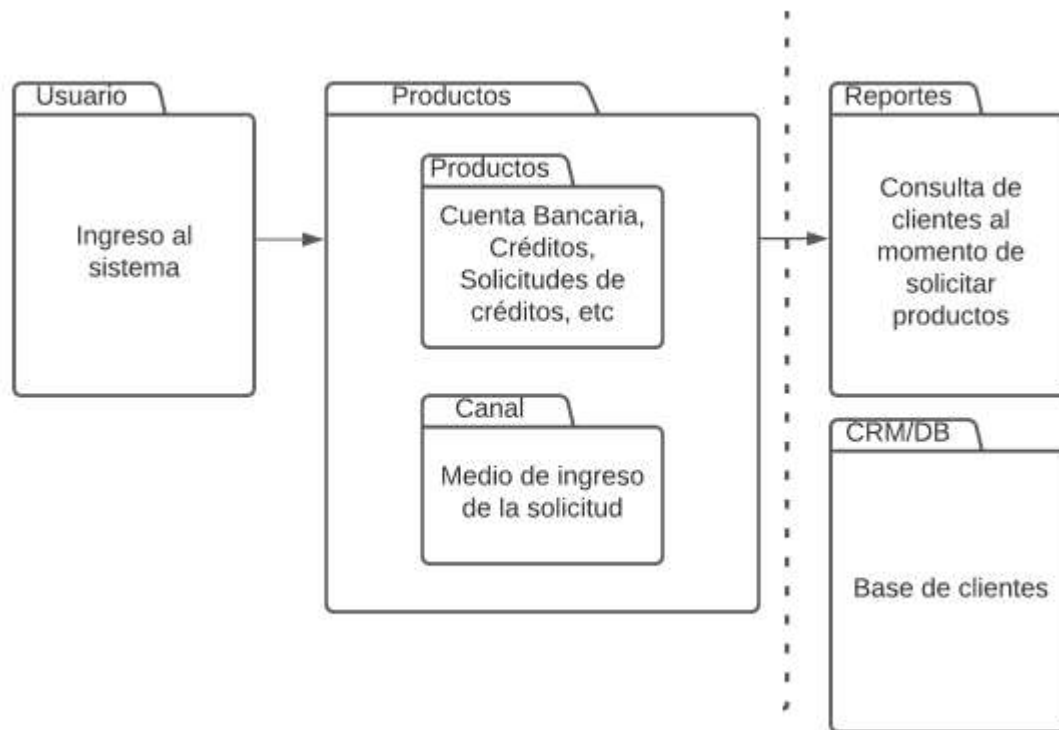
#### 3.1. Diagrama de clases:



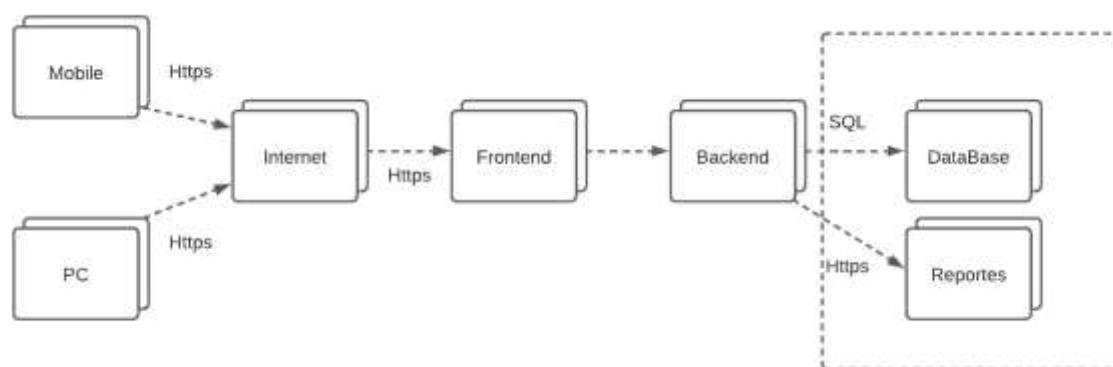
#### 3.2. Diagrama de componentes:



### 3.3. Diagrama de paquetes:



### 3.4. Diagrama de despliegue:



#### 4. Cuáles serían las entradas para la definición de una arquitectura de un proyecto software

Lo que entiendo por las entradas para la definición de una arquitectura de un proyecto de software es que son los componentes previos que se deben tener para el desarrollo de la arquitectura para el proyecto, es decir, lo que debemos tener previamente al momento de iniciar con el proyecto, como, por ejemplo, descripción del negocio, requisitos funcionales y no funcionales, requisitos de escalabilidad a futuro, etc. Requisitos que llevan a la comprensión del problema para su posterior desarrollo empleando las vistas y estructuras.

#### 5. Cuáles son los pasos lógicos para definir una arquitectura de software

Para la apropiada definición de la arquitectura para nuestro proyecto, debemos comprender el entorno en el que está dicho proyecto, es decir, lo primero que debemos comprender es el Negocio, luego de esto, comprender lo que realmente necesita y no necesita el cliente, lo que conocemos como Requisitos (F y NF), luego, la selección de la arquitectura a seguir, es decir, la estructura que tendrá nuestra aplicación, desarrollando cada una de las vistas (Escenarios), las cuales documentamos para que las demás personas involucradas en el proyecto comprendan, interpreten y desarrollen en el equipo de trabajo, siguiendo una metodología de las muchas que existen.

Evidentemente debemos hacer la evaluación a conveniencia que tiene la arquitectura seleccionada para seguir posteriormente con la materialización de la arquitectura con sus drivers arquitectónicos e implementarlos, seguido de esto, debemos definir las responsabilidades de



todos los módulos definidos en el proyecto e implementarlos con protocolos, flujo de datos, despliegue, etc.

En resumen, las entradas para la definición de la arquitectura de un proyecto de software son: Negocio (Comprender el negocio), Requisitos (Funcionales y NO funcionales), Arquitectura (Definición de estructura y vistas o escenarios), Evaluación de la arquitectura, implementación.

#### 6. Defina la diferencia entre un patrón de diseño y un patrón arquitectónico

Se entiende que un patrón arquitectónico está enfocado en los detalles grandes del proyecto, es decir, a nivel general se centra en el contexto del problema, se eligen las tecnologías necesarias para la solución del problema y los requisitos no funcionales, por otro lado, el diseño se centra en los componentes particulares de la solución del problema, es decir, cada uno de los requisitos funcionales.

Mientras que en un patrón de arquitectura se centra en los frameworks que se deben emplear para solucionar el problema de la aplicación, en el patrón de diseño se centran en el uso paso a paso para la solución del problema usando ese framework.

Se entiende entonces la analogía: Arquitectura a general como diseño a lo particular.