



Expertos enseñando a Expertos

Curso Java SE8 Programming

Caso Práctico: Banca Electrónica



101101110
101101110100011
10110111010
1011011101000

Caso Práctico: Banca Electrónica

1. Introducción
2. Definición
3. Análisis de Requerimientos
4. Restricciones
5. Casos de Uso
6. Diagrama de Clases
7. Implementación

1.Introducción al Caso Práctico: Banca Electrónica

Escenario

Su compañía esta planeando cambiarse a la plataforma de la versión Java SE 8, y debido a que su empresa tiene miles de desarrolladores que utilizan diferentes lenguajes, desea realizar un prototipo de una aplicación que sirva como guía para el resto de los desarrollos y migraciones, para ello se ha elegido el realizar una aplicación simplificada de “BANCA ELECTRONICA”.

2. Definición de la Aplicación de Banca Electrónica

Se desea realizar una aplicación bancaria que permita el acceso a sus servicios de forma online y en horario 7 x 24 que acerque la sucursal bancaria a los clientes ofreciendo todo tipo de servicios, en la primera versión se iniciará una aplicación standalone con los servicios básicos, como son el manejo de diversos tipos de cuentas y sus operaciones.

3. Análisis de Requerimientos

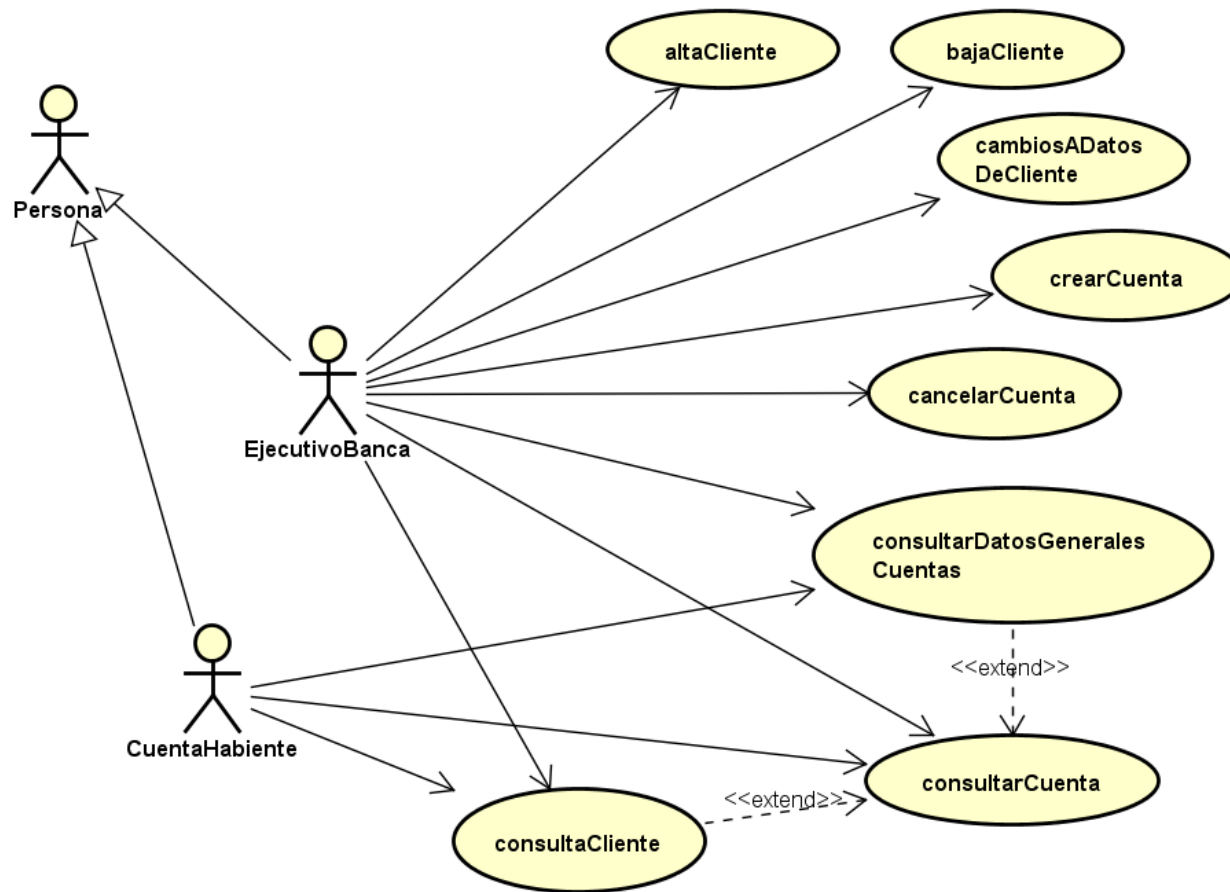
Los requerimientos iniciales de orden general son:

- Ofrecer facilidades a los ejecutivos del banco para el manejo de sus clientes (altas, bajas, cambios y consultas)
- Los ejecutivos también podrán realizar el manejo de las cuentas tales como apertura, cancelación, consultas e impresión de cuentas.
- Los clientes deberán de poder realizar consultas a sus cuentas y realizar las operaciones básicas.

