**Integrantes:** Nathaly Cumbicos

David Velasco

**Curso:** GR2

**Fecha:** 20/10/2016

**PROYECTO SERVIDOR DE TRANSACCIONES BANCARIAS**

1. **Alcance**

La arquitectura a implementarse será la descrita en la figura 2. Se implementará un Servidor de transacciones bancarias instalado en un computador de escritorio, denominado Servidor Banco, donde se podrá realizar operaciones de Retiro, Depósito y Consultas asociados a los clientes del banco. Los datos provenientes de estas operaciones se almacenan en un Servidor de base de datos, el cual estará instalado conjuntamente en el mismo Servidor Banco. Cada servidor de transacciones bancarias cuenta con una sonda, la cual recopilará los datos de procesamiento (CPU, Memoria) en un archivo de texto en tiempo real y serán enviados al balanceador cuando los solicite.

Se implementará también un balanceador de carga, el que será el encargado de recibir las peticiones de los clientes, luego enviará una petición a todos los servidores de banco solicitando su información de carga, de acuerdo a estos datos, se determinará el servidor a utilizarse, y se enviará la solicitud proveniente del cliente.

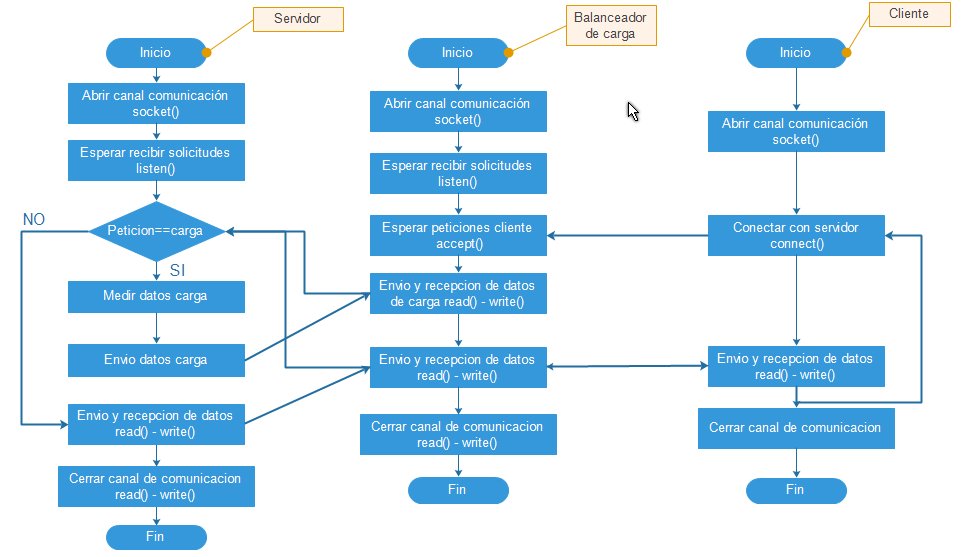
Cada cliente está provisto de una GUI, donde podrá loguearse con sus datos (cedula, contraseña) desde la cual podrá realizar las transacciones antes mencionadas.

Cada petición de cliente a enviarse, se guardará en un solo String y tendrá el formato:

