

Percobaan 5

Pemrograman IVR dengan Database(Outbound)

5.1. Tujuan :

Setelah melaksanakan praktikum ini mahasiswa diharapkan mampu :

- Membuat Database Layanan IVR dengan MySQL
- Membuat koneksi antara Database dengan C++
- Membuat sebuah aplikasi IVR outbound yang terkoneksi dengan database

5.2. Peralatan :

- PABX
- 2 pesawat Telepon
- 1 PC dilengkapi dengan Dialogic Card (D/41JCT-LS atau D/4PCI-U)

5.3. Teori :

5.3.1 Aplikasi Database

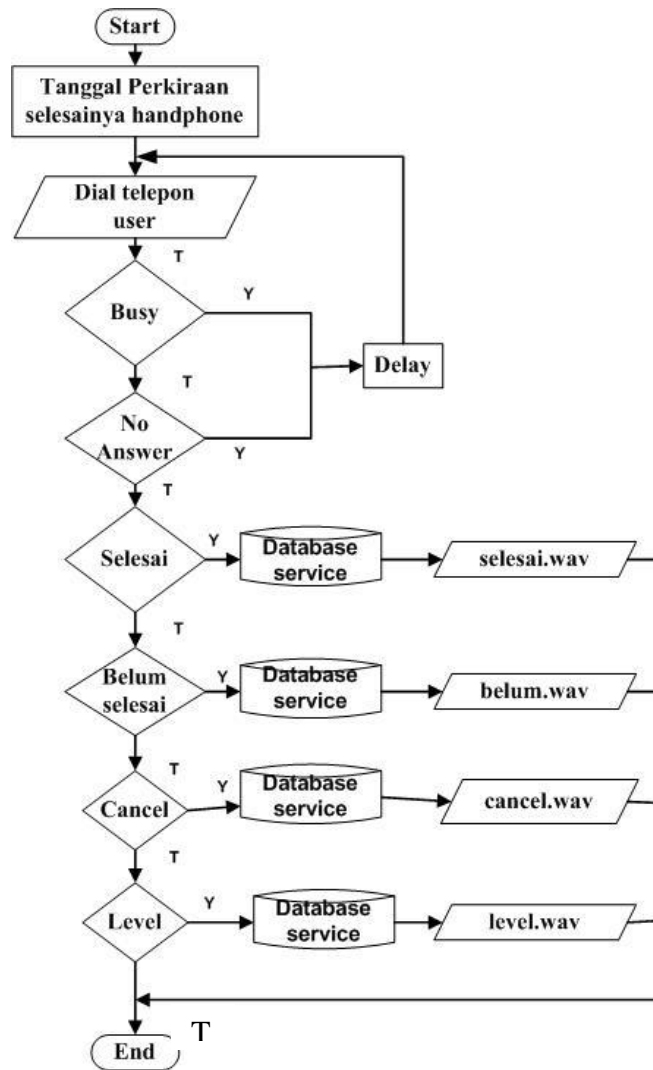
Aplikasi data base diperlukan jika Layanan sistim informasi menggunakan jenis data inventory, yaitu data yang memerlukan inventarisasi, seperti data Pelanggan, data Jenis Barang, data Nilai Mahasiswa dan sebagainya. Data base yang digunakan dalam sistim Layanan ini adalah database MySQL yang terintegrasi pada program XAMPP. XAMPP ini mengintegrasikan Apache, PHP serta MySQL dalam satu bundel sehingga sangat efektif dan mudah dalam proses penginstalannya. Tidak menutup kemungkinan menggunakan jenis data base yang lain, selama data base tersebut dapat di-akses dari program C yang disediakan oleh Dialogic.

5.3.2 Aplikasi Layanan Informasi Outbound

Aplikasi ini merupakan lanjutan dari aplikasi IVR inbound yang telah dilakukan pada praktikum sebelumnya. Dimana pada praktikum kali ini akan dibuat sebuah sistem remainder untuk mengingatkan customer akan status handphone yang sedang di service.

Layanan remainder ini diberikan berdasarkan dari tanggal yang sudah tersimpan di database .

