

**AUTOMASI TELER PADA KOPERASI SIMPAN PINJAM DENGAN
METODE *REMOTE PROCEDURE CALL* PADA SISTEM TERDISTRIBUSI**
*AUTOMATION TELER ON COOPERATIVE FINANCE WITH METHOD OF REMOTE
PROCEDURE CALL IN DISTRIBUTED SYSTEM.*

Dosen Pengampu Mata Kuliah Sistem Paralel dan Terdistribusi :

Muhammad Al Makky, S.Kom,.M.T.

Disusun untuk memenuhi Tugas Besar Mata Kuliah Sistem Paralel dan Terdistribusi

Disusun Oleh:

- | | | |
|---|-----------------------|------------|
| 1 | Sageri Fikri Ramadhan | 1301164341 |
| 2 | Yulistia Elsa Nursida | 1301164229 |
| 3 | Riyan Kuncoro Jati | 1301164021 |
| 4 | Aji Priambodo Santoso | 1301164096 |



**PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
UNIVERSITAS TELKOM
2019**

- Peran Anggota

Nama	Peran
Aji Priambodo Santoso	Perancangan fungsionalitas
Riyan Kuncoro Jati	Perancangan fungsionalitas
Sageri Fikri Ramadhan	Perancangan unit pengujian, Basis data
Yulistia Elsa Nursida	Perancangan Host, Port Jaringan, dan Dokumentasi

1. Gambaran Umum Sistem

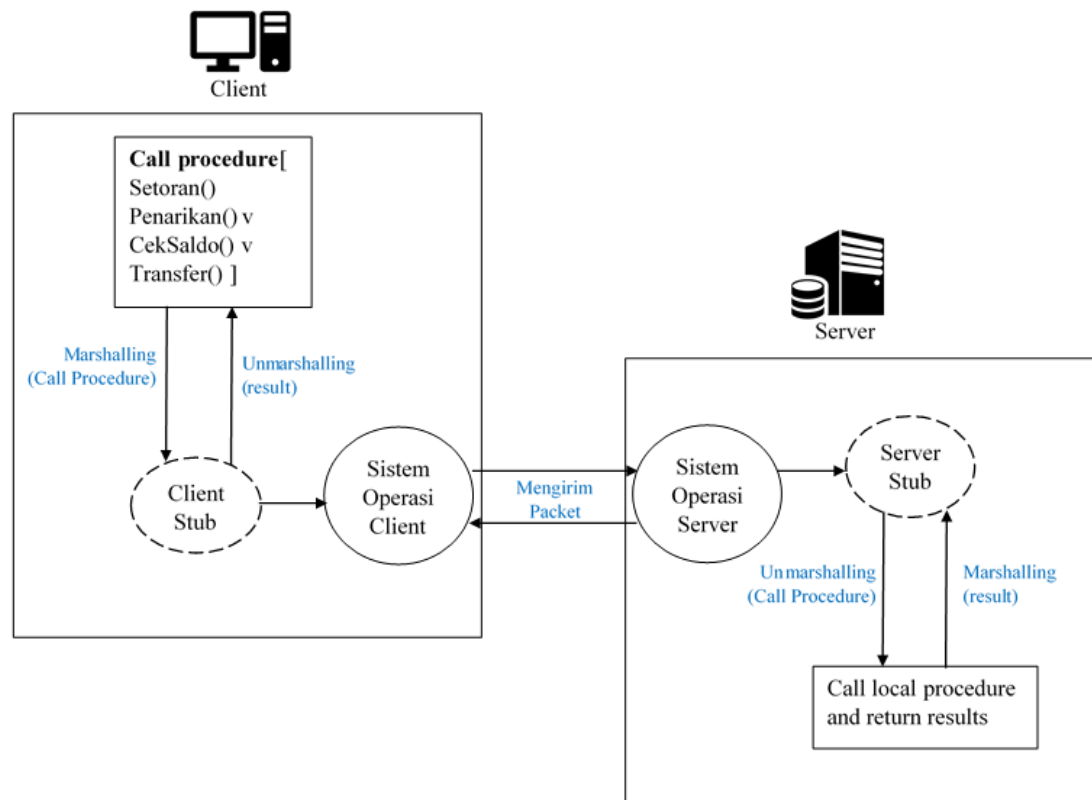
ATM (Automated Teller Machine) merupakan sebuah mesin elektronik yang mempermudah nasabah dalam mengambil, menyetor, mengecek saldo yang ada pada rekening tabungan, melakukan transfer atau transaksi perbankan lainnya tanpa harus ke bank dan menggunakan pelayanan dari teller. Seperti halnya pada sistem yang akan kami buat yaitu sistem untuk mesin ATM. Jadi dalam sistem ini ATM menggunakan arsitektur *client-server* dimana *client* merupakan nasabah dan *server* merupakan database pusat yang menyimpan data nasabah. Data nasabah yang disimpan di *server* yang berupa nomor rekening dan saldo yang dimilikinya. Kemudian *client* bisa langsung mengakses sistem dengan memilih menu apa yang akan dijalankan. menu yang kami sediakan adalah melakukan setor tunai, melihat saldo, dan melakukan *transfer*. Setelah memilih fitur mana yang akan dilakukan *client* cukup memasukkan nomor rekeningnya maka *server* akan menampilkan hasilnya, untuk transfer *client* harus memasukkan nomor rekeningnya, nomor rekening tujuan dan nominal yang akan ditransfer.