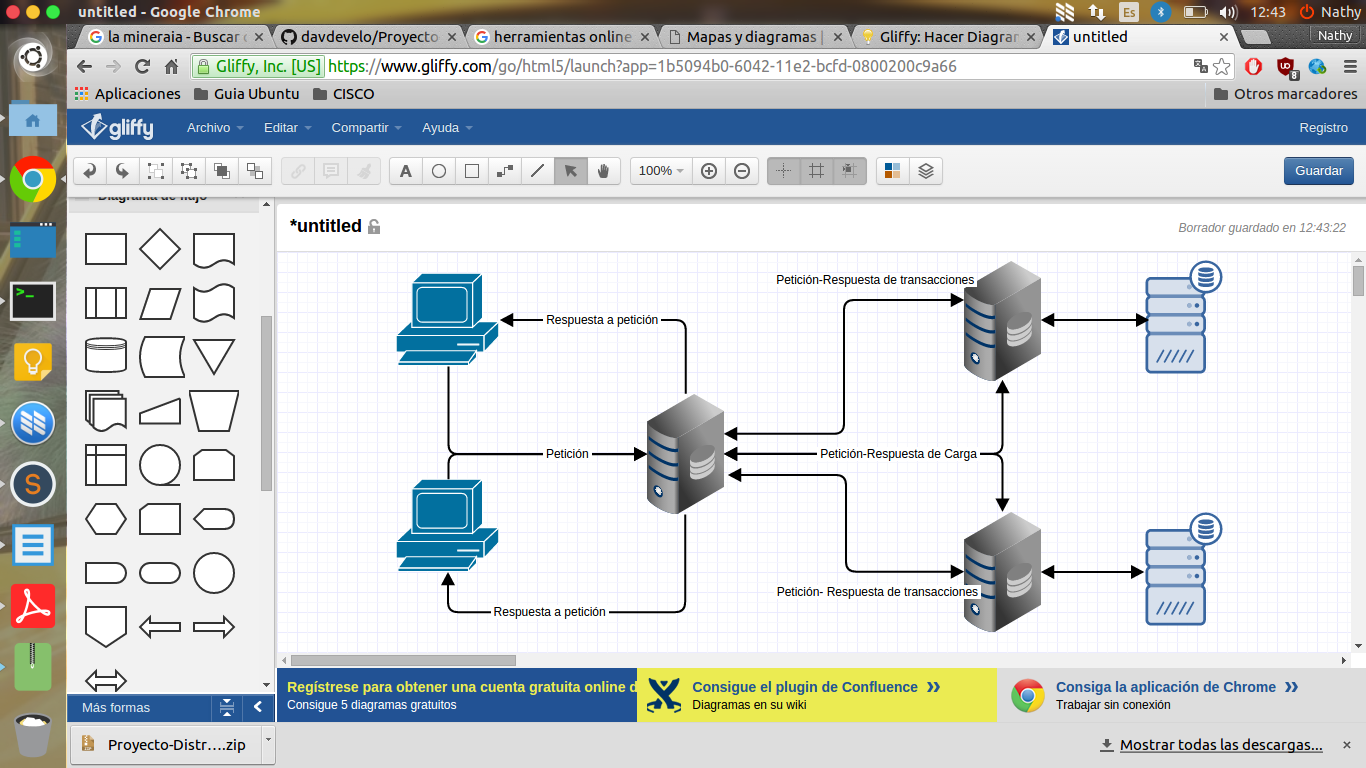
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Cédula** | **Contraseña** | **Identificador** | **Tipo de transacción** | **Valor** | **Identificador** |
| 10 caracteres | Tamaño variable | **#** | 0 ≈ Consulta  1 ≈ Depósito  2 ≈ Retiro | Tamaño variable | **#** |

Para estresar a los servidores bancarios, desde consola se enviarán múltiples peticiones generadas aleatoriamente.

1. **Diagrama de Flujo de arquitectura de alta disponibilidad**

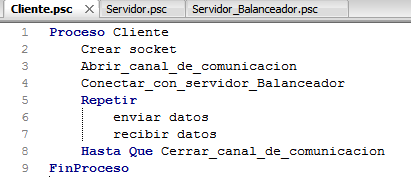


1. **Diagrama de bloques**

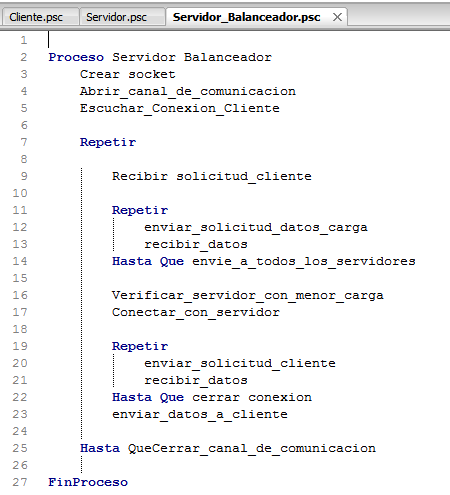


1. **Seudocódigo**

**Cliente**



**Servidor: Balanceador de carga**



**Servidor**