Diagram Alir cara pengaksesan Sistem remainder ditunjukkan pada Gambar 1



Gambar 1. Diagram Alir cara pengaksesan Sistem Remainder Care Centre yang Terkoneksi dengan Database

Jenis file wav yang akan dibunyikan adalah seperti pada Tabel 1

Tabel 1. Tabel Isi file wav

| Nama File Wav | Kalimat |
|-------------------|---|
| cancel.wav | Dicancel |
| selesai.wav | Sudah selesai |
| belum.wav | belum selesai |
| level.wav | dikirim kelevel tiga |
| status.wav | Status handphone anda saat ini adalah |
| selamatdatang.wav | Selamat datang di layanan remainder care centre |
| keluar.wav | Terima kasih atas kepercayaan anda pada care centre kami. |

5.4. Prosedur Percobaan :

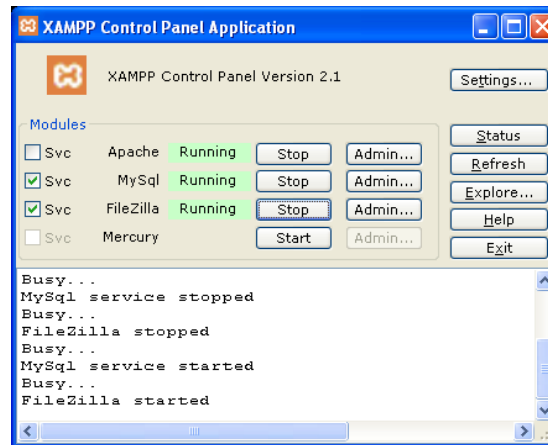
1. Aktifkan DCM(*Dialogic Configuration Manager*) sehingga dialogic dalam posisi running. Siapkan file-file wav yang akan digunakan, dengan cara merekam melalui SAMPLE PROGRAM . Isi dari file wav seperti pada Tabel 1
2. Menyiapkan aplikasi database. Dimana field-field yang disiapkan untuk database adalah :
 1. id_remainder (int)
 2. nama (varchar(20))
 3. alamat (varchar(20))
 4. status_handphone(int)
 5. tgl_selesai(Date and Time)

Tabel 2. Field-field yang disiapkan untuk database

| id_remainder | nama | alamat | StatusHandphone | Tanggal_selesai |
|--------------|------|------------|-----------------|-----------------------|
| 1 | Ali | Jl.Mawar | 1 | 1-april-2010 09:00:00 |
| 2 | Anis | Jl.Melati | 2 | 1-april-2010 09:05:00 |
| 3 | Agri | Jl.Anggrek | 3 | 1-april-2010 09:10:00 |

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

- a. Menjalankan XAMPP
- b. Double klik “*xampp-control panel*” untuk memastikan bahwa Apache dan MySQL sudah berjalan, dan akan muncul tampilan seperti gambar 2



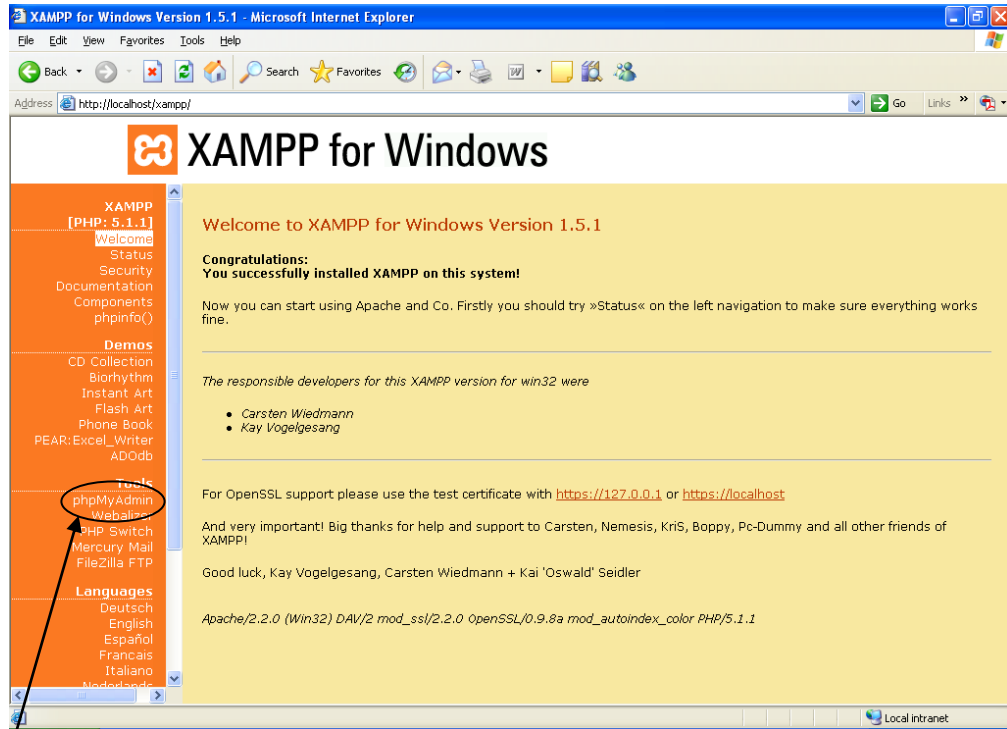
Gambar 2. Tampilan XAMPP Control Panel Application

- c. Mengaktifkan PHP MyAdmin

Dalam pembuatan database pada MySQL ada beberapa cara yang dapat digunakan yaitu : dengan menggunakan Phpmyadmin, menggunakan MySQL *Prompt* atau dengan membuat sebuah program. Untuk memudahkan pembuatan database pada MySQL dalam praktikum ini digunakan Phpmyadmin karena dalam pengkonfigurasianya tidak diperlukan perintah-perintah tertentu seperti jika kita menggunakan MySQL *Prompt*. Sehingga dengan menggunakan Phpmyadmin konfigurasi MySQL dapat dilakukan dengan mudah dan cepat.

