**LAPORAN PEMROGRAMAN JARINGAN**

**“APLIKASI CRUD DATABASE MAHASISWA MENGGUNAKAN TEKNOLOGI RMI”**



“Disusun untuk memenuhi salah satu tugas Pemrograman Jaringan.”

**Disusun oleh :**

Arry Febryan 201610225200  
Yusuf Bagas Satria 201610225257

Mata Kuliah : Pemrograman Jaringan  
Dosen Pembimbing : Rakhmat Purnomo, S.Pd., S.Kom., M.Kom.

**PROGRAM STUDI INFORMATIKA FAKULTAS TEKNIK  
UNIVERSITAS BHAYANGKARA JAKARTA RAYA**

**JANUARI 2019**

1. **Tujuan**

Tujuan dibuatnya laporan ini adalah:

1. Dibuatnya project ini bertujuan untuk menyelesaikan tugas akhir dari mata kuliah Pemrograman Jaringan.
2. Agar membantu memberi pemahaman kepada Mahasiswa bagaimana proses dari aplikasi client-server.
3. Agar Mahasiswa dapat membangun, dan tahu bagaimana cara membuat aplikasi database (CRUD) dengan menggunkan teknologi RMI.
4. **Remote Method Invocation (RMI)**

RMI (*Remote Method Invocation*) menyediakan sarana dimana client dan server dapat berkomunikasi dan saling bertukar infomasi. RMI memungkinkan pengembang perangkat lunak untuk merancang aplikasi terdistribusi dimana methods dari remote object dapat dipanggil dari JVM (*Java Virtual Mechine*) lain, yang mungkin berjalan pada host yang berbeda. Remote object adalah obyek dalam Java yang dapat direferensikan secara remote. Pemrogram seakan- akan memanggil methods lokal dari file kelas lokal, sedang dalam kenyataannya semua argumen dikirimkan ke remote target dan diinter-pretasikan, kemudian hasilnya dikirimkan kembali ke pemanggil. Dalam RMI, server akan membuat remote objects, membuat referensi, dan menunggu client untuk memanggil methods dari remote object ini. Sedangkan client akan mendapatkan remote reference dari satu atau lebih remote object dan memanggil methods untuk remote object tersebut.

1. **Tutorial Membuat Program CRUD dengan Remote Method Invocation**

Bismillaahir-rohmaanir-rohiim.

Pada tulisan ini, saya akan membuat tutorial pembuatan aplikasi desktop sederhana berbasis client-server dengan Bahasa Pemrograman Java, dan menggunakan teknologi RMI (Remote Method Invocation). Dalam Tutorial kali ini saya akan membuat aplikasi yang dapat melakukak proses CRUD untuk database Mahasiswa. Semoga tutorial ini dapat membantu dan memberikan pemahaman kepada para pembaca. Aamiin.

Sebelum memulai membuat program ada beberapa hal yang perlu dipersiapkan terlebih dahulu, yaitu sebuah laptop yang di dalamnya terinstal:

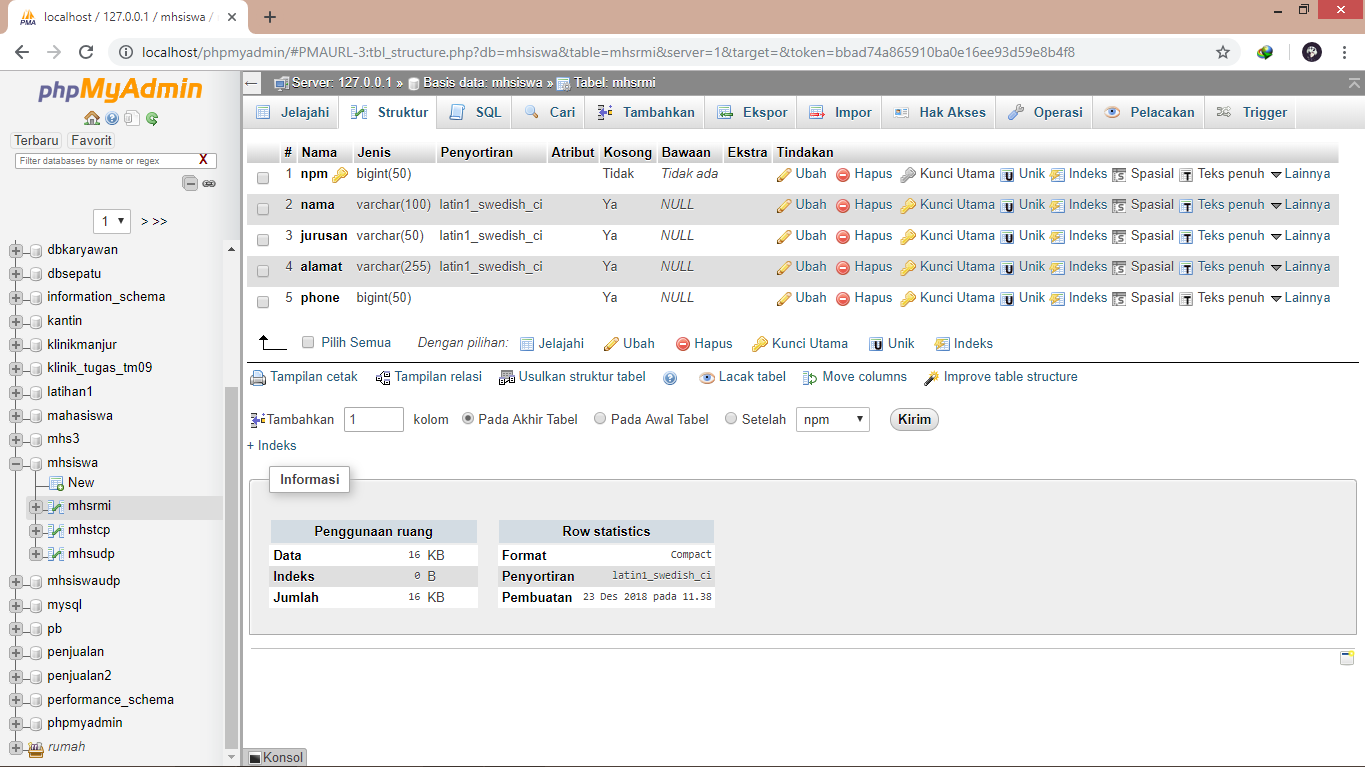
* Java Netbeans
* dan XAMPP

Setelah kebutuhan diatas terpenuhi, baru kita bisa melanjutkan ke langkah-langkah berikutnya.

1. **Membuat Database**

Langkah pertama yang harus kita lakukan adalah menyiapkan terlebih dahulu sebuah database beserta tabel dan entias-entitasnya. Nantinya database ini akan digunakan untuk menyimpan data-data yang akan di-input.

