

## **PERCOBAAN 3**

### **APLIKASI DATABASE UNTUK PEMROGRAMAN IVR**

#### **3.1. Tujuan :**

Setelah melaksanakan praktikum ini mahasiswa diharapkan mampu :

- Membuat database untuk pendukung layanan IVR menggunakan MySQL
- Membuat koneksi antara pemrograman Database dengan C++
- Menjalankan program IVR dengan aplikasi database

#### **3.2. Peralatan :**

##### **Hardware :**

- PABX untuk penyedia jalur telepon analog
- 2 pesawat Telepon
- 1 PC dilengkapi dengan Dialogic Card (D/4PCI-U)

##### **Software :**

- Visual C++ ver 6.00
- Xampp yang mempunyai aplikasi MySQL
- MultiThread Program, untuk perekaman suara

#### **3.3. Teori :**

##### *3.3.1. Menyiapkan Aplikasi Data Base*

Aplikasi data base diperlukan jika Layanan sistim informasi menggunakan jenis data inventory, yaitu data yang memerlukan inventarisasi, seperti data Pelanggan, data Jenis Barang, data Nilai Mahasiswa dan sebagainya.

Beragam jenis pemrograman Database yang bisa mendukung Layanan IVR berbasis C++, di antaranya MySQL, Oracle maupun Microsoft Access. Pemrograman database yang akan dilakukan pada praktikum ini adalah dengan menggunakan database MySQL, dimana table-tebel yang berisi nilai dari data-data yang akan diambil dan diinformasikan melalui layanan IVR, disusun dan diisi melalui MySQL.

Koneksi antara C++ sebagai bahasa pemrograman aplikasi dengan MySQL sebagai bahasa pemrograman database dilakukan langsung pada project setting di C++, di mana beberapa header file dari MySQL yang perlu diproses di C++ sudah dijadikan satu dalam folder yang sama dengan folder header file-nya C++.

Langkah-langkah yang diperlukan untuk menyiapkan aplikasi data base dalam layanan IVR adalah :

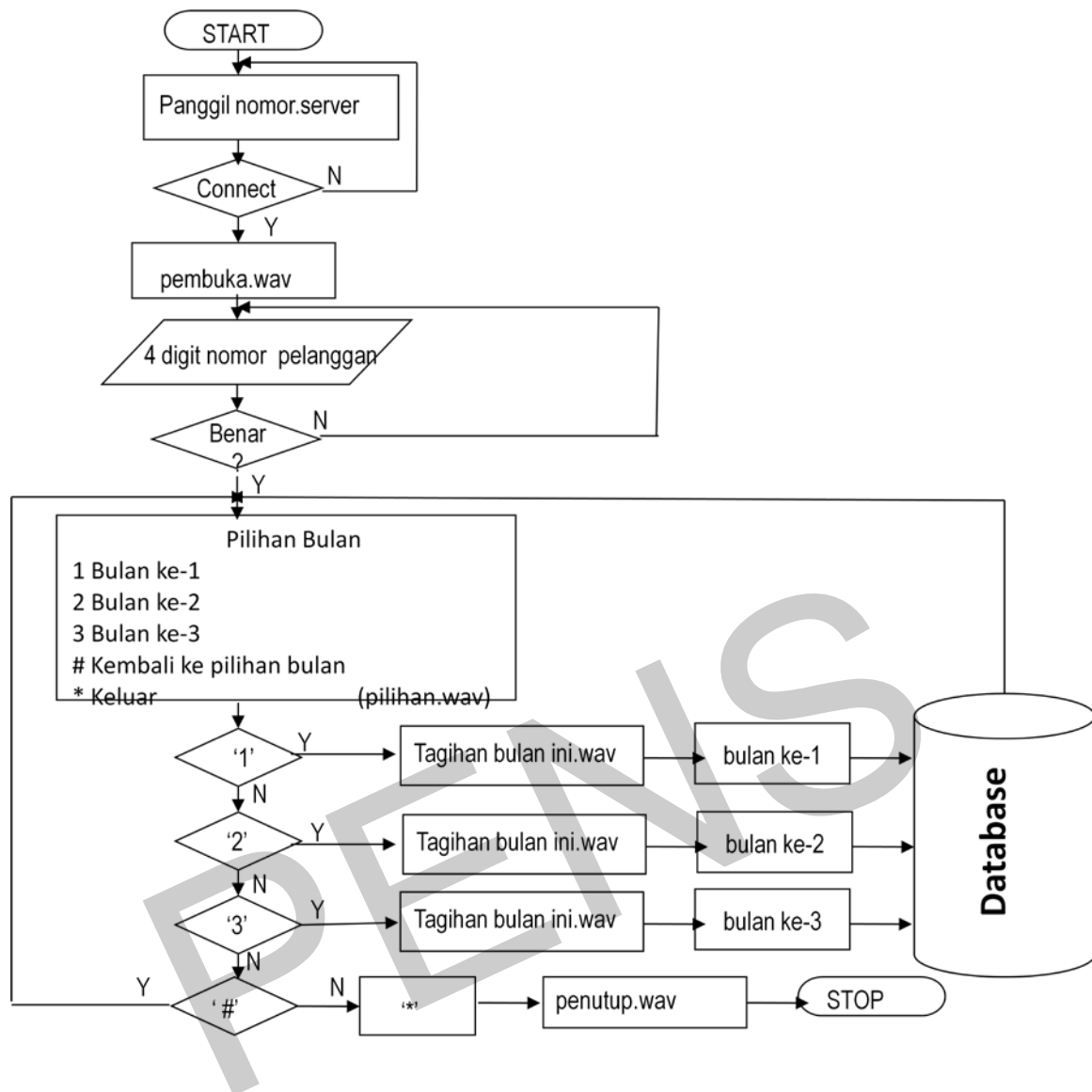
1. Membuat Tabel di dalam database
2. Membuat file .wav untuk dibunyikan
3. Me-run database melalui aplikasi C++
4. Menginformasikan layanan melalui pesawat telepon