4. Restricciones

- Para empezar a disfrutar de este servicio, será necesario acudir físicamente a una oficina del banco para que un ejecutivo nos dé el alta como clientes y se firmen los contratos oportunos.
- Existen algunas operaciones que no debe poder realizar el cliente si no es a través de su ejecutivo de cuenta, principalmente apertura y cancelación de cuentas, las operaciones de alta y baja de cliente.
- Las cuentas de ahorros tendrán un rendimiento mensual del 2% abonados al final de cada mes.

5. Casos de Uso



powered by Astah

6. Diagrama de Clases

- El Diagrama de Clases es el diagrama principal para el análisis y diseño.
- Un diagrama de clases presenta las clases de objetos del sistema con sus relaciones estructurales de composición y de herencia.
- La definición de una clase incluye especificar los atributos y operaciones.
- El modelo de requerimientos aporta información para establecer las clases, objetos, atributos y operaciones.

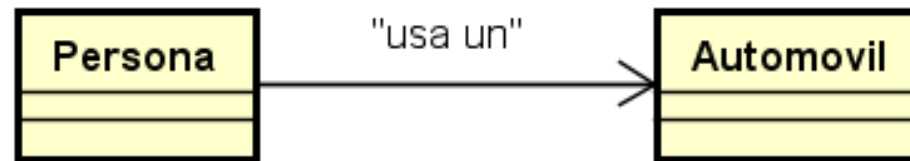
Relaciones en el Diagrama de Clases

Tipo: Relación simple de uso

Frase asociada: “usa un”

Simbología: una flecha clara

Ejemplo:



powered by Astah

Relaciones en el Diagrama de Clases

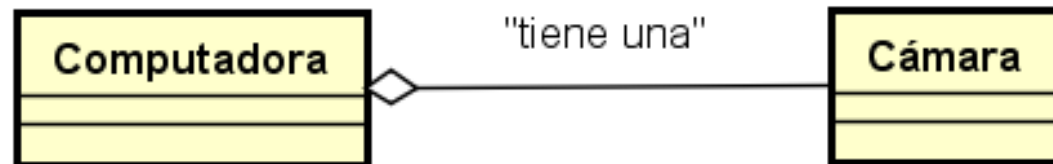
Tipo: Relación de Agregación

Comentario: La agregación es opcional

Frase asociada: “tiene un”

Simbología: una línea con un rombo vacío del lado de quien tiene a los objetos relacionados

Ejemplo:



powered by Astah

Relaciones en el Diagrama de Clases

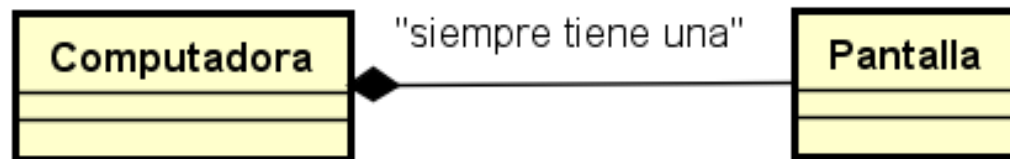
Tipo: Relación de Composición

Comentario: La composición es obligatoria

Frase asociada: “siempre tiene un”

Simbología: una línea con un rombo lleno del lado del compuesto

Ejemplo:



powered by Astah

Relaciones en el Diagrama de Clases

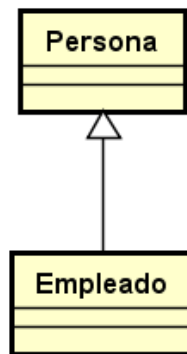
Tipo: Relación de Herencia

Comentario: La herencia crea alta dependencia entre clases padre y clases hijas (superclases y subclasses)

Frase asociada: “es un”