Disini saya akan membuat sebuah database dengan nama **mhsiswa** beserta struktur tabel sebagai berikut:



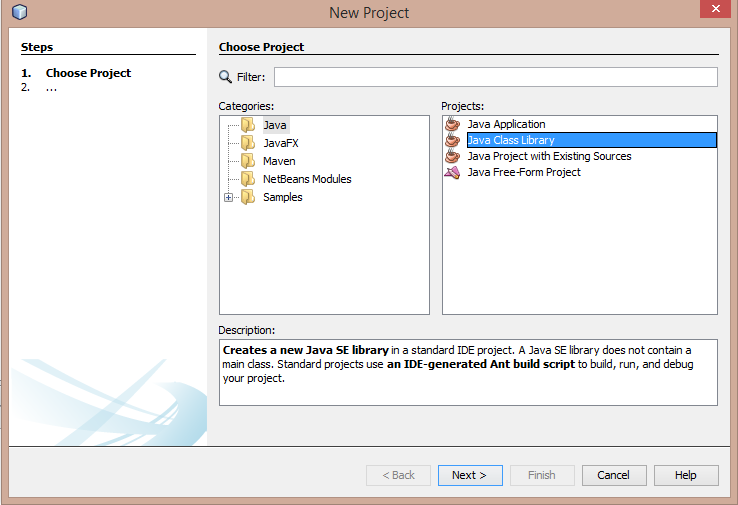
1. **Membuat Project Java**

Langkah selanjutnya adalah menyiapkan project java pada netbeans. Pada program kali ini kita akan membuat 3 project baru yaitu **Application-API, Application-Server** dan **Application-Client.**

* 1. **Application API**

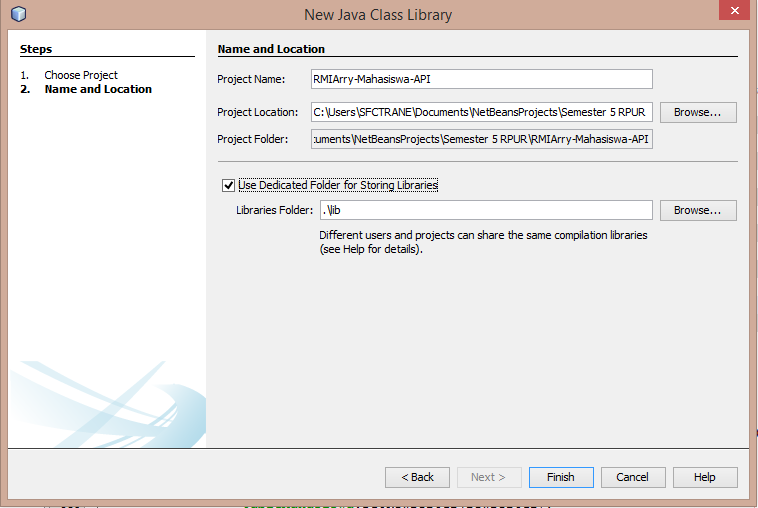
Porject pertama yang kita buat adalah project untuk **Application-API** yang nantinya project ini akan digunakan oleh server dan client sebagai remote object.

Untuk membuat project klik **File > New Project**, nanti akan muncul tampilan seperti ini:

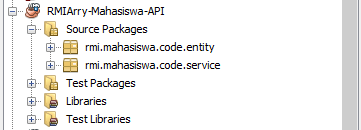


Pilih **Java Class Library** untuk project Application-API, lalu klik next.

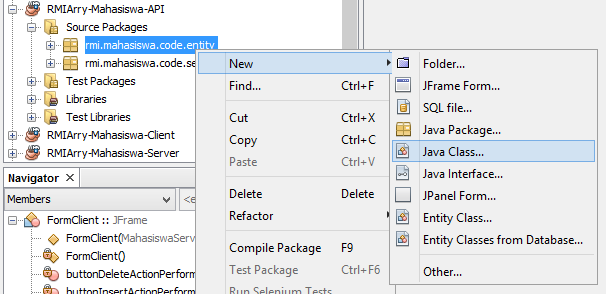
Setelah itu isilah nama project sesuai dengan yang kita inginkan, lalu klik finish. Untuk memudahkan, beri codename ‘API’ pada akhiran nama project API. Di sini saya memberi nama project API dengan nama **‘RMIArry-Mahasiswa-API’**.



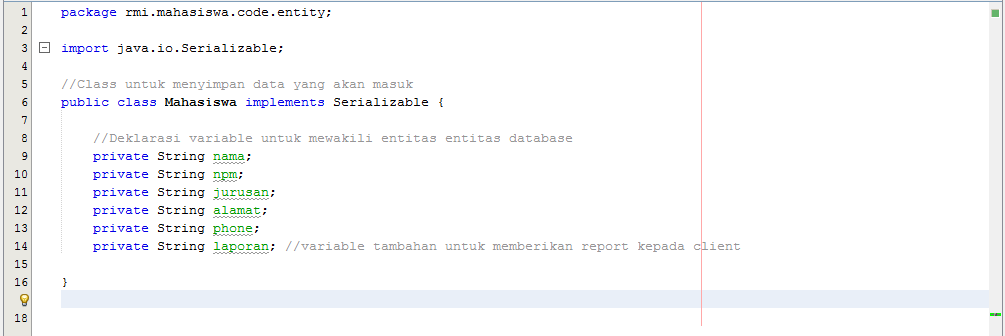
Selanjutya buatlah 2 buah package seperti gambar dibawah ini.



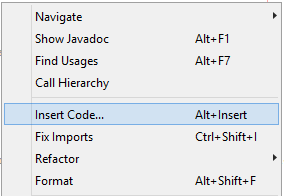
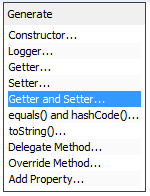
Setelah itu buatlah sebuah Java Class dengan nama ***Mahasiswa*** pada package entity dengan cara klik kanan pada package lalu klik **New > Java Class.**



Kemudian buatlah variable didalam class tersebut untuk mewakili tiap tiap entitas yang sebeumnya telah kita buat pada table dalam database.

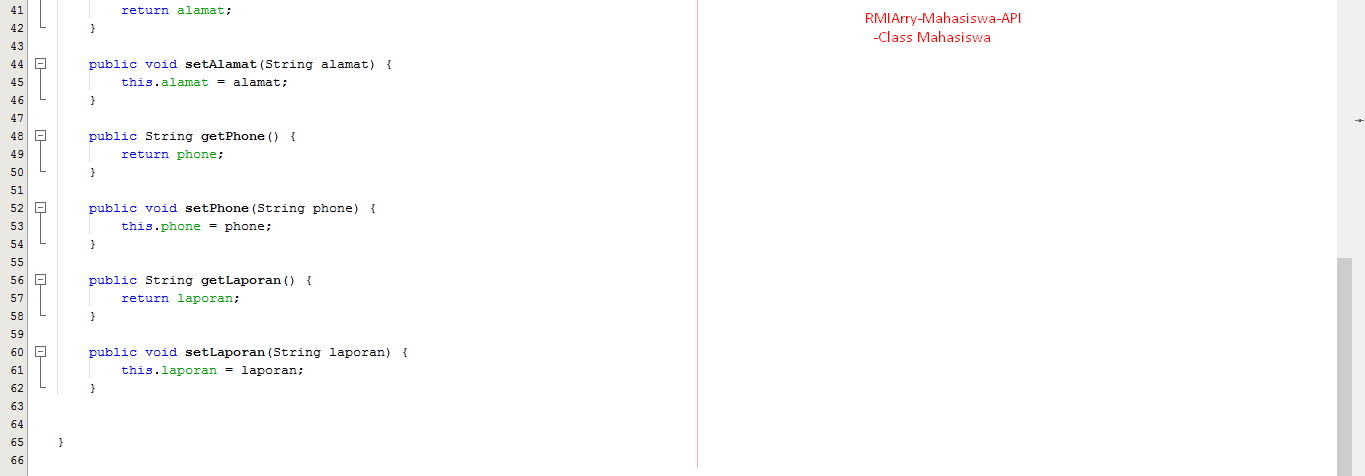
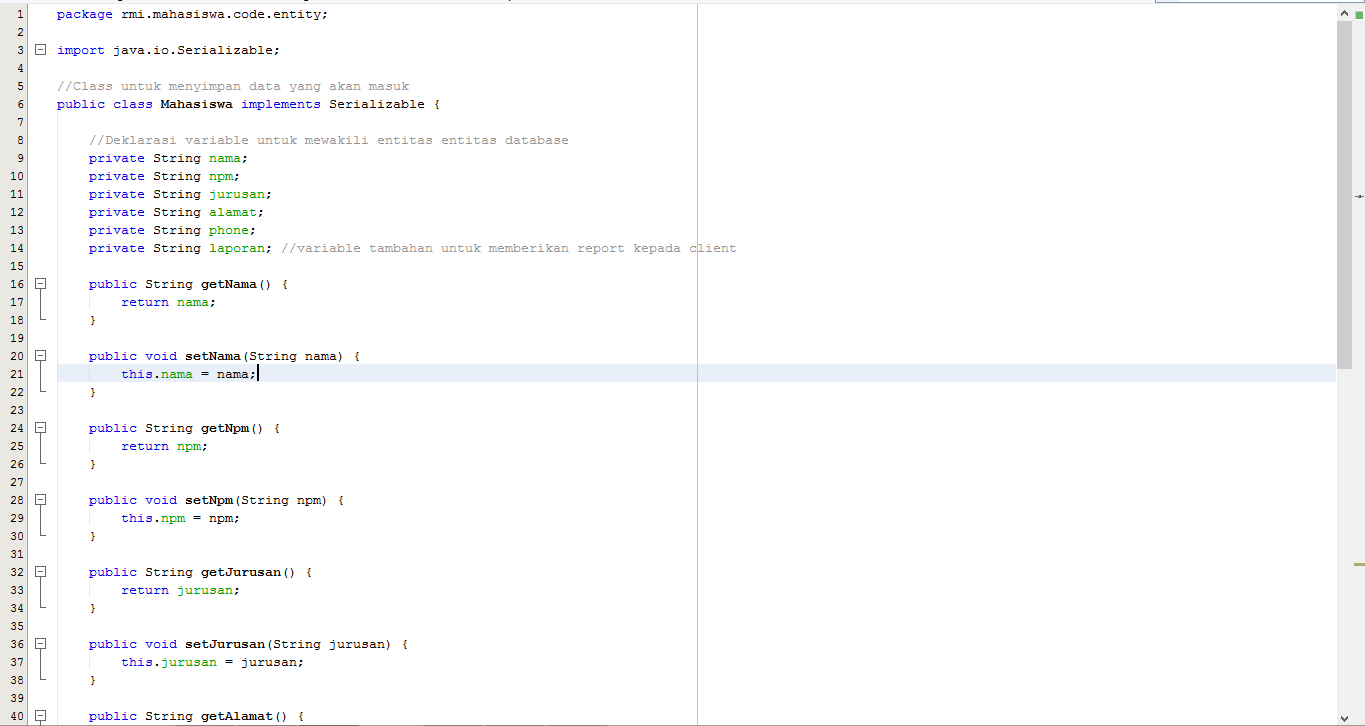