### **3.4. Prosedur Percobaan :**

Layanan IVR dengan aplikasi database yang akan dibuat adalah tentang tagihan PLN. Pada sistim ini pelanggan akan dapat mengakses sistim layanan tersebut dengan cara menghubungi nomor IVR server. Setelah terhubung, pelanggan diminta untuk memasukkan nomor rekening PLN (sebesar 4 digit). Jika nomor pelanggan ini benar (berarti terdaftar di database PLN), maka pelanggan diberi kesempatan untuk memilih tagihan bulan ke satu, kedua atau ke tiga. Sesuai dengan digit pilihan pelanggan, IVR akan membunyikan jumlah tagihan untuk nomor rekening tersebut. Sistim layanan akan berakhir jika pelanggan menekan tombol '\*'. Flowchart selengkapnya diberikan pada gambar 3.1

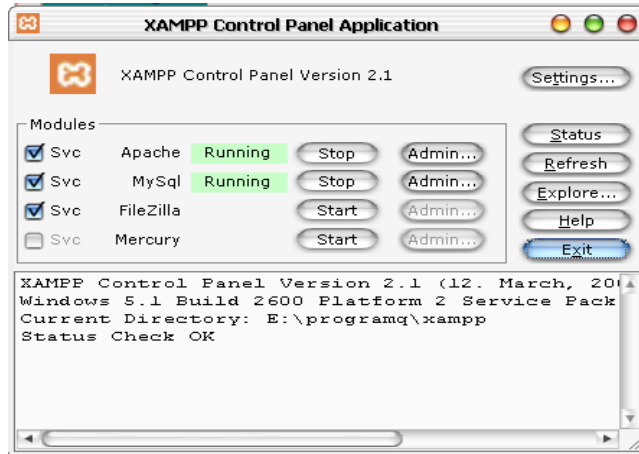
#### **1. Menyiapkan Tabel**

Tabel yang akan dibuat bernama Tagihan, berada di dalam database Tagihan\_PLN. Tabel tersebut terdiri dari 5 field, yang terdiri dari Nomor urut, Rekening, bulan ke-1, bulan ke-2 dan bulan ke-3. Tipe value dari nomor urut adalah *integer*, sedangkan tipe value dari rekening, bulan ke-1, bulan ke-2 dan bulan ke-3 adalah *var char*.



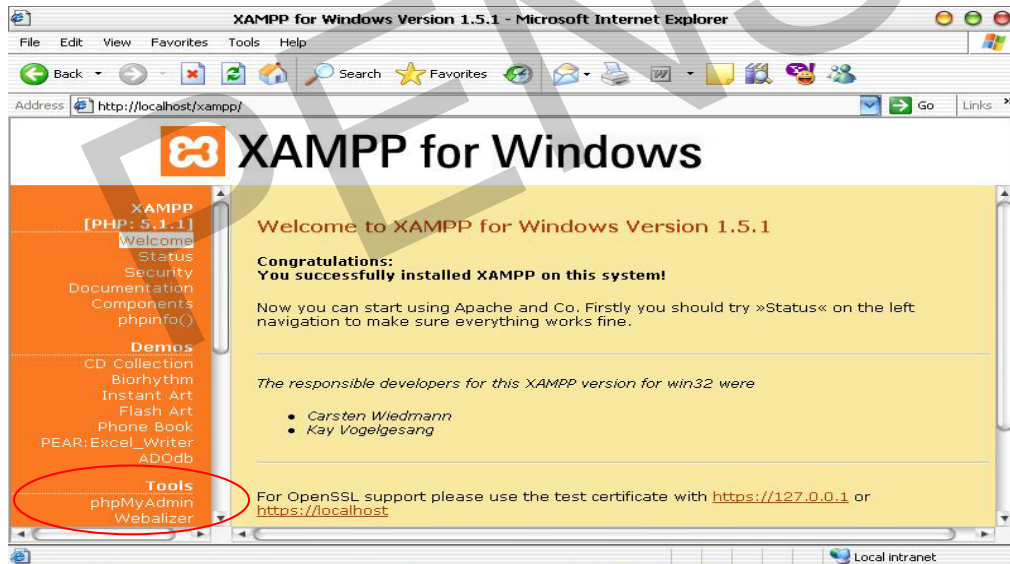
Gambar 3.1. Flow Chart pengaksesan Layanan Tagihan PLN

1. Pada folder Xampp, double klik Xampp\_start dan Xampp\_control untuk memastikan bahwa Apache dan MySQL dalam kondisi start / running.