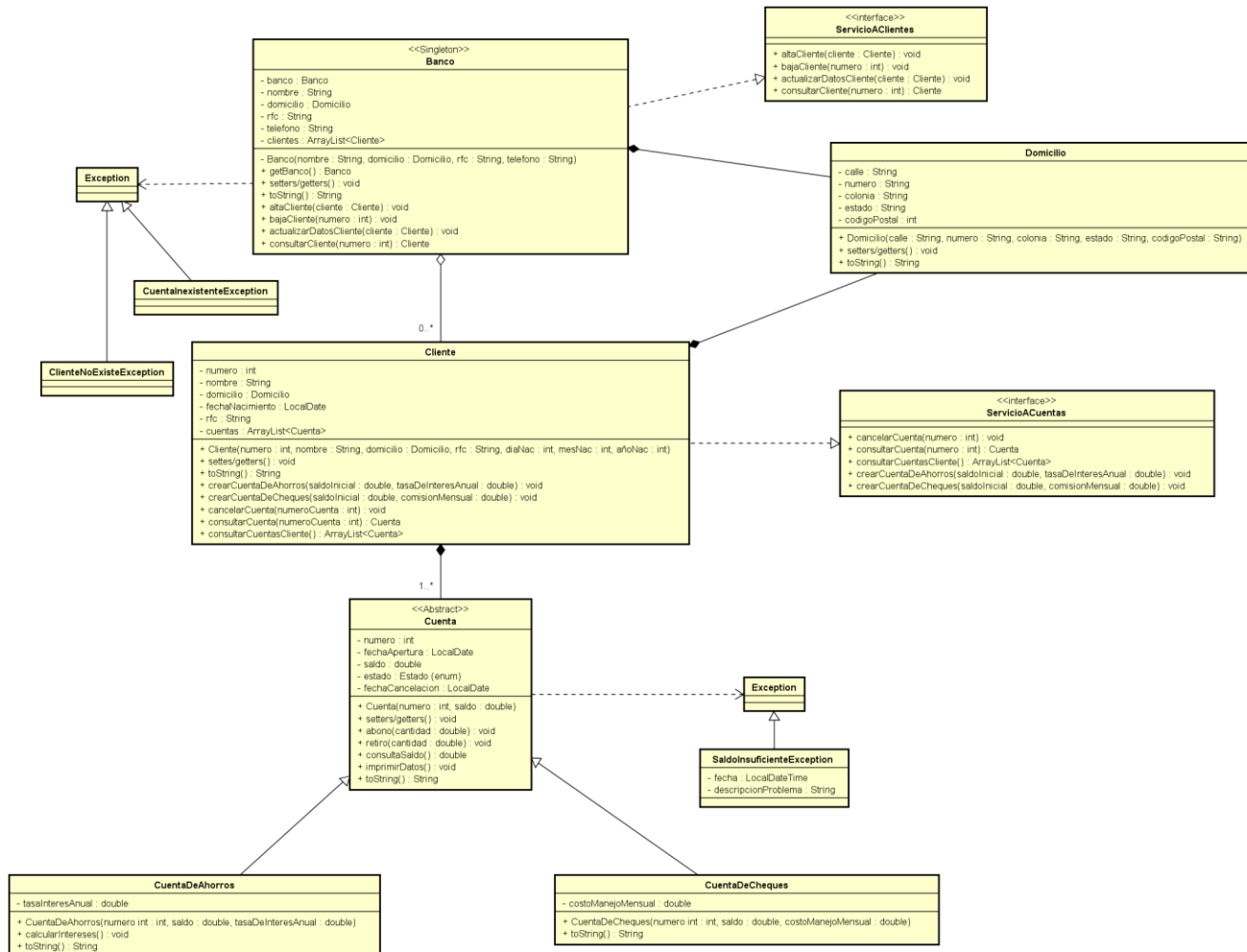
Simbología: una línea con un triángulo apuntando a la clase padre y en el otro extremo la clase hija

Ejemplo:



powered by Astah

Diagrama de Clases en el Diseño



powered by Astah

7. Implementación Básica

- Interpretar los diagramas UML presentados y desarrollar el modelo del negocio en lenguaje Java usando los diagramas de casos de uso y clases proporcionados, realizando las pruebas necesarias para el modelo del negocio en formato texto por medio de la consola.

Implementación Opcional

- Adicionalmente desarrollar el acceso a la base de datos usando el patrón DAO (Data Access Object) y crear las siguientes clases DAOBanco, DAOClientes y DAOCuentas.
- Hacer una Interface Gráfica de Usuario que permita cumplir con los requerimientos funcionales y los casos de uso preestablecidos usando Swing en Netbeans.

Gracias por tu atención

Netec

Expertos enseñando a Expertos



- **Ciudad de México**

Piso 9 Torre WTC
Montecito 38
Col. Nápoles, Cd. De
México
Tel. (55) 9000 4500

- **Monterrey**

Av. Morones Prieto No.1500 Pte.
Convex, Planta baja, Suite 104,
Col. Nuevas Colonias,
Monterrey, N.L. C.P. 64710
Tel. 01 (81) 1534 5656

- **Guadalajara**

Av. López Mateos Sur No. 2077
Int. Z9B Col. Jardines de Plaza
del Sol Guadalajara, Jal.
C.P. 45510 (Centro del Software)
Tel: (33) 3030 7230

Síguenos