Selanjutnya buatlah method **‘Set’** dan **‘Get’** untuk tiap variable dengan cara klik kanan pada kolom source code pilih **Insert Code** > **Getter and Setter.**

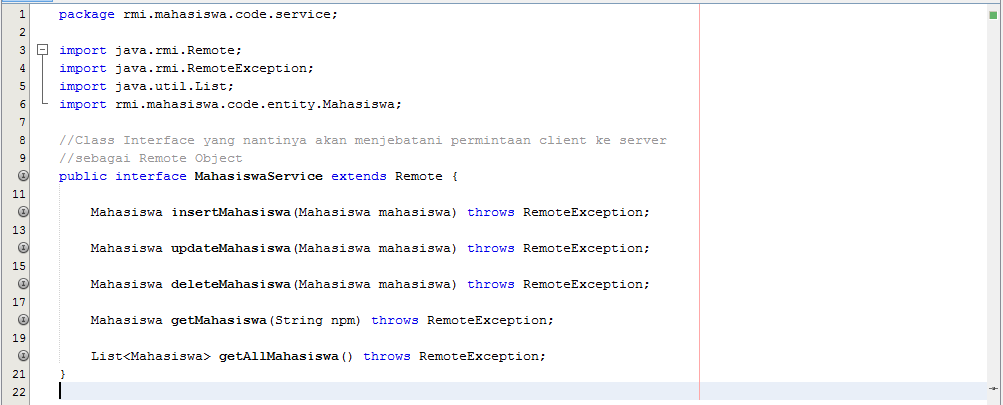
Lalu centang semua variable yang telah kita buat di class Mahasiswa, klik Generate.

Setelah itu akan muncul method Get dan Set seperti gambar dibawah ini:



Langkah selanjutnya adalah membuat kelass interface dengan nama ***MahasiswaService*** pada package service, kelas inilah yang akan menjebatani antara client dan server.

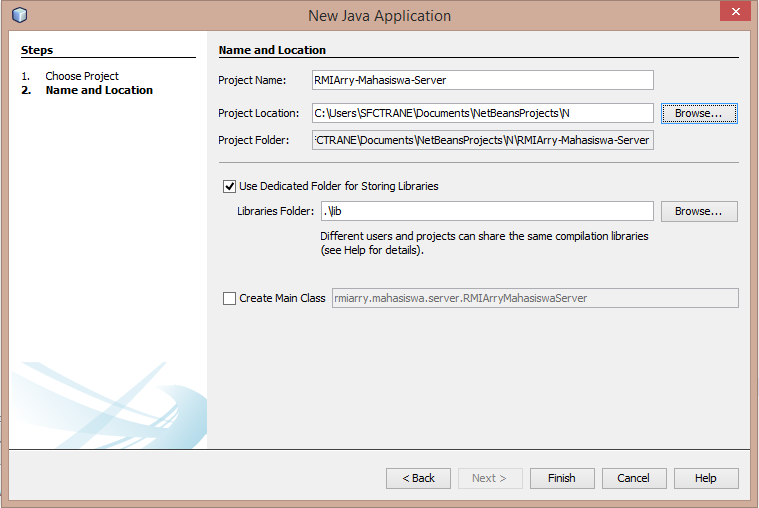
Klik kanan pada package service lalu pilih **New > Java Interface**. Lalu tulis kodingan dibawah ini:



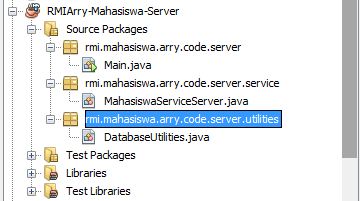
* 1. **Application-Server**

Setelah project Application-API selesai langkah selanjutnya adalah membuat project untuk **Application-Server** dengan cara klik **File > New Project > pilih Java Application**.

Lalu beri nama untuk project Application-Server sesuai dengan yang kita inginkan, beri codename Server diakhiran nama project untuk mempermudah kita mengenali project, disini saya memberi nama untuk project Server dengan nama **RMIArry-Mahasiswa-Server**.

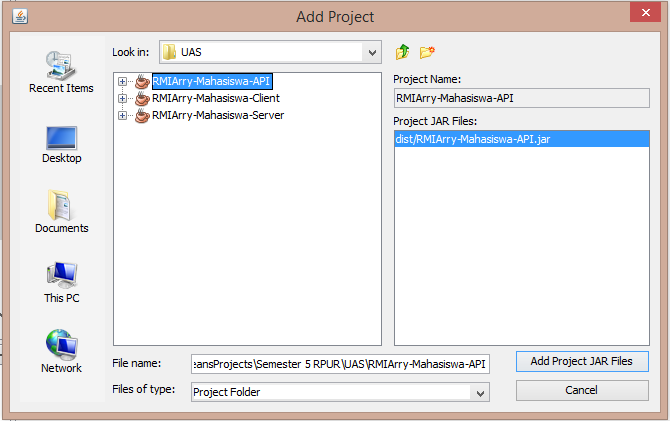


Setelah project terbuat, buatlah tiga buah project beserta classnya seperti dibawah ini:

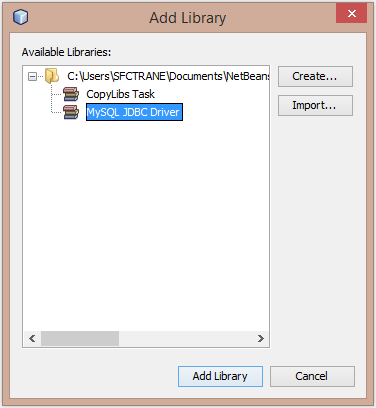


Kemudian import project Application-API yang sebelumnya kita buat kedalam project Server dengan cara klik kanan pada **Libraries > Add Project..**

Akan muncul tampilan seperti dibawah ini, pilih project Application-API lalu klik Add Project.