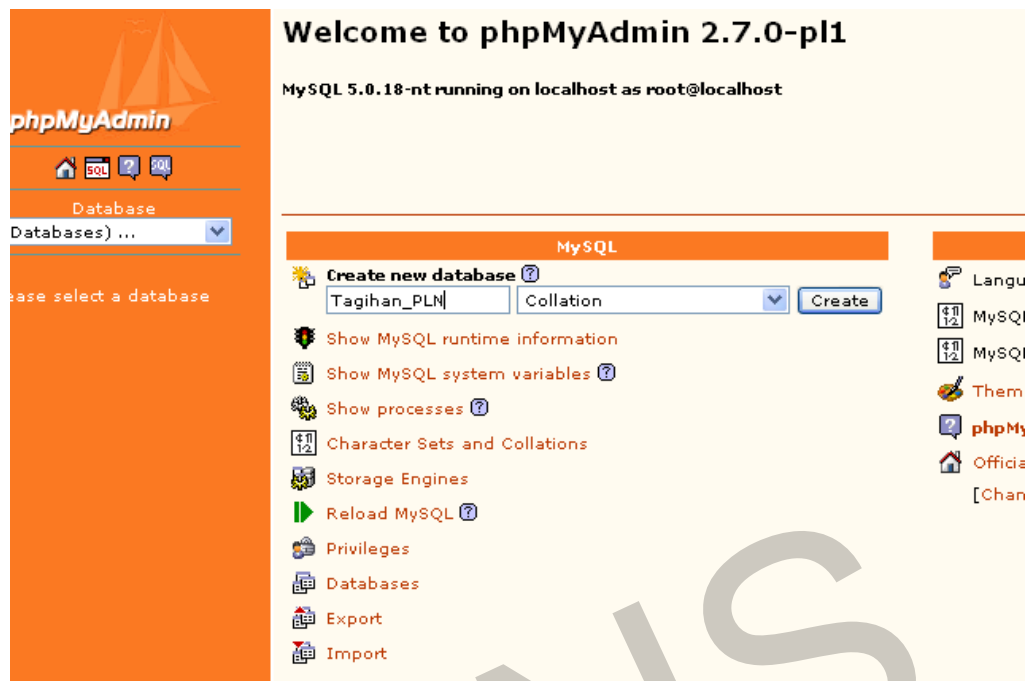
Gambar 3.2. Memulai Xampp

2. Aktifkan phpmyadmin, dengan cara memasuki localhost dulu. Ketik <http://localhost>, pada browser, sebelumnya matikan koneksi ke proxy server melalui **Internet Options**.



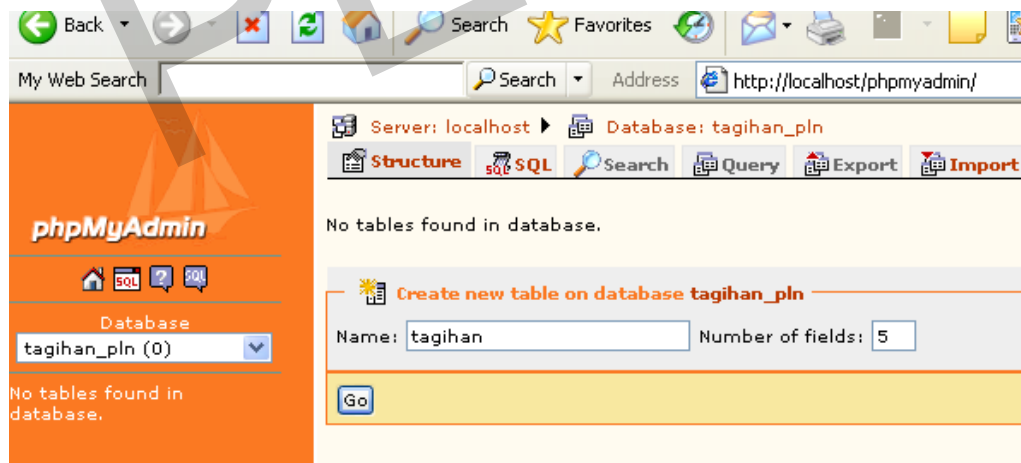
Gambar 3.3. Tampilan xampp

3. Setelah klik phpmyadmin, create new database : *ketik* Tagihan\_PLN, klik **Create**.



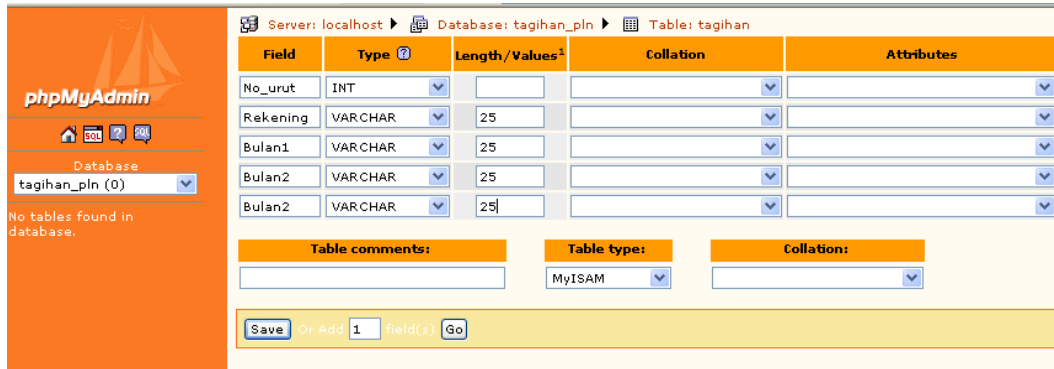
Gambar 3.4. Meng-create database baru

Buat Tabel baru dengan nama Tagihan, tabel tersebut berisi 5 field. Klik **Go**



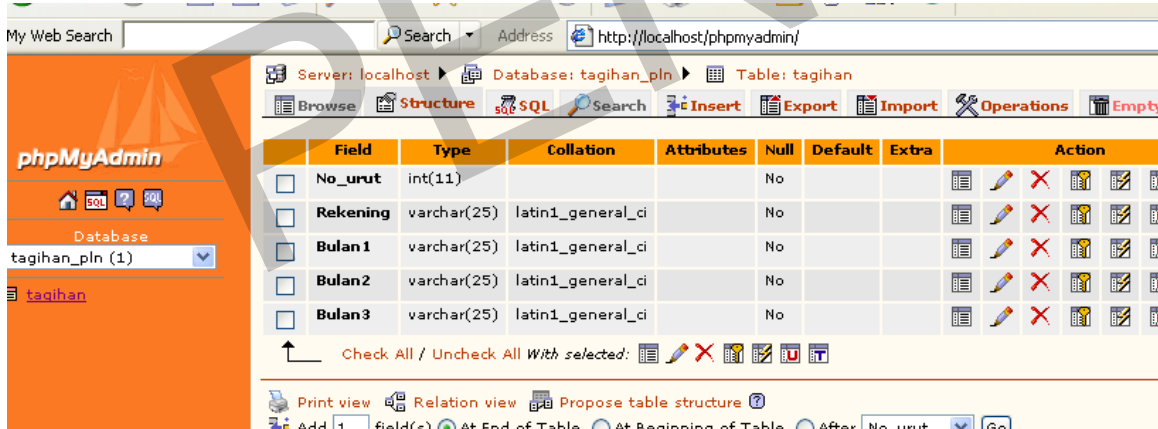
Gambar 3.5. Meng-create tabel baru dengan field berjumlah 5 buah

4. Mengisi nama field dan menentukan tipe masing-masing field dan jumlah digit dalam 1 field (jika diperlukan). Isi sesuai yang telah disebutkan sebelumnya.