2. Rancangan Sistem

a. Environment Development System

Pada sistem ini *client* akan melakukan *call procedure* yang disediakan oleh *server*. Sebelum dikirimkan ke *server*, *call procedure* ini akan di-*marshalling* terlebih dahulu oleh *client stub* lalu dengan bantuan sistem operasi *call procedure* yang sudah di-*marshalling* dikirim dalam bentuk paket melalui jaringan. Kemudian akan diterima di komputer *server*, dengan bantuan sistem operasi paket akan

diberikan kepada *server stub*. Lalu dilakukan unmarshalling untuk membuka pesan yang dikirimkan oleh *client*. Setelah itu, *server* akan memanggil *procedure* yang diminta oleh *client* lalu mengembalikan hasil pemanggilan tersebut kepada *server stub*. Dari *server stub* akan dilakukan proses *marshalling* lalu hasil akan dikirimkan dalam bentuk paket dari sistem operasi *server* ke *client*. Sistem operasi *client* akan menerima paket dan memberikan ke *client stub* untuk di-unmarshalling dan diterima hasilnya. Untuk lebih jelasnya bisa dilihat pada gambar berikut.



Gambar 1. Deployment Diagram

b. Architectural System

Sistem ini dirancang pada perangkat lunak, perangkat keras, serta antarmuka komunikasi sebagai berikut :

1) Spesifikasi Perangkat Keras

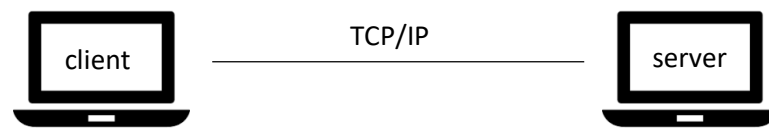
- Intel Core i5-4460S 2.9 GHz (4CPU), RAM 6GB
- Harddisk 500GB
- Windows 10 Pro 64-Bit Version 1809

2) Spesifikasi Perangkat Lunak

- Bahasa Pemrograman : Python Versi 2.7
- *Development Tools* : JetBrains PyCharm, Visual Studio Code, Windows PowerShell

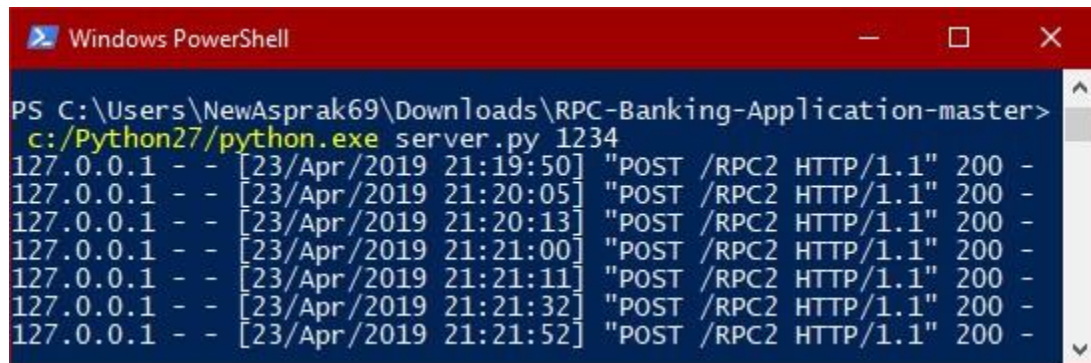
3) Antarmuka Komunikasi

Sistem ini dijalankan dengan lingkup jaringan internet ataupun intranet yang berbasis *Transmission Control Protocol/Internet Protocol (TCP/IP)*. Penggunaann program ini dapat dilakukan dengan menjalankannya pada Terminal Bash dengan sarana komputer dengan terhubung pada internet.