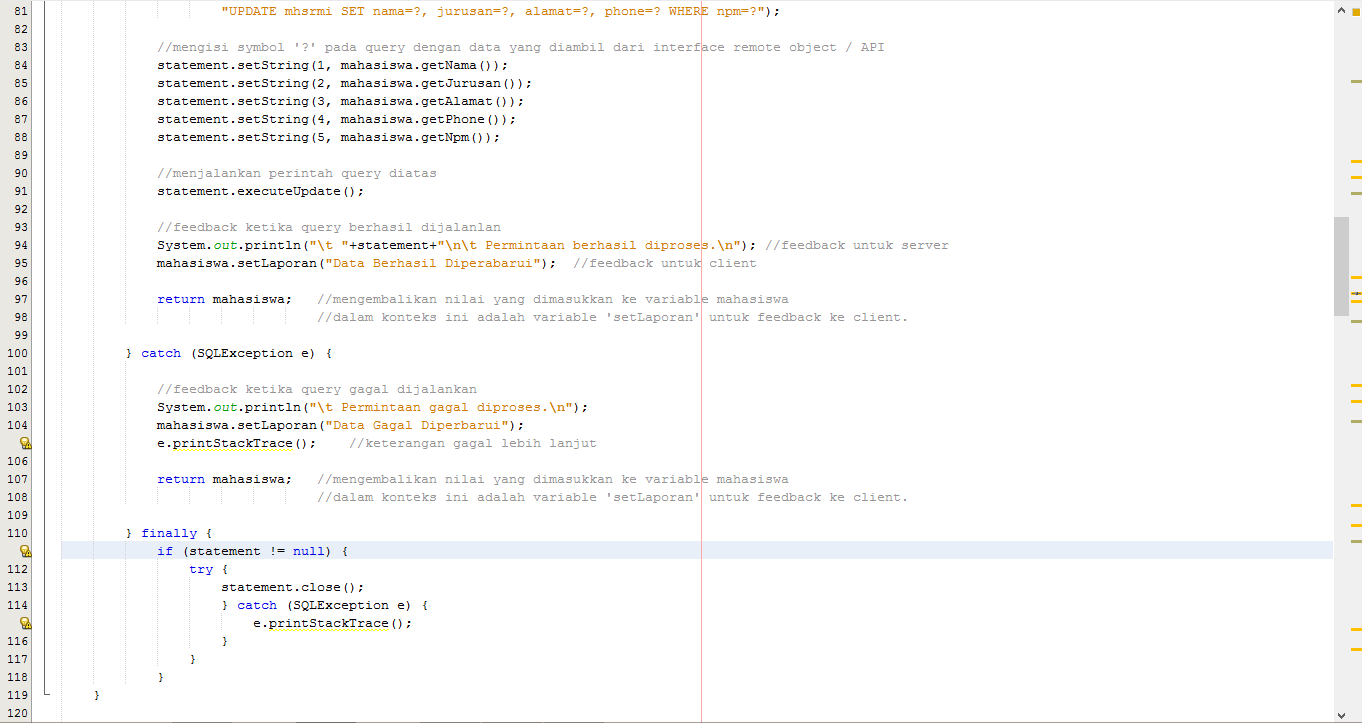
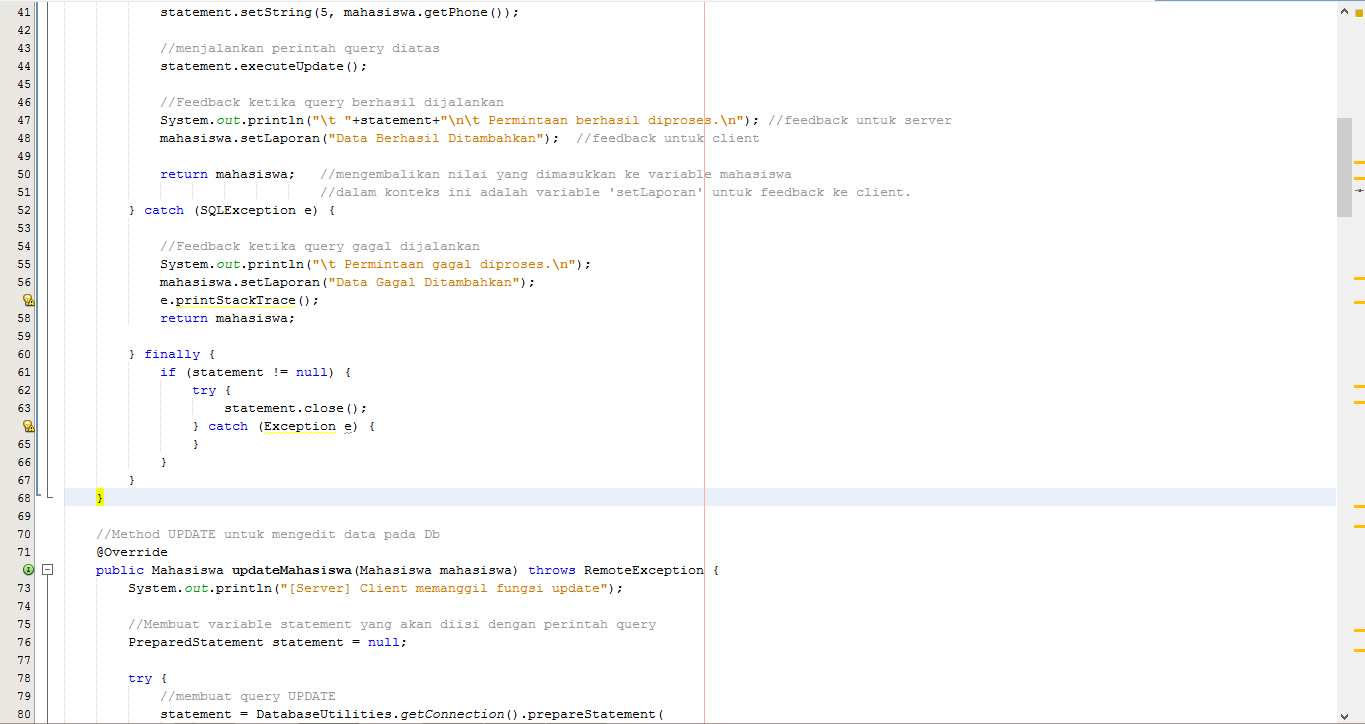
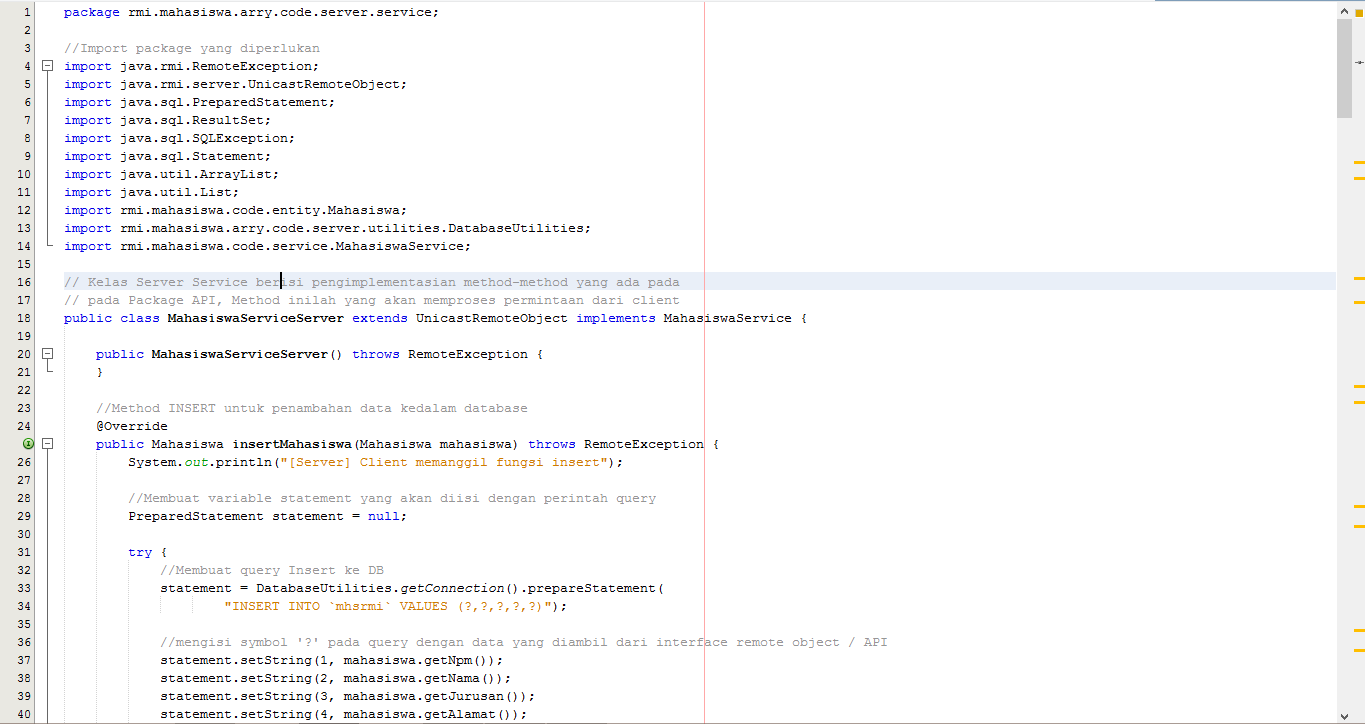
Selanjutnya import driver MySQL JDBC dengan cara klik kanan pada **Libraries > Add Librarys**