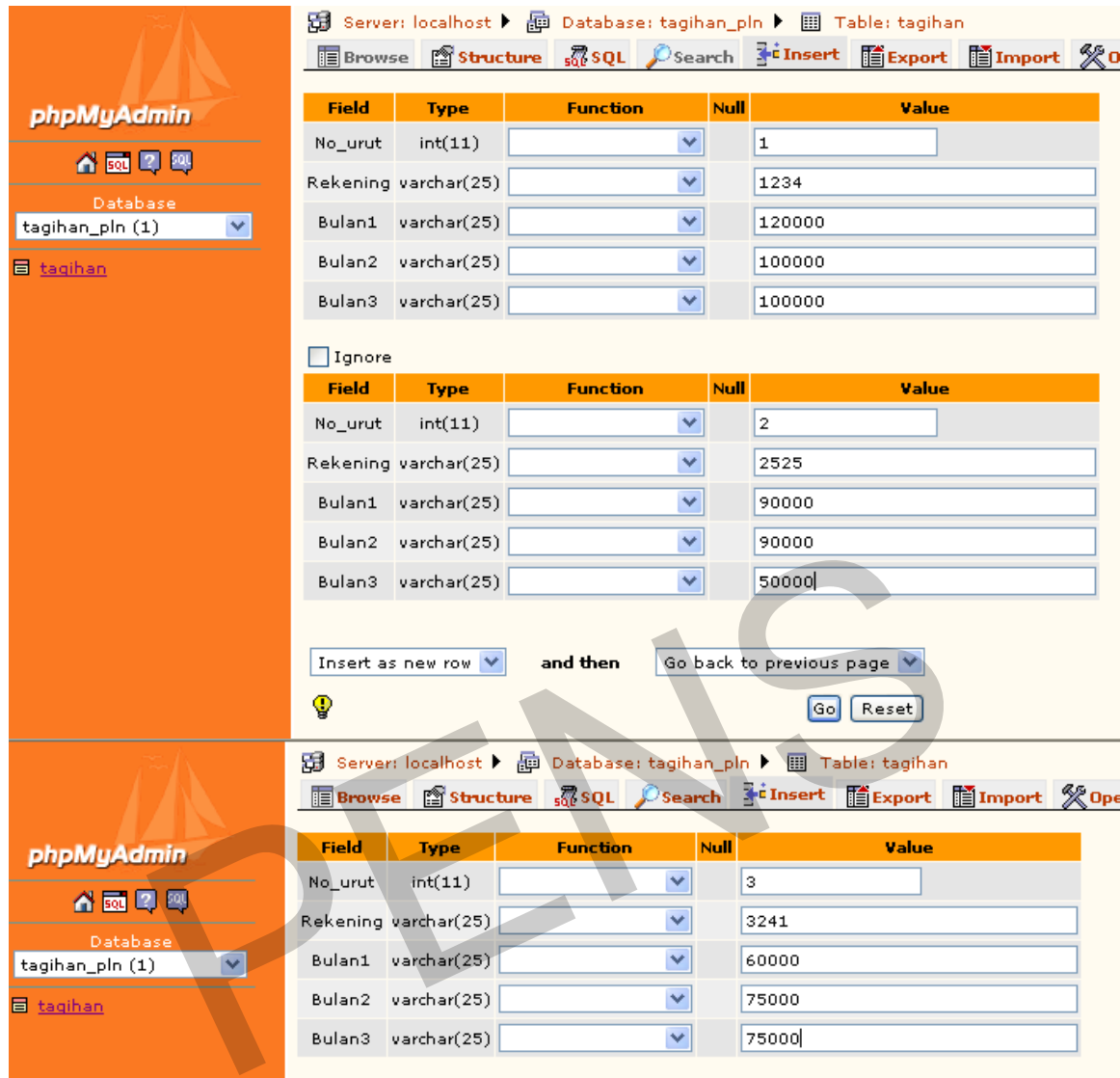
Gambar 3.6. Mengisi field dan tipe-nya

Setelah Tabel selesai dibuat, klik **Save**. Jika syntax penulisan benar, akan ditampilkan seperti pada gambar 3.10.