Adapun langkah-langkah yang dilakukan dalam pembuatan database menggunakan Phpmyadmin adalah sebagai berikut :

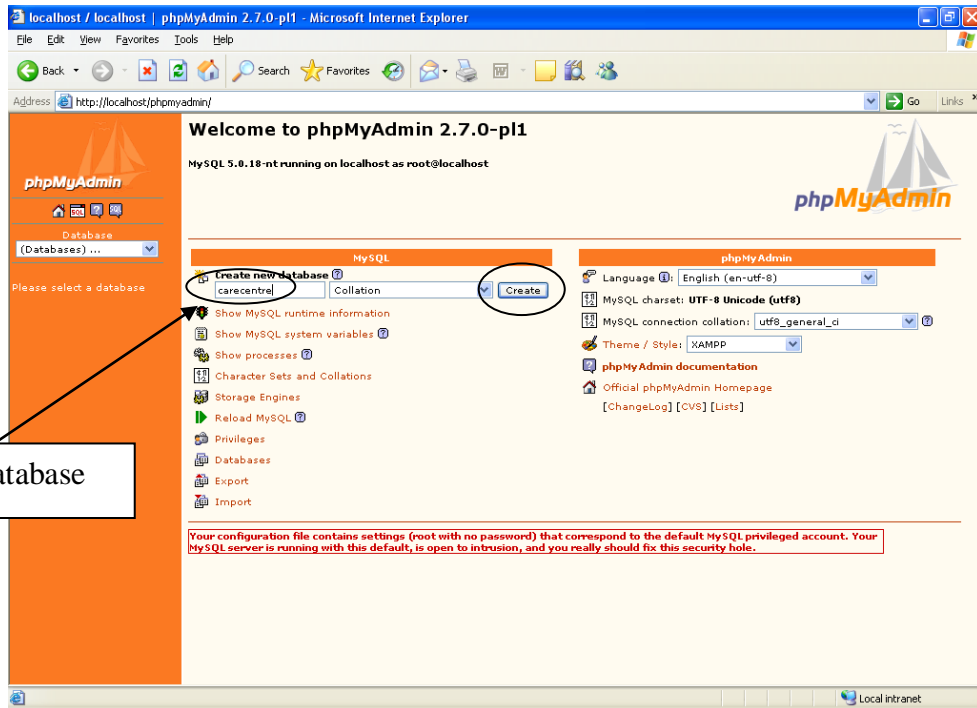
- Double klik “*Internet Explorer*” ketikkan <http://localhost/> pada *address*-nya, untuk menampilkan tampilan awal xampp. Seperti gambar 3



Gambar 3. Tampilan awal XAMPP

Klik bagian ini

- klik "phpmyadmin" dan akan nampak tampilan seperti gambar 4.