Gambar 2. Antarmuka Jaringan Komunikasi

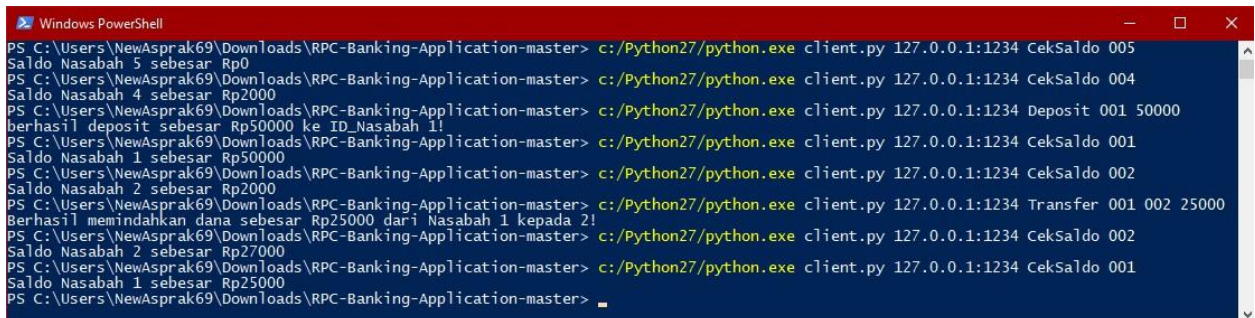
3. Hasil *Running* Program



```
Windows PowerShell
PS C:\Users\NewAsprak69\Downloads\RPC-Banking-Application-master> c:/Python27/python.exe server.py 1234
127.0.0.1 - - [23/Apr/2019 21:19:50] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [23/Apr/2019 21:20:05] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [23/Apr/2019 21:20:13] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [23/Apr/2019 21:21:00] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [23/Apr/2019 21:21:11] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [23/Apr/2019 21:21:32] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [23/Apr/2019 21:21:52] "POST /RPC2 HTTP/1.1" 200 -
```

Gambar 3. Server Teller Dijalankan

Gambar 3, merupakan hasil *running* sebagai *server* dimana untuk melakukan *running* kita harus menentukan *port* berapa yang akan digunakan. Setelah dijalankan maka *client* dapat langsung mengakses sistem yang telah dibangun.



```
Windows PowerShell
PS C:\Users\NewAsprak69\Downloads\RPC-Banking-Application-master> c:/Python27/python.exe client.py 127.0.0.1:1234 CekSaldo 005
Saldo Nasabah 5 sebesar Rp0
PS C:\Users\NewAsprak69\Downloads\RPC-Banking-Application-master> c:/Python27/python.exe client.py 127.0.0.1:1234 CekSaldo 004
Saldo Nasabah 4 sebesar Rp2000
PS C:\Users\NewAsprak69\Downloads\RPC-Banking-Application-master> c:/Python27/python.exe client.py 127.0.0.1:1234 Deposit 001 50000
Berhasil deposit sebesar Rp50000 ke ID_Nasabah 1!
PS C:\Users\NewAsprak69\Downloads\RPC-Banking-Application-master> c:/Python27/python.exe client.py 127.0.0.1:1234 CekSaldo 001
Saldo Nasabah 1 sebesar Rp50000
PS C:\Users\NewAsprak69\Downloads\RPC-Banking-Application-master> c:/Python27/python.exe client.py 127.0.0.1:1234 CekSaldo 002
Saldo Nasabah 2 sebesar Rp2000
PS C:\Users\NewAsprak69\Downloads\RPC-Banking-Application-master> c:/Python27/python.exe client.py 127.0.0.1:1234 Transfer 001 002 25000
Berhasil memindahkan dana sebesar Rp25000 dari Nasabah 1 kepada 2!
PS C:\Users\NewAsprak69\Downloads\RPC-Banking-Application-master> c:/Python27/python.exe client.py 127.0.0.1:1234 CekSaldo 002
Saldo Nasabah 2 sebesar Rp27000
PS C:\Users\NewAsprak69\Downloads\RPC-Banking-Application-master> c:/Python27/python.exe client.py 127.0.0.1:1234 CekSaldo 001
Saldo Nasabah 1 sebesar Rp25000
PS C:\Users\NewAsprak69\Downloads\RPC-Banking-Application-master>
```

Gambar 4. Client Teller Dijalankan, dan melakukan beberapa simulasi perintah

Gambar 4, merupakan hasil *running* sebagai *client*. *Client* dapat mengakses sistem setelah *server* dijalankan. Disini terdapat 4 menu utama yaitu setoran(deposit), penarikan, cek saldo dan transfer.