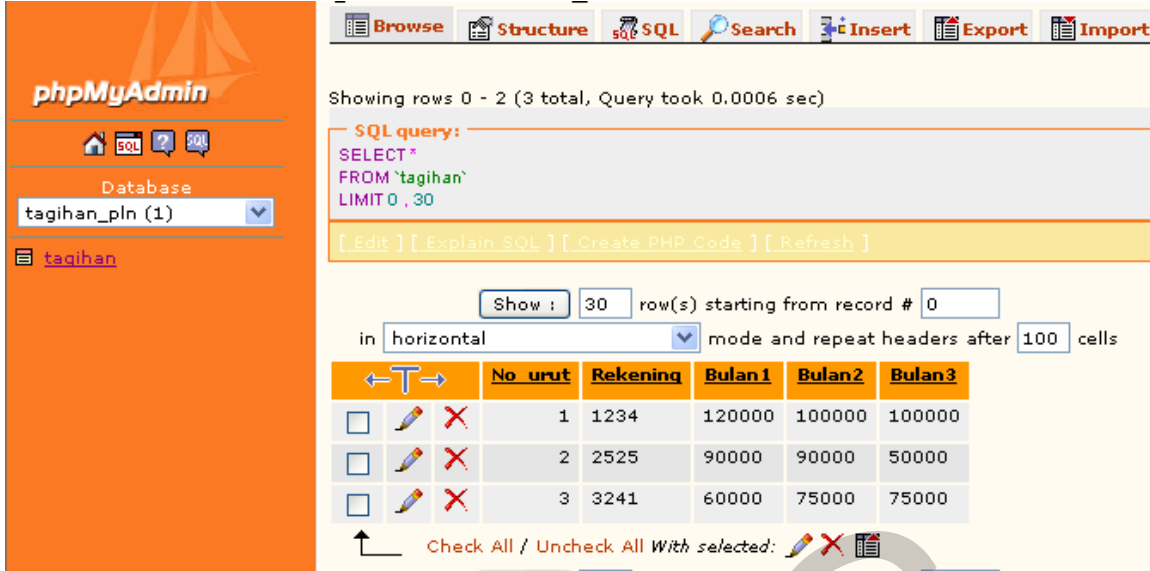
Gambar 3.10. Tampilan field dan tipe pada Tabel *tagihan*

5. Klik **Insert** di menu atas. Isilah masing-masing field tersebut dengan nilai seperti ditunjukkan pada gambar 3.11. klik **Go** untuk menyimpan hasil.



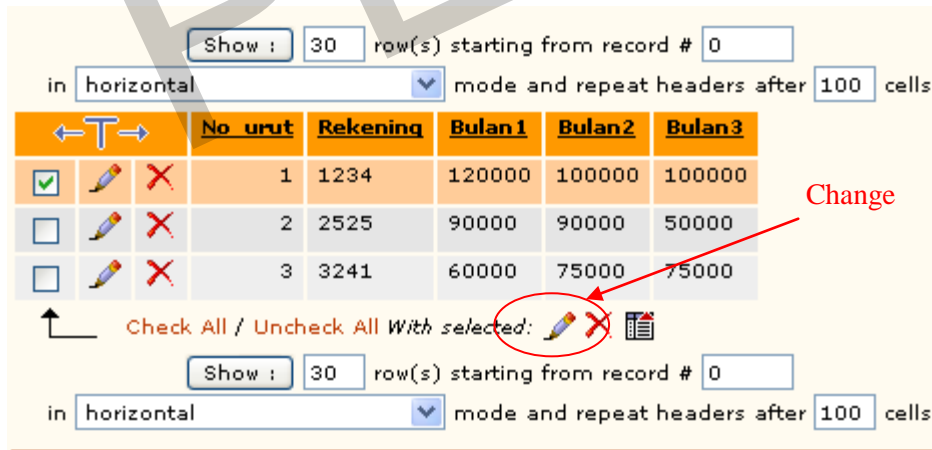
Gambar 3.11. Pengisian field dengan value yang telah ditentukan

Untuk mengecek apakah semua field sudah terisi dengan benar, klik **Browse** pada menu atas. Hasilnya akan nampak seperti pada gambar 3.12.



Gambar 3.12. Tabel yang sudah terisi value

6. Jika terjadi kesalahan saat pengisian, dapat dilakukan perbaikan (update) dengan cara memilih baris yang akan di-update, kemudian klik gambar pensil (change). Ini adalah update dengan cara manual. Update dengan programming lebih disarankan, bisa melalui pemrograman mysql atau melalui C++.



Gambar 3.13. Update data pada Tabel Tagihan



## 2. Meyiapkan file rekaman .wav

File-file wav yang diperlukan untuk dibunyikan selama proses pengaksesan perlu direkam lebih dulu. Gunakan *Multi Thread program*, dengan format perekaman **Linier PCM 8 Khz**, untuk merekam file-file seperti yang tertulis pada Tabel 3.1. Jangan lupa untuk mengaktifkan *Dialogic Configuration Manager (DCM)* ! Letakkan file-file tersebut pada sebuah folder tertentu.

Tabel 3.1. File-file wav untuk perekaman

Nama file	Kalimat
pembuka.wav	Selamat datang di Layanan informasi Tagihan PLN wilayah Surabaya Selatan
input.wav	Masukkan empat digit nomor rekening anda
konfirmasi.wav	Nomor rekening anda adalah...
benar.wav	Jika Benar, tekan 1. Jika Salah, tekan 2
pilihan.wav	Untuk pilihan bulan kesatu, tekan 1. Untuk pilihan bulan kedua, tekan 2. Untuk pilihan bulan ketiga, tekan3. Untuk kembali ke menu pilihan, tekan #. Untuk keluar, tekan *.
satu.wav	satu
dua.wav	dua
tiga.wav	tiga
empat.wav	empat
lima.wav	lima
limapuluhribu.wav	limapuluh ribu
enampuluhribu.wav	enampuluh ribu
tujuhpuhlimaribu.wav	tujuhpuh lima ribu
sembilanpuluhribu.wav	sembilan puluh ribu
seratusribu.wav	seratus ribu
seratusduapuluhribu.wav	seratus dua puluh ribu
rupiah.wav	rupiah
Tagihan bulan ini.wav	Tagihan anda bulan ini adalah..
penutup.wav	Terima kasih telah menggunakan layanan ini. Untuk pengaduan silakan hubungi lima sembilan empat dua dua tiga satu