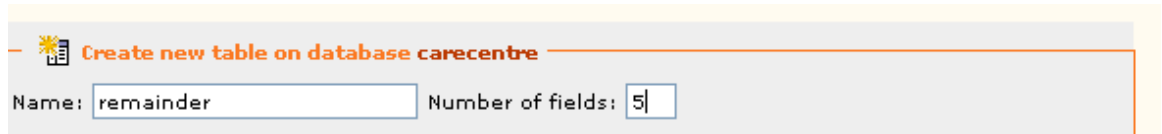


Nama database

Gambar 4. Pemberian Nama Database

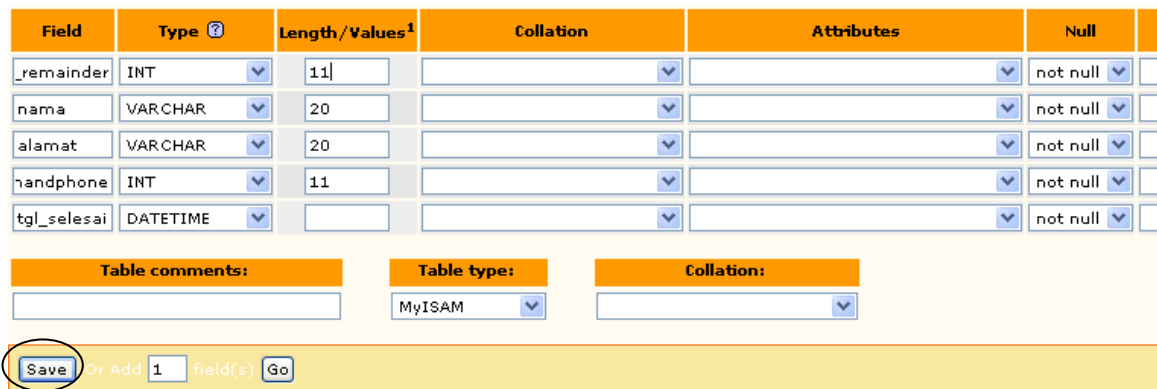
Kemudian beri nama database yang akan dibuat setelah itu pilih create

- Beri nama tabel yang akan dibuat, tentukan jumlah field-nya kemudian klik go



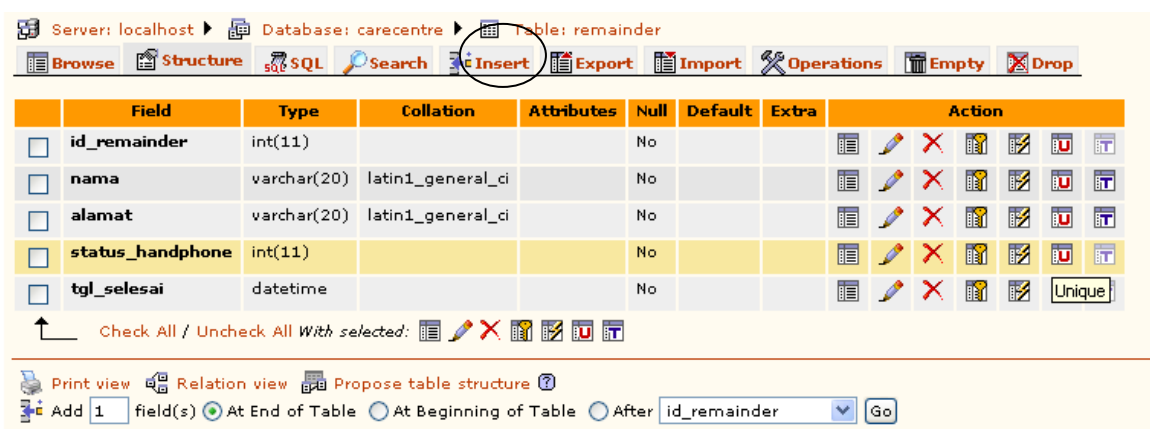
Gambar 5. Pemberian Nama Tabel

- Siapkan Tabel menggunakan MySQL seperti pada Gambar 6 dan 7. Nama: remainder



Gambar 6. Pembuatan Field-field

klik save, sehingga didapat tampilan seperti dibawah ini kemudian klik insert



Gambar 7. Tampilan Setelah Pembuatan field

- Isilah masing-masing field dengan record seperti ditunjukkan pada Gambar 8.

Ignore

| Field | Type | Function | Null | Value |
|------------------|-------------|----------|------|---------------------|
| id_remainder | int(11) | | | 1 |
| nama | varchar(20) | | | ali |
| alamat | varchar(20) | | | jl.mawar |
| status_handphone | int(11) | | | 1 |
| tgl_selesai | datetime | | | 2010-04-01 09:00:00 |

| Field | Type | Function | Null | Value |
|------------------|-------------|----------|------|---------------------|
| id_remainder | int(11) | | | 2 |
| nama | varchar(20) | | | anis |
| alamat | varchar(20) | | | jl.melati |
| status_handphone | int(11) | | | 2 |
| tgl_selesai | datetime | | | 2010-04-01 09:05:00 |

Insert as new row Go back to previous page

Klik bagian ini

Calendar - Microsoft Internet Explorer

| « Apr » | | « 2010 » | | | | |
|---------|-----|----------|-----|-----|-----|-----|
| Sun | Mon | Tue | Wed | Thu | Fri | Sat |
| | | | | 1 | 2 | 3 |
| 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 |
| 18 | 19 | 20 | 21 | 22 | 23 | 24 |
| 25 | 26 | 27 | 28 | 29 | 30 | |