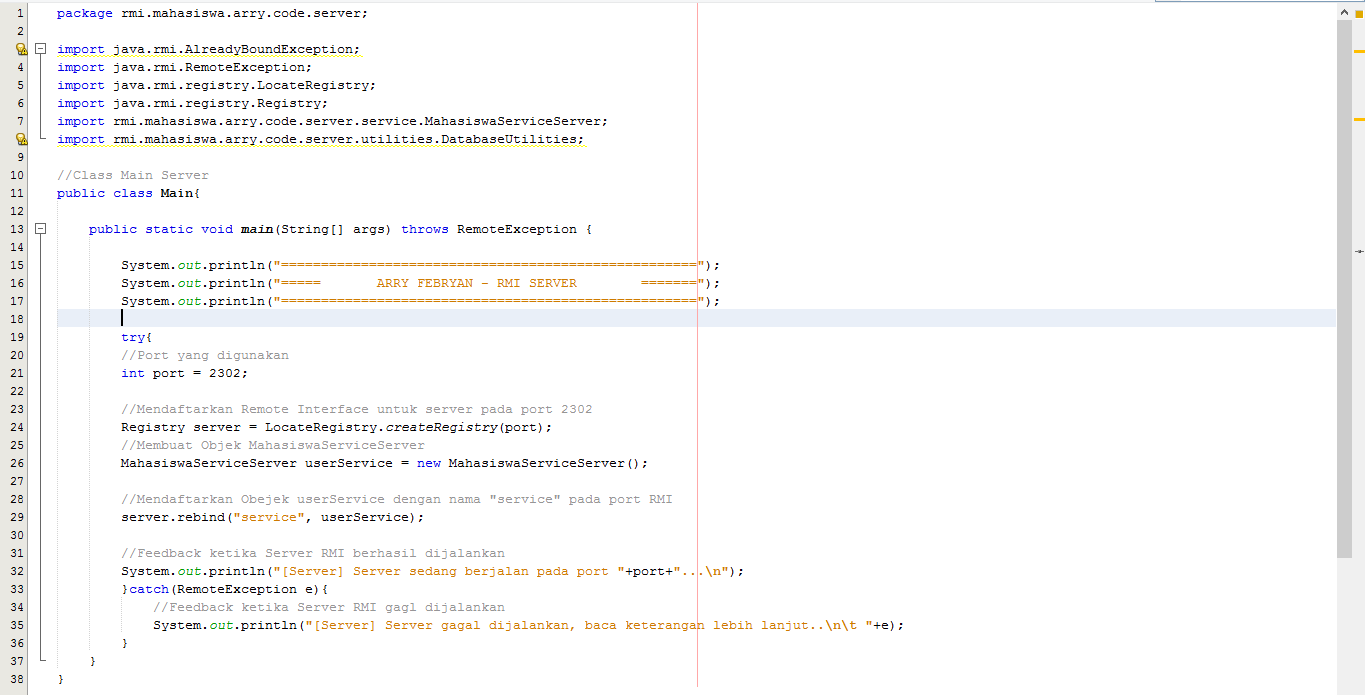


Lalu pilih **MySQL JDBC Driver**.

Langkah selanjutnya adalah mengisi class-class yang ada pada project Server dengan kode program sebagai berikut:

**Class Utilities:** 

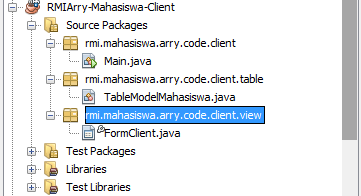
**Class Service:**   

**Class Main:**

* 1. **Application Client**

Selanjutnya buatlah sebuah project untuk Application-Client dengan cara yang sama seperti membuat project untuk server.

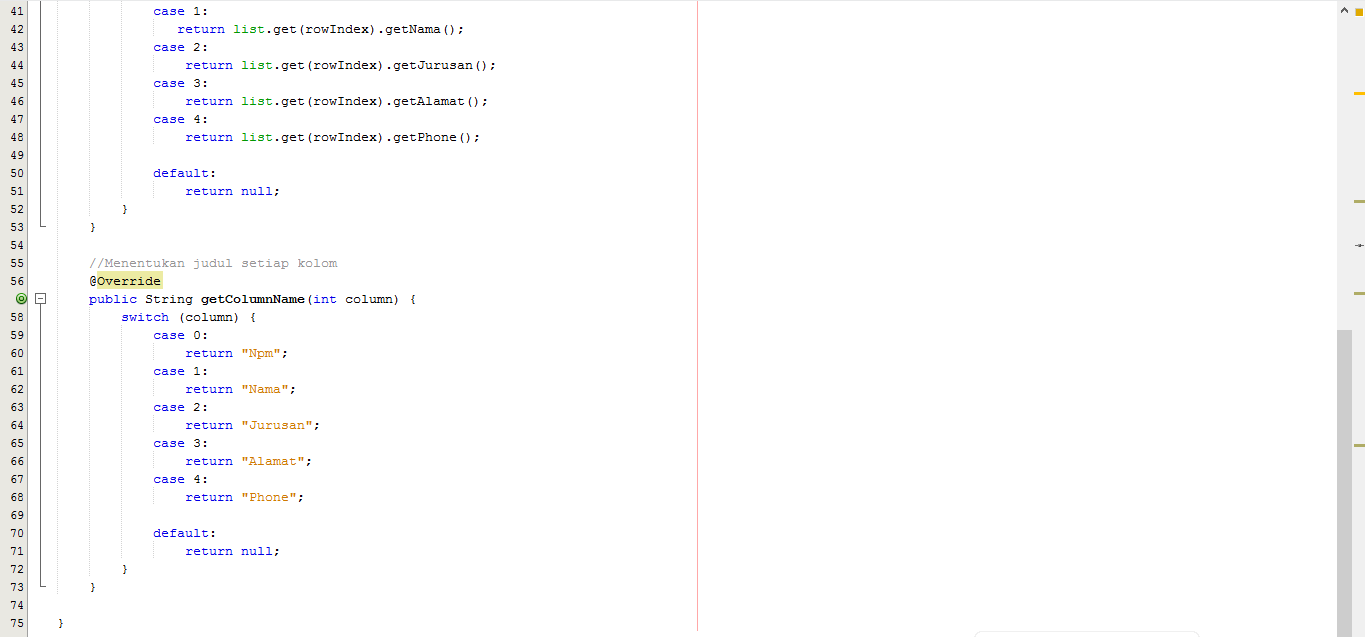
Kemudian buatlah struktur package beserta classnya seperti dibawah ini:

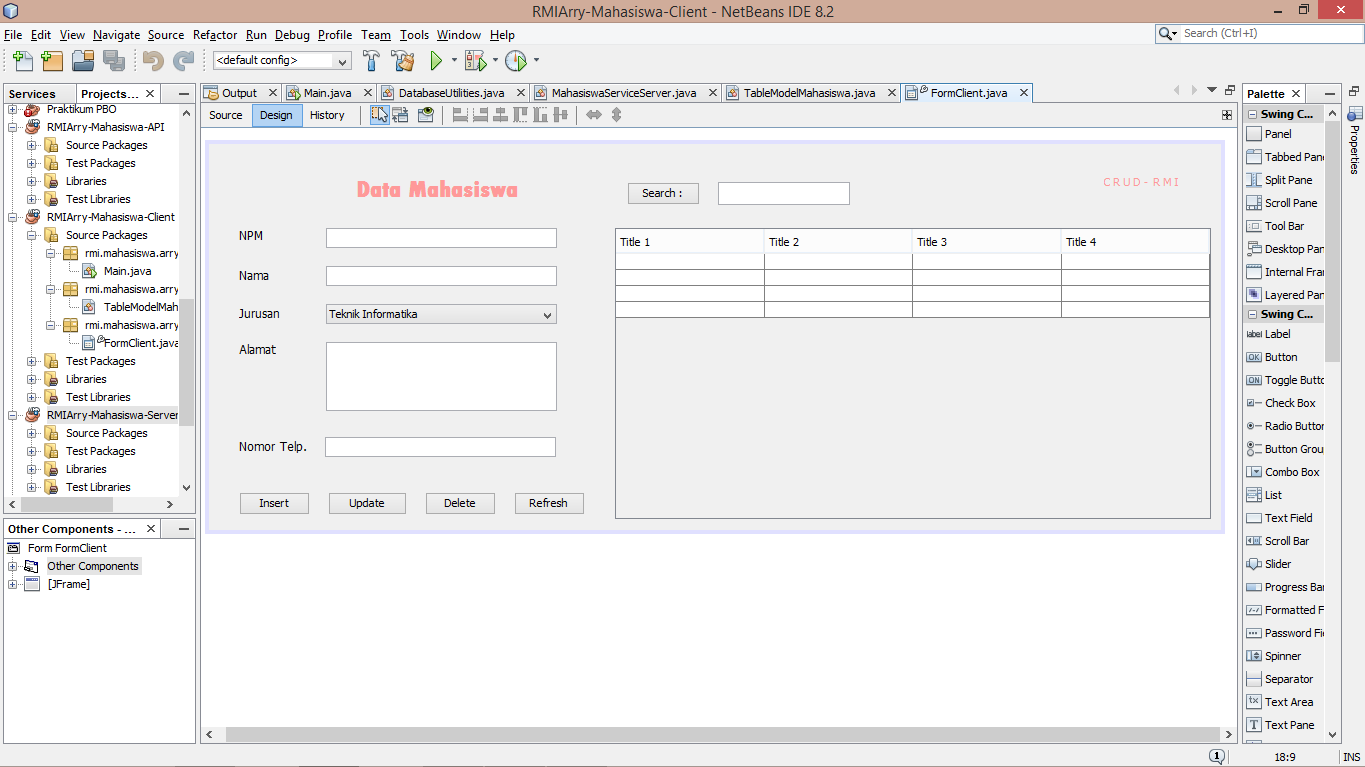


Untuk FormClient kita buat dengan Jframe Form sebagai tatap muka kepada user client.

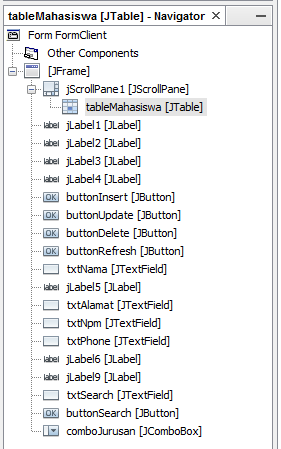
Berikutnya import project Application-API kedalam project client dengan cara yang sama seperti sebelumnya.

Selanjutnya isilah class-class yang ada pada project Client dengan kode program sebagai berikut:

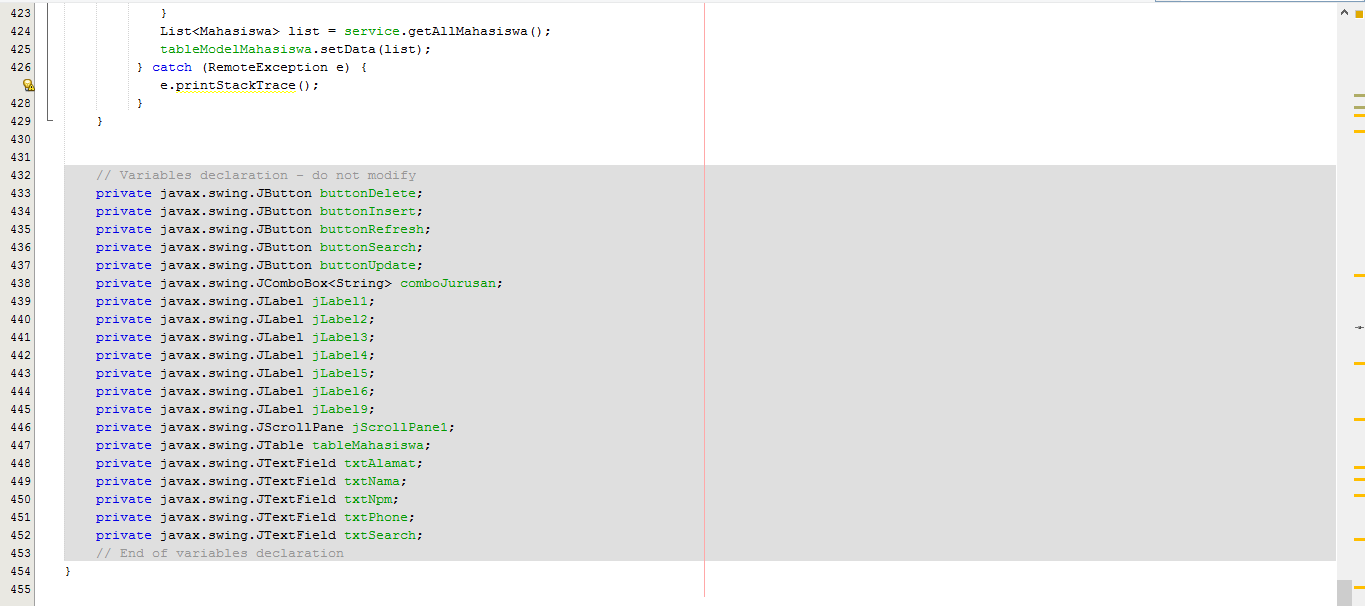
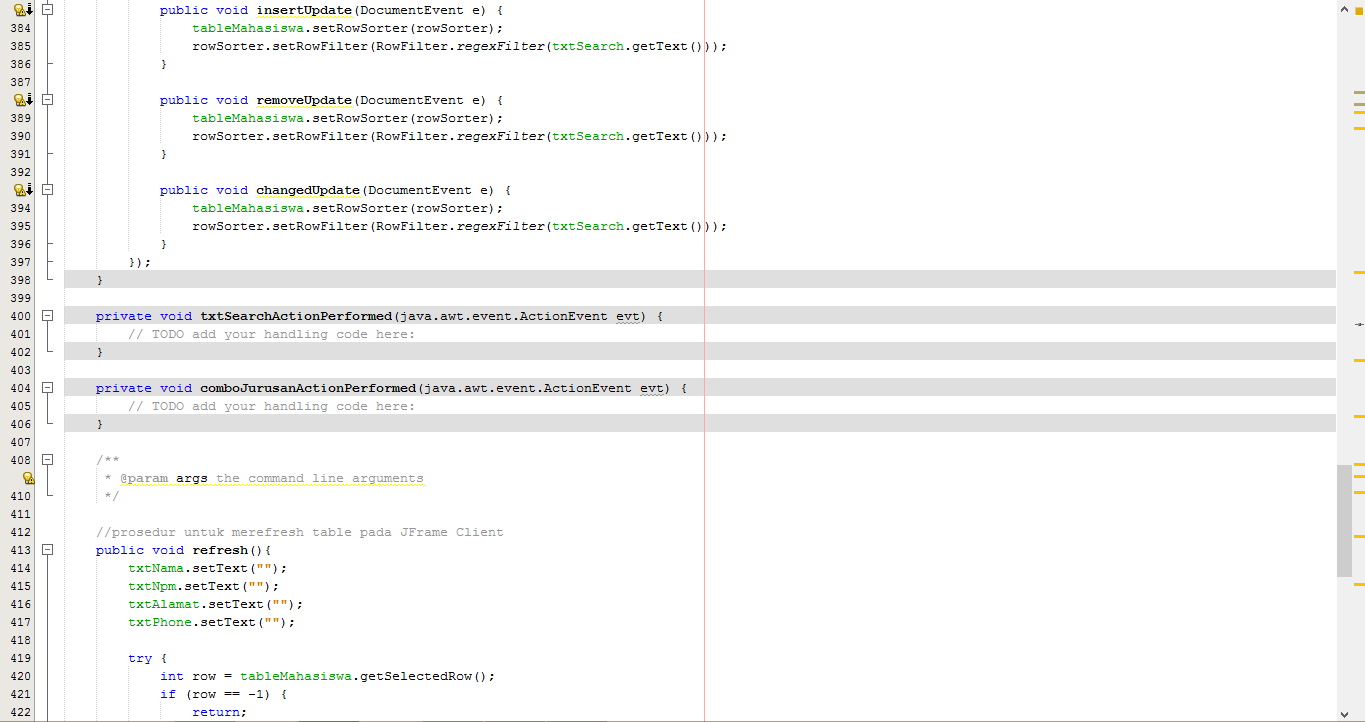
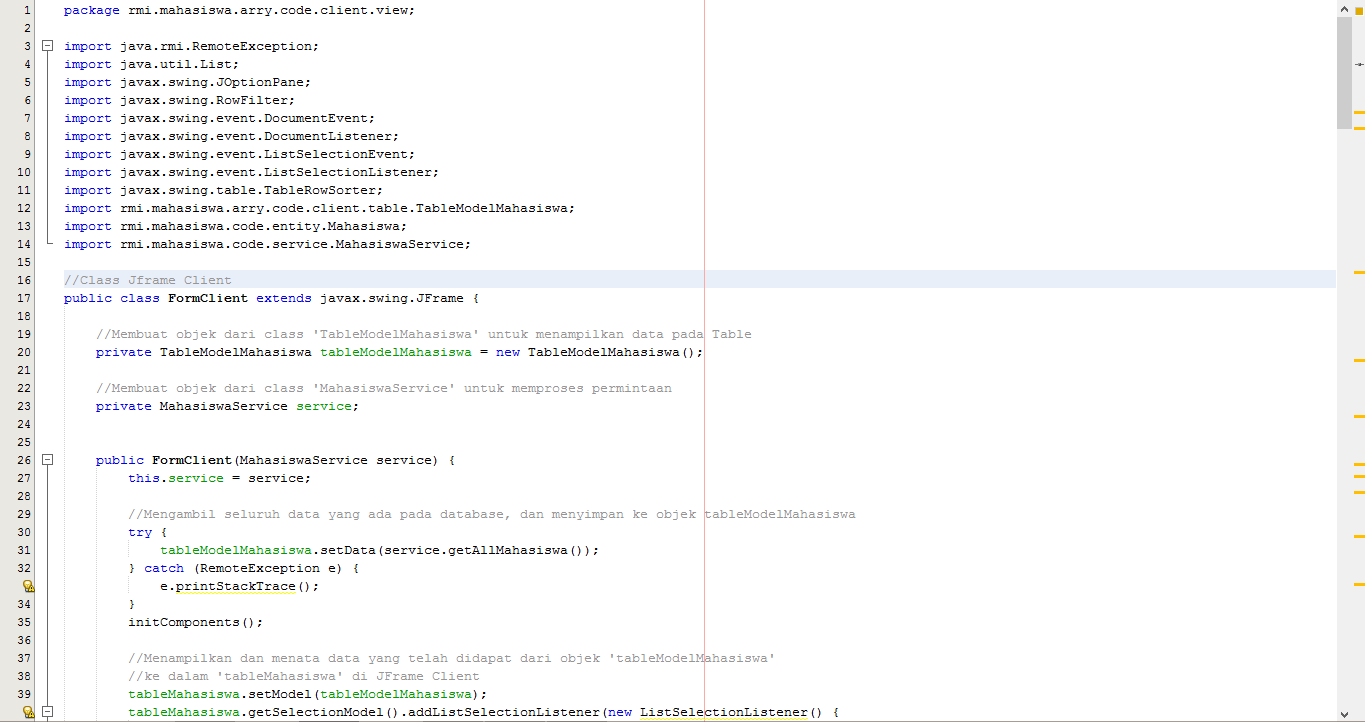
**TableModelMahasiswa:** 

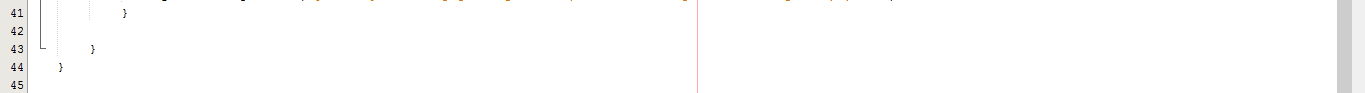
Buatlah desain untuk antarmuka client pada **Jframe FormClient.**

Dengan penamaan setiap variable sebagai berikut:



Selanjutnya tulislah kode program untuk **Jframe FormClient** seperti dibawah ini:

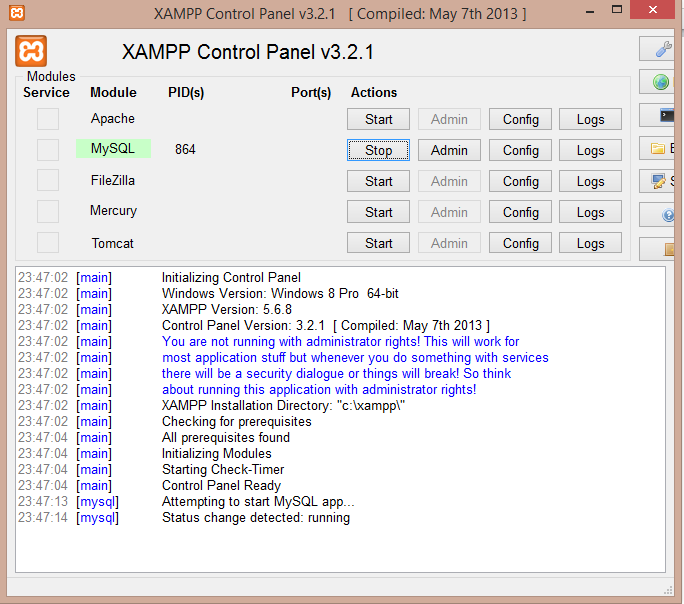


**Class Main:** 

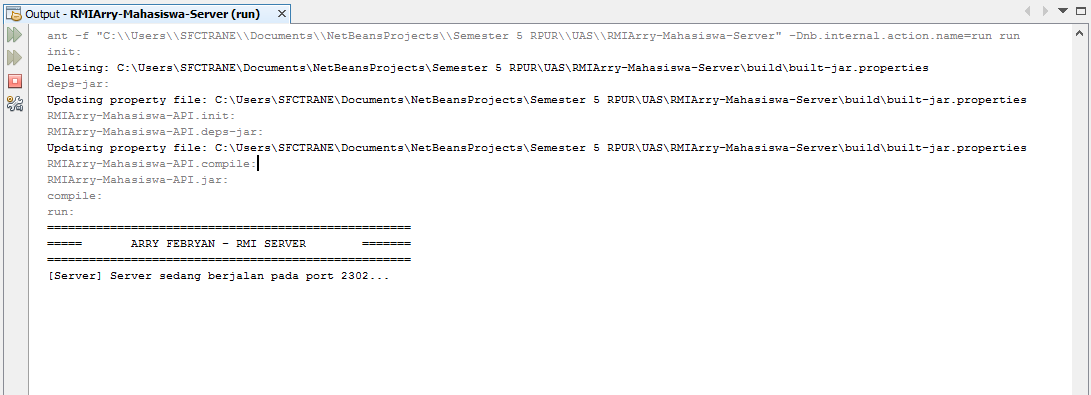
**Selesai**

1. **Testing Aplikasi**

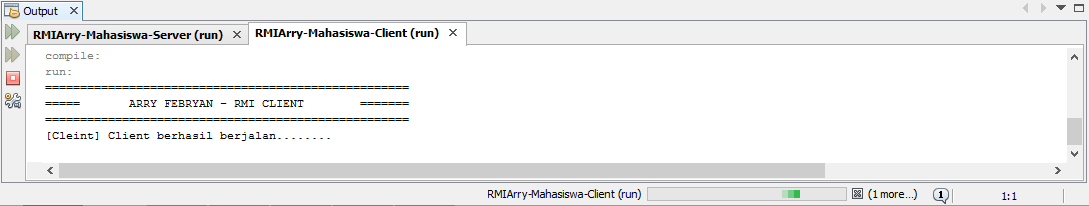
Langkah pertama, pastikan modul MySQL pada XAMPP telah berjalan.

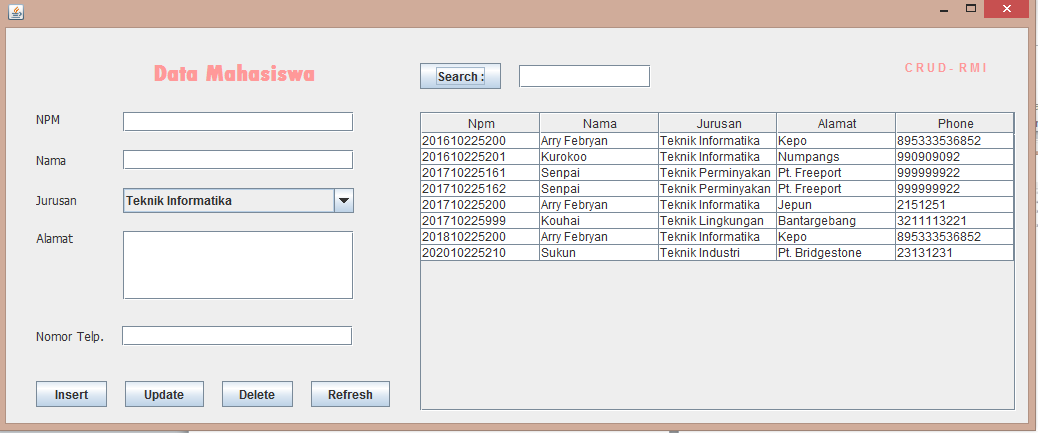


Kemudian jalankan program server terlebih dahulu.

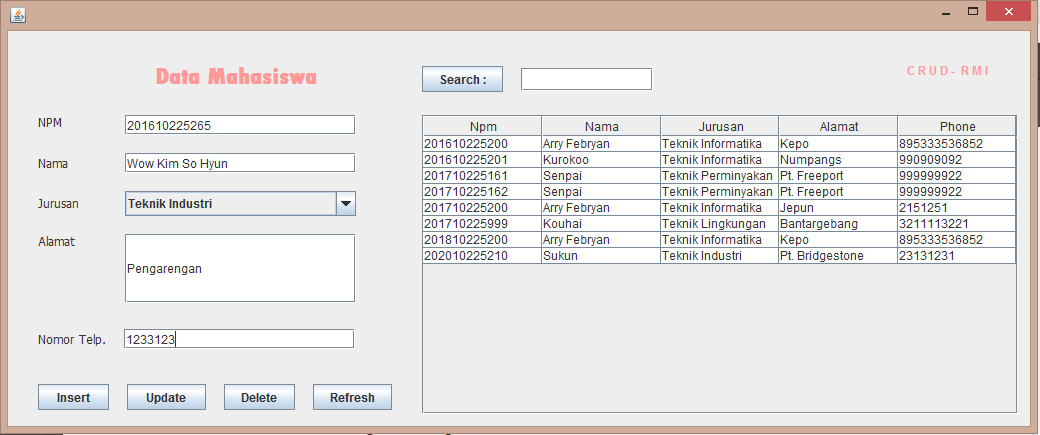


Selanjutnya jalankan program client.

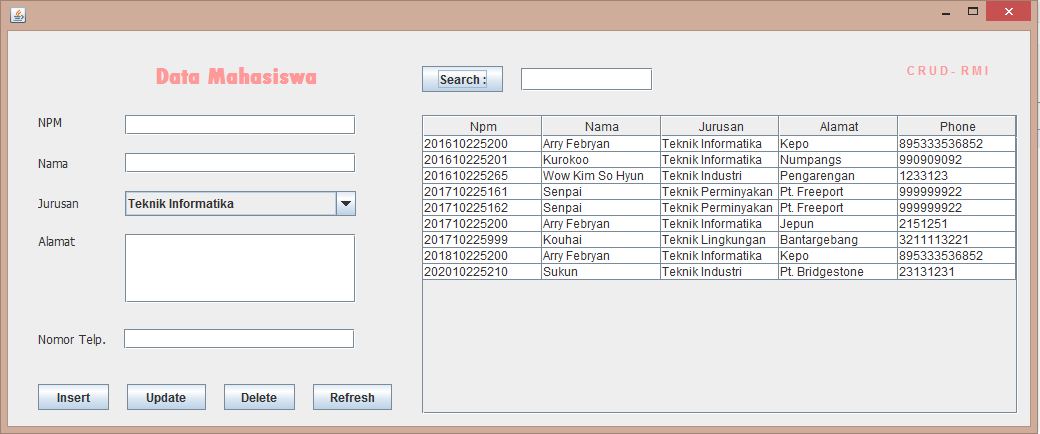




Test input mahasiswa baru:



Hasil:



1. **Referensi**

Maulana Achmad, “Membuat Aplikasi Desktop Client Server Sederhana dengan Java RMI” 2016.

Ratnasari Nur Rohmah, “Client/Server dengan Java Remote Method Invocation (Java RMI),” *J. Tek. Elektro Dan Komput. Emit.*, vol. 3, p. 5, 2003.

Purnomo Rakhmat, “Pemrograman Jaringan Remote Method Invocation (RMI)”.

[*http://java.sun.com/javase/6/docs/technotes/guides/rmi/index.html*](http://java.sun.com/javase/6/docs/technotes/guides/rmi/index.html)