### 3. Bekerja dengan Visual C++

a. Buat Proyek baru → *File* → *New* → *Project* → pilih **Win32 Console Application**, beri nama. Lalu buat file .cpp baru dengan memilih *File* → *New* → **C++ Source File**, letakkan pada project yang sudah dibuat sebelumnya → beri nama → OK

b. pilih *project* → *setting*

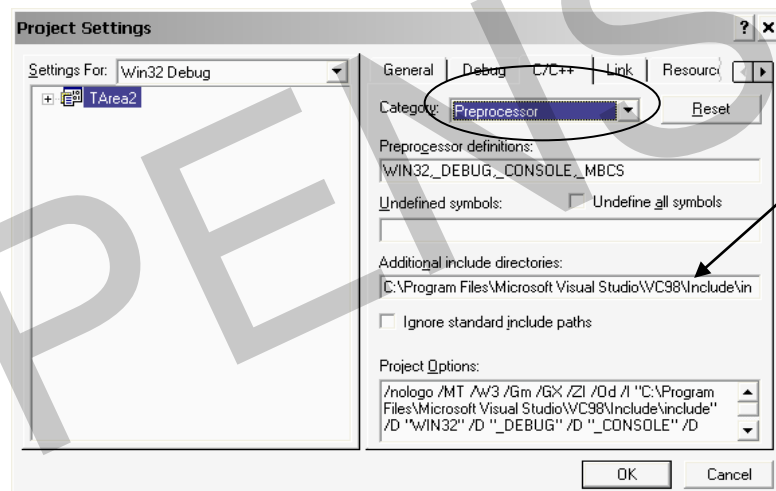
*Tab: C/C++*

- [*Category: Preprocessor*]

*Additional include Directories*

<cari letak folder Include MySQL pada folder Include C++>

Hasil pengesetannya seperti pada gambar 3.14.



Gambar 3.14. Setting Category *Preprocessor*

Pastikan bahwa folder include dari MySQL sudah di-copy kan ke dalam folder include VC98 milik Visual C++ yang ada di Directory Program Files.

- [*Category: Code Generation*]

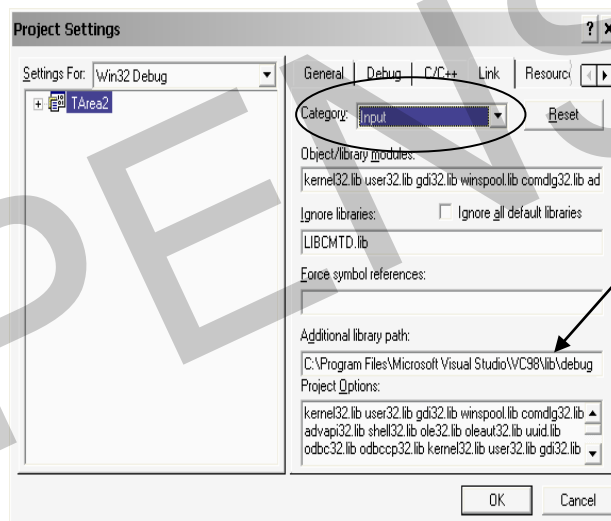
*Use run-time library:*

Pilih *Multithreaded*

- [ *Category: Precompiled Headers* ]  
Pilih 'Not using precompiled headers'

Tab: **Link**

- [ *Category: Input* ]  
Object/library modules: (tambahkan)  
**wsock32.lib mysqlclient.lib libmysql.lib mysys.lib**  
Ignore libraries:  
(tuliskan) **LIBCMTD.lib**  
Additional library path:  
<cari letak folder lib MySQL pada folder lib C++>  
Hasil pengesetannya seperti pada gambar 3.15



Gambar 3.15. Setting Category Input

Pastikan bahwa folder lib dari MySQL sudah di-copy kan ke dalam folder lib VC98 milik Visual C++ yang ada di Directory Program Files.

- [ *Category: General* ]  
Object/library modules: (tambahkan)  
**libsrmt.lib libdxxmt.lib**

c. Menambahkan *Directory* INC dan LIB

Masih pada *sheet* Project → *Tools* → *Options* → *Directories* → *Show Directories for* : pilih *Include file* → *browse folder* dimana *Directory* INC untuk Dialogic berada, sehingga didapatkan : C:\Program Files\Dialogic\INC.

*Show Directories for* : pilih *Library file* → *browse folder* dimana *Directory* LIB untuk Dialogic berada, sehingga didapatkan : C:\Program Files\Dialogic\LIB → **OK**.

d. Menambah *header-header*

Pada Project yang masih kosong, tambahkan header-header di bawah ini.

```
#include <windows.h>
#include <fcntl.h>
#include <srllib.h>
#include <dxxplib.h>
#include <stdio.h>
```

e. Membuat fungsi **Deteksidigit()** dan **PlaySuara()**.

Lakukan seperti pada pemrograman sebelumnya.

f. Menambahkan *prototype* fungsi Deteksidigit() dan PlaySuara().

```
char DeteksiDigit(int chdev, char digit[10], int x);
void PlaySuara(int chdev, char fname[10]);
```

g. Menyiapkan koneksi dengan database

Untuk keperluan koneksi database, ada beberapa penambahan di dalam penulisan program. Penambahan itu meliputi penambahan header file, penambahan fungsi yang berhubungan dengan database dan pendefinisian lokasi database oleh C++.

(i) Menyiapkan Header file yang berhubungan dengan database MySQL

```
#include<stdio.h>
#define W32_LEAN_AND_MEAN
#include<winsock2.h>
#include"mysql.h"
```

(ii) Mendefinisikan lokasi Tabel dalam Database

```
#define TABLE_OF_INTEREST "tagihan"
#define SERVER_NAME "localhost"
```

```
#define DB_USER "root"  
#define DB_USERPASS ""  
#define DB_NAME "tagihan_pln"
```

(iii) Mendeklarasikan prototype fungsi yang berhubungan dengan pemrosesan database

(iv) Menuju ke lokasi database

Perintah-perintah untuk menuju ke lokasi database dibuat di C++ dengan tujuan agar program yang ada di C++ dapat mengenali lokasi dari database, dimana terdapat table yang memuat nilai-nilai yang akan diproses di C++. Nama database, nama table dan nama server sudah didefinisikan di #define.