09 : 00 : 00

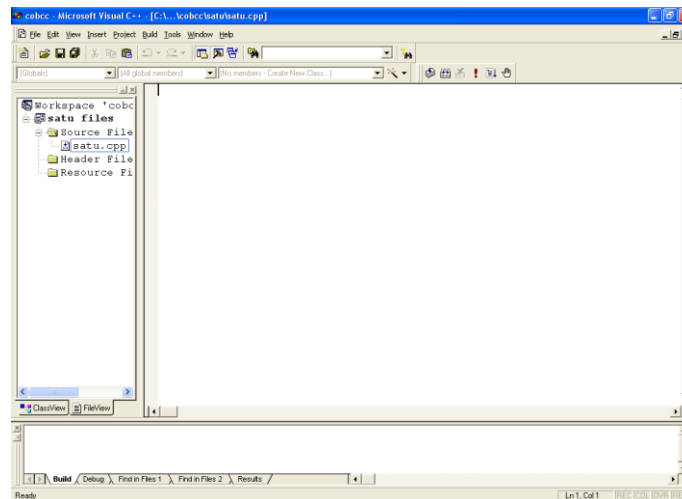
Gambar 8. Pengisian Record di masing-masing Field

Setelah record terisi, akan didapatkan sebuah tabel dengan data-data sebagai berikut:

| | | | id_remainder | nama | alamat | status_handphone | tgl_selesai |
|--------------------------|--|--|--------------|------|------------|------------------|---------------------|
| <input type="checkbox"/> | | | 1 | ali | jl.mawar | 1 | 2010-04-01 13:45:00 |
| <input type="checkbox"/> | | | 2 | anis | jl.melati | 2 | 2010-04-01 09:05:00 |
| <input type="checkbox"/> | | | 3 | agri | jl.anggrek | 3 | 2010-04-01 09:10:00 |

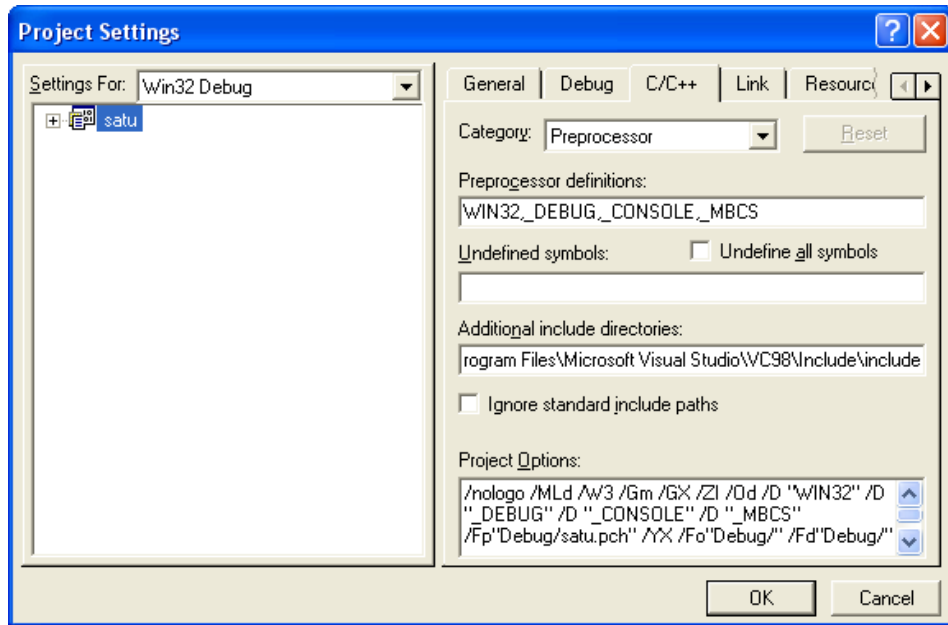
Gambar 9. Isi dari Tabel Remainder

3. Buka C++, buat Project baru untuk Sistim Layanan ini
 - a. Buka Visual C++ , kemudian buat workspace, project, dan file baru sehingga didapat tampilan seperti gambar 10



Gambar 10. Membuat workspace baru dengan Visual C++

- b. Setelah membuat project dan file *.cpp, pilih *project* → *setting*
 - Tab: C/C++
 - a. [Category Preprocessor]
 - Addition include directories
 - < path to your Mysql installation > \include
- Hasil pengesetannya seperti pada gambar 11