#### *h. Main Program*

Langkah-langkah yang telah dijelaskan di atas, akan dibuat menjadi program lengkap. Ada beberapa proses yang harus dijadikan fungsi tersendiri, seperti proses pendeteksian digit (fungsi *Deteksidigit()*), proses membunyikan suara hasil rekaman (fungsi *PlaySuara()*), proses menampilkan Tabel (fungsi *showTables()*), proses menyeleksi isi table (fungsi *showContents()*), dan proses koneksi database (fungsi *koneksi()*).

Untuk memahami alur pemrograman, usahakan mengetik sendiri main program. Untuk isi fungsi *DeteksiDigit* dan *PlaySuara* boleh dilakukan peng-copy an guna menghindari kesalahan *syntax*. Setelah seluruh program selesai dibuat, lakukan *compiling* dan perbaiki error yang terjadi.

```
#define W32_LEAN_AND_MEAN  
#include <winsock2.h>  
#include "mysql.h";  
#include <windows.h>  
#include <fcntl.h>  
#include <srllib.h>  
#include <dxxplib.h>  
#include <stdio.h>  
#include <string.h>
```

```

// definisi koneksi ke database
#define TABLE_OF_INTEREST "tagihan"
#define SERVER_NAME "localhost"
#define DB_USER "root"
#define DB_USERPASS ""
#define DB_NAME "tagihan_pln"

//prototype
char DeteksiDigit(int chdev,char digit[10],int x);
void PlaySuara(int chdev,char fname[10]);
void menu_pilihan(int chdev);
void koneksi_databases1(int chdev,MYSQL *handle,const char
*tbl,int nomor,int y);
void nilai(int chdev,char nm[5]);
void mulai(int chdev);

//Fungsi Deteksi Digit
char DeteksiDigit(int chdev,char digit[10],int x)
{
    DV_TPT tpt[3];
    DV_DIGIT digp;
    int numdigs,cnt;

    dx_clrtp(tpt,3);
    tpt[0].tp_type=IO_CONT;
    tpt[0].tp_termno=DX_MAXDTMF;
    tpt[0].tp_length=x;
    tpt[0].tp_flags=TF_MAXDTMF;

    tpt[1].tp_type=IO_CONT;
    tpt[1].tp_termno=DX_LCOFF;
    tpt[1].tp_length=10;
    tpt[1].tp_flags=TF_LCOFF|TF_10MS;

    tpt[2].tp_type=IO_EOT;
    tpt[2].tp_termno=DX_MAXTIME;
    tpt[2].tp_length=50;
    tpt[2].tp_flags=TF_MAXTIME;

    //Get digit//
    if((numdigs=dx_getdig(chdev,tpt,&digp,EV_SYNC))===-1){
        printf ("Error get digit");
        exit(1);
    }

    for(cnt=0;cnt<numdigs;cnt++){
        digit[cnt]=digp.dg_value[cnt];
    }
    return(digp.dg_value[0]);
}

```

```

}

//Fungsi Memainkan Suara
void PlaySuara(int chdev,char fname[10])
{
    int fd;
    DX_IOTT iott;
    DV_TPT tpt;
    DX_XPB xpb;

    if((fd=dx_fileopen(fname,O_RDONLY|O_BINARY)) ==-1) {
    }

    tpt.tp_type           =IO_EOT;
    tpt.tp_termno        =DX_MAXDTMF;
    tpt.tp_length        =1;
    tpt.tp_flags         =TF_MAXDTMF;

    iott.io_fhandle      =fd;
    iott.io_bufp         =0;
    iott.io_offset       =0;
    iott.io_length       =-1;
    iott.io_type         =IO_DEV|IO_EOT;

    xpb.wFileFormat      =FILE_FORMAT_WAVE;
    xpb.wDataFormat      =DATA_FORMAT_DIALOGIC_ADPCM;
    xpb.nSamplesPerSec   =DRT_8KHZ;
    xpb.wBitsPerSample   =4;

    if(dx_playiottdata(chdev,&iott,&tpt,&xpb,EV_SYNC)==
    -1){
        printf("Error play wav file");
        exit(1);
    }
}

//Fungsi memilih menu
void menu_pilihan(int chdev)
{
    char number[4],dig[10],angka[4] ;
    int nomor,i,pil;
    MYSQL *hnd=NULL; // mysql connection handle
    const char *sinf=NULL; // mysql server information
    hnd = mysql_init(NULL);

    salah:
        PlaySuara(chdev,"input.wav");
        number[0]=DeteksiDigit(chdev,dig,1);
        number[1]=DeteksiDigit(chdev,dig,1);

```

```

        number[2]=DeteksiDigit(chdev,dig,1);
        number[3]=DeteksiDigit(chdev,dig,1);
        number[4]='\0';

        nomor = atoi(number);

        PlaySuara(chdev,"konfirmasi.wav");

        for(i=0;i<4;i++)
        {
            if(number[i]== '1')
                PlaySuara(chdev,"satu.wav");
            else if(number[i]== '2')
                PlaySuara(chdev,"dua.wav");
            else if(number[i]== '3')
                PlaySuara(chdev,"tiga.wav");
            else if(number[i]== '4')
                PlaySuara(chdev,"empat.wav");
            else if(number[i]== '4')
                PlaySuara(chdev,"empat.wav");
            else if(number[i]== '5')
                PlaySuara(chdev,"lima.wav");
        }

        PlaySuara(chdev,"benar.wav");
        number[0]=DeteksiDigit(chdev,dig,1);
        number[1]='\0';
        if(number[0]=='2')
            goto salah;
        else if(number[0]=='1')
        {
pilihan:
            PlaySuara(chdev,"pilihan.wav");
            angka[0]=DeteksiDigit(chdev,dig,1);
            angka[1]='\0';
            pil = atoi(angka);

            if((angka[0]=='1')||(angka[0]=='2')||(angka[0]=='3'))
            {
                PlaySuara(chdev,"Tagihan bulan ini.wav");

            if(NULL==mysql_real_connect(hnd,SERVER_NAME,DB_USER,DB_USERPASS,DB
            B_NAME,0,NULL,0))
            {
                fprintf(stderr,"Terdapat masalah pada %s database
            user %s.\n",DB_NAME,SERVER_NAME);
            }
            else

```



```

        {
            fprintf(stdout,"Koneksi ke %s database pada %s dengan user
'%s'.\n",DB_NAME,SERVER_NAME,DB_USER);
            sinf = mysql_get_server_info(hnd);

            if (sinf != NULL)
            {

                fprintf(stdout,"Got server information: '%s'\n",sinf);

                koneksi_databasel(chdev,hnd,TABLE_OF_INTEREST,nomor,pil);

                goto pilihan;

            }
            else
            {

                fprintf(stderr,"Failed to retrieve the server information
string.\n");
            }

            mysql_close(hnd);
        }

    }
    else if(angka[0]=='#')
        goto pilihan;
    else if(angka[0]=='*')
        PlaySuara(chdev,"penutup.wav");
    }

    mulai(chdev);
}

//Fungsi terhubung ke database
void koneksi_databasel(int chdev,MYSQL *handle,const char
*tbl,int nomor,int y)
{
    MYSQL_RES *res=NULL; // result of querying for all rows in
                        table
    MYSQL_ROW row; // one row returned
    char sql[1024]; // sql statement used to get all rows
    int i; // number of fields returned from the query
    char abl[5],abn[5],abc[5];

    sprintf(sql,"select bulan1,bulan2,bulan3 from tagihan where
rekening=%d",nomor);

```

```

printf("nomor rekeningnya adalah %d", nomor);
if (!mysql_query(handle, sql))
{
    res = mysql_use_result(handle);
    if (res)
    {
        while (row=mysql_fetch_row(res))
        {
            for (i=0;i<mysql_num_fields(res);i++)
            {
                strcpy(abl,row[0]);

                strcpy(abn,row[1]);

                strcpy(abc,row[2]);

            }
        }
    }
    else
    {
        fprintf(stderr,"Failed to use the result
acquired!\n");
    }
}
else
{
    fprintf(stderr,"Failed to execute query. Ensure table
is valid!\n");
}
if(y==1)
    nilai(chdev,abl);
else if(y==2)
    nilai(chdev,abn);
else if(y==3)
    nilai(chdev,abc);
}

//Fungsi Membunyikan nilai yang didapat dari database
void nilai(int chdev,char nm[5])
{
    int nn;

    nn=atoi(nm);

    if(nn == 50000)
    {

        PlaySuara(chdev,"limapuluhribu.wav");
    }
}

```

```

    PlaySuara(chdev, "rupiah.wav");
}
else if(nn == 60000)
{
    PlaySuara(chdev, "enam puluhribu.wav");
    PlaySuara(chdev, "rupiah.wav");
}
else if(nn == 75000)
{
    PlaySuara(chdev, "tujuh puluhlimaribu.wav");
    PlaySuara(chdev, "rupiah.wav");
}
else if(nn == 90000)
{
    PlaySuara(chdev, "sembilan puluhribu.wav");
    PlaySuara(chdev, "rupiah.wav");
}
else if(nn == 100000)
{
    PlaySuara(chdev, "seratusribu.wav");
    PlaySuara(chdev, "rupiah.wav");
}
else if(nn == 120000)
{
    printf("Nilai sekarang untuk adalah %d", nn);
    PlaySuara(chdev, "seratusduapuluhribu.wav");
    PlaySuara(chdev, "rupiah.wav");
}
}

//Fungsi memulai sistim layanan
void mulai(int chdev)
{
    //Open channel//
    if((chdev = dx_open("dxxxxB1C1", NULL)) == -1)
    {
        printf("Error open channel");
        exit(1);
    }
    printf("Open channel success\n");

    //Set on hook//
    if(dx_sethook(chdev, DX_ONHOOK, EV_SYNC) == -1)
    {
        printf("Error on hook");
        exit(1);
    }
    printf("On hook success\n");
}

```

```

//Wait ring tone//
if(dx_wtring(chdev,2,DX_OFFHOOK,-1)==-1)
{
    printf("Error off hook");
    exit(1);
}
printf("Off hook success\n");

    PlaySuara(chdev,"pembuka.wav");
    menu_pilihan(chdev);
}

void main(int argc, char* argv[])
{
    int chdev;

    mulai(chdev);
}

```

### 3.5. Analisa

1. Apa yang terjadi jika Nomer rekening yang anda masukkan tidak terdaftar pada Tabel Tagihan ?
2. Apa yang terjadi jika jumlah digit untuk nomer rekening yang ditekan lebih dari 4 digit ?
3. Apa yang terjadi jika fungsi mulai() yang terletak di akhir fungsi menu\_pilihan() tidak diaktifkan ?

### 3.6. Pertanyaan & Tugas :

Buat program *Database Inventory* sebuah toko berbasis IVR. Jenis toko terserah anda (usahakan hanya satu jenis saja supaya tidak terlalu banyak record yang diisikan, misalkan : toko roti, toko tanaman, toko obat dll). Disainlah variable yang diperlukan untuk pembuatan database. Buat flow chart pengaksesan dan lengkapi dengan *breakdown* menu pilihan.