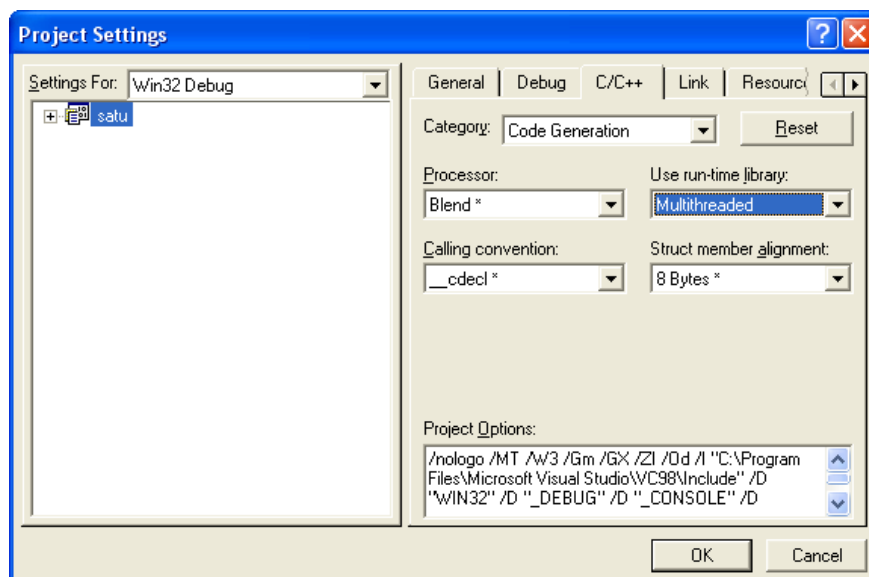
Gambar 11. Setting Category Preprocessor

b. [Category: Code Generation]

use run-time library:

Multithreaded

Hasil pengesetannya tampak seperti gambar 12.

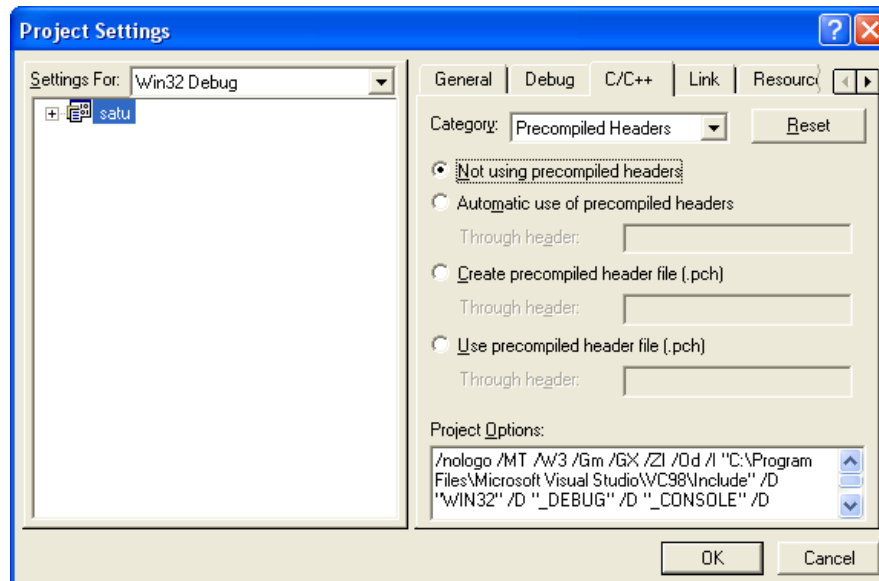


Gambar 12. Setting Category Code Generation

c. [Category:Precompiled Headers]

Select 'not using Precompiled Headers'

Hasil pengesetannya tampak pada gambar 13



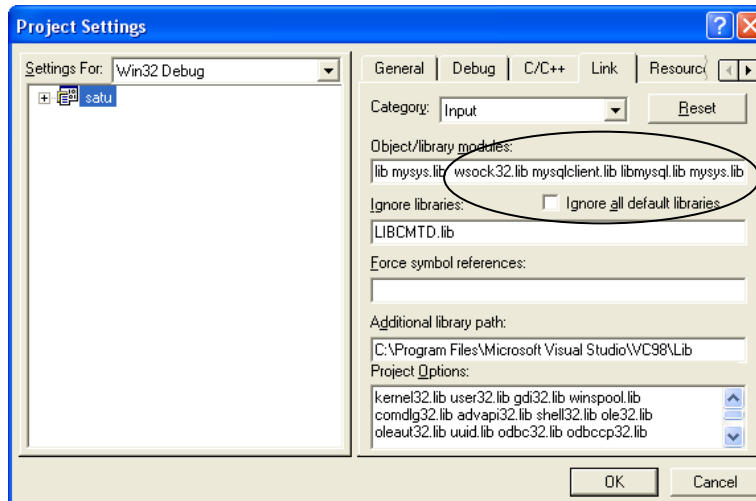
Gambar 13. Setting Category Precompiled Headers

▪ Tab:Link

a. [Category:Input]

- *Object/library modules: (tulis)*
wsock32.lib mysqlclient.lib libmysql.lib mysys.lib
- *Ignore libraries:*
LIBCMTD.lib
- *Additional library path:*
<path to your MySql installation>\lib\debug

Hasil pengesetannya seperti gambar 14

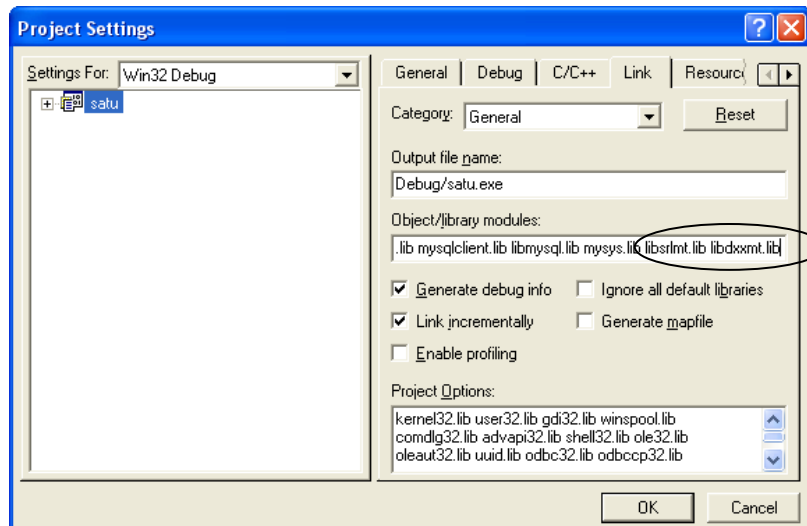


Gambar 14. Setting Category Input

b. [Category:General]

- o Object/library modules:(tulis)
libslmt.lib dan libdxxmt.lib

Hasil pengesetannya tampak seperti gambar 10

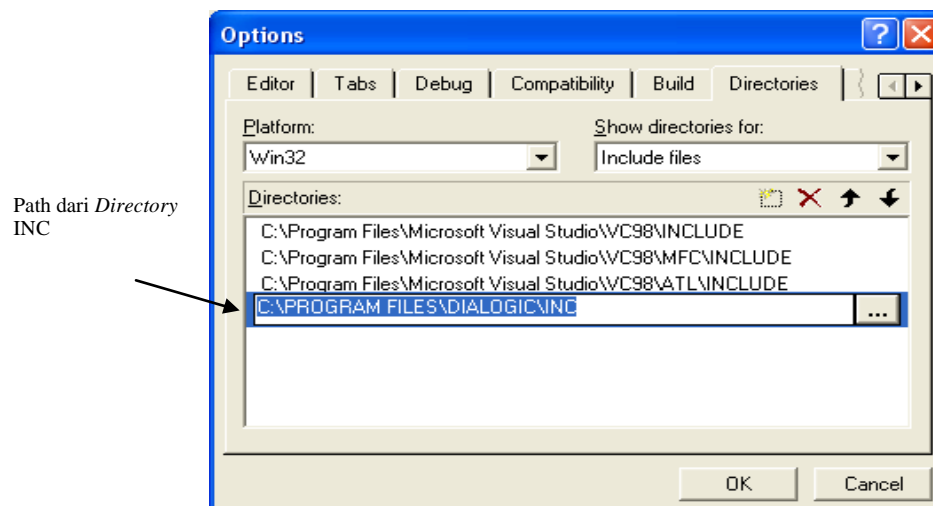


Gambar 15. Setting Category Input

- c. Menambahkan *Directory* INC dan LIB

Masih pada *sheet* Proyek → *Tools* → *Options* → *Directories* → *Show Directories for* : pilih *Include file* → *browse folder* dimana *Directory* INC untuk Dialogic berada, sehingga didapatkan : C:\Program Files\Dialogic\INC. Hal ini diperlihatkan seperti pada Gambar 11.

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



Gambar 16. Project Setting Box untuk Menambahkan File Library

- d. Menambah *header-header*. Harus diingat, penulisan *header-header* ini harus berurutan.

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

```
#include <time.h>
```

e. Tambahkan fungsi PlaySuara.

```
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\n");
        exit(1);
    }

    printf("play wav succes\n");
}
}
```

f. Tambahkan fungsi deteksi digit

```
char DetectDigit(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\n");
exit(1);
}
printf("get digit succes\n");
for(cnt=0;cnt<numdigs;cnt++)
{
digit[cnt]=digp.dg_value[cnt];
}
return(digp.dg_value[0]);
}

```

g. Buat main program seperti yang ditunjukkan pada listing program dibawah ini!

```

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

MYSQL *hnd=NULL; // mysql connection handle
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,j=0,numf=0; // number of fields returned from the query
int x,bil;
const char *sinf=NULL; // mysql server information
hnd = mysql_init(NULL);

if(NULL==mysql_real_connect(hnd,SERVER_NAME,DB_USER,DB_USERPASS,D
B_NAME,0,NULL,0))
{
fprintf(stderr,"ada masalah\n");
}

else
{
fprintf(stdout,"koneksi ok\n");
sinf = mysql_get_server_info(hnd);
if (sinf != NULL)
{

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

```

```

sprintf(sql,"select NOW()");

if (!mysql_query(hnd,sql))
{
    res = mysql_use_result(hnd);
    if (res)
    {
        while (row=mysql_fetch_row(res))
        {
            for (i=0;i<mysql_num_fields(res);i++)
            {
                strcpy(data,row[i]);
                printf("%s\n",data);
            }
        }
    }
    else
    {
        fprintf(stderr,"Failed to use the result
        acquired!\n");
    }
}

sprintf(sql,"select status_handphone from remainder where tgl_selesai =
'%s' ",data);
if (!mysql_query(hnd,sql))
{
    res = mysql_use_result(hnd);
    if (res)
    {
        while (row=mysql_fetch_row(res))
        {
            for (i=0;i<mysql_num_fields(res);i++)
            {
                strcpy(data1,row[i]);
                printf("%s\n",data1);
                bil=atoi(data1);
                printf("%d", bil);

                if((chdev = dx_open("dxxxB1C1",NULL))== -1)
                {
                    printf("Error open channel");
                    exit(1);
                }
                printf("open channel succes\n");

                awal:

                if(dx_sethook(chdev,DX_OFFHOOK,EV_SYNC)== -1)
                {
                    printf("Error off hook");
                    exit(1);
                }
            }
        }
    }
}

```

```

printf("off hook succes\n");

dx_clrcap(&capp);
capp.ca_nbrdna=4;

if((cares=dx_dial(chdev,dialstrg,&capp,DX_CALLP|EV_SYNC)
== -1 )
{
    printf("error dial");
    exit(0);
}

printf("proses dial");
switch(cares)
{
case CR_CNCT:
    printf("answer\n");

    PlaySuara(chdev,"selamatdatang.wav");
    PlaySuara(chdev,"status.wav");

    if(bil==1)

    {

        PlaySuara(chdev,"selesai.wav");

    }
    else if(bil==2)
    {

        PlaySuara(chdev,"belum.wav");

    }
    else if(bil==3)
    {

        PlaySuara(chdev,"level.wav");

    }
    else if(bil==4)
    {

        PlaySuara(chdev,"cancel.wav");

    }

    break;

case CR_BUSY:

    printf("Busy");
    if(dx_sethook(chdev,DX_ONHOOK,EV_SYNC)==-
1)
    {

```



```

        printf("Error on hook\n");
        exit(1);
    }
    printf("on hook succes\n");
    printf("\ntunggu 10 detik\n");
    Sleep(10000);
    goto awal;
    break;

case CR_NOANS:
    printf("no answer");
    if(dx_sethook(chdev,DX_ONHOOK,EV_SYNC)==-
    1)
    {
        printf("Error on hook\n");
        exit(1);
    }
    printf("on hook succes\n");
    printf("\ntunggu 10 detik\n");
    Sleep(10000);
    goto awal;
    break;

default :
    printf("else");
    printf("\ntunggu 10 detik\n");
    Sleep(10000);
    goto awal;
    break;
}

PlaySuara(chdev,"keluar.wav");

```

```

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

        printf("on hook succes\n");
    }
}
else
{
    fprintf(stderr,"Failed to use the result acquired!\n");
}

goto balik;

}

else
{
    fprintf(stderr,"Failed to execute query. Ensure table is
valid!\n");
}

}

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

}

mysql_free_result(res);
goto balik;
mysql_close(hnd);
}

```

g. Tambahkan prototype fungsi

```

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

```

i. Diantara Header dan prototype fungsi, tambahkan program dibawah ini!

```

#define TABLE_OF_INTEREST "remainder"
#define SERVER_NAME "localhost"
#define DB_USER "root"
#define DB_USERPASS ""
#define DB_NAME "carecentre"

```

```
char data[10];
char data1[10];
int chdev;
DX_CAP capp;
char dialstrg[5]="1000";
int cares;
```

- j. Jalankan program sampai tidak ada error yang didapat.
- k. Ikuti instruksi program remainder dari layanan care centre sehingga bisa didapat informasi yang diharapkan

Analisa:

1. Rubahlah tanggal dan status yang ada di database, apakah sistem remainder juga ikut berubah?
2. coba lakukan percobaan dibawah ini, untuk melihat apa yang terjadi pada IVR Server:
 - a. kondisi sibuk
 - b. kondisi tidak ada jawaban
 - c. Rubahlah nilai fungsi Sleep menjadi 5000, 60000, 100000!

5.5. Pertanyaan & Tugas :

Buatlah aplikasi program remainder yang terkoneksi ke database. Lengkapi dengan flow chart pengaksesan serta breakdown menu pilihan.