UNIVERSIDAD DE GUADALAJARA

Centro Universitario de Ciencias Exactas e Ingenierías

Diseño de Interfaces (D02)



Punto Extra 1: Base de Datos

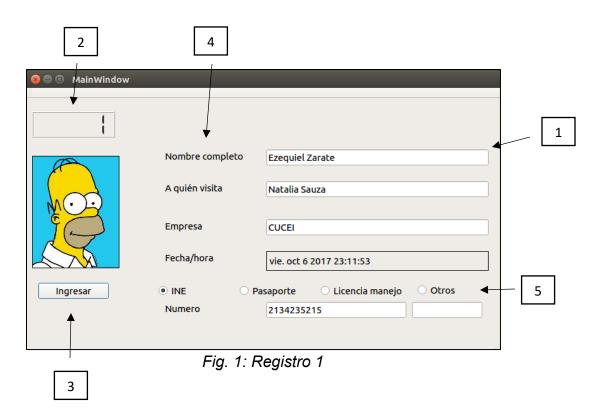
Alumno: Pérez Zárate Isaías Ezequiel **Código:** 209681558

Profesor: Rubén Estrada Marmolejo. **Fecha de entrega:** 06/10/2017

Resumen

En esta tarea se diseñará la interfaz de un registro de una base de datos en cual al momento de ejecutar nuestro programa automáticamente deberá mostrar el ultimo registro en el cual se quedó, para esto se tiene a continuación los elementos que se ocuparon para diseñar la interface.

- 5 Line Edit (1)
- 1 LCD Number (2)
- 1 Push Botton (3)
- 6 Label (4)
- 4 Radio Button (5)



En la figura 1 se observa el primer registro, al momento de darle click en ingresar automaticamente nos generara un archivo Excel que nos permite almacenar cada uno de los elementos que se obtienen al hacer el registro de cada uno de los datos, en la figura 2 se muestra de manera diferente, aquí únicamente muestra la interface al momento de abrirla de nuevo, en el cual se puede mostrar que el registro ya lleva un conteo de 10 personas



Fig. 1: Registro 2

Para hacerlo de manera más sencilla se llenó con usuarios en blanco y únicamente en la posición 10 se le ingresaron los datos correspondientes, en la figura 2 se muestran los resultados obtenidos de la base de datos capturada a través de la interface creada

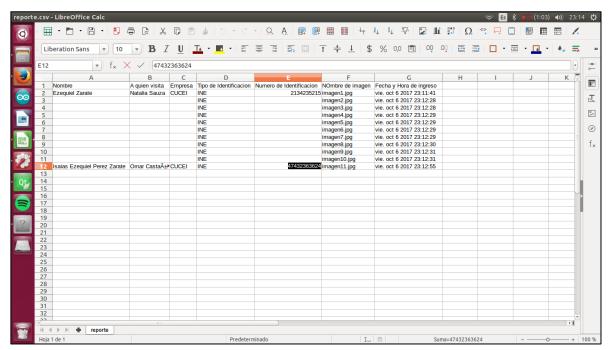


Fig. 1: Base de Datos

Código:

```
*mainwindow.cpp
#include "mainwindow.h"
#include "ui mainwindow.h"
#include "mat2qimage.h"
#include<opencv2/core/core.hpp>
#include<opencv2/ml/ml.hpp>
#include<opencv/cv.h>
#include<opencv2/imgproc/imgproc.hpp>
#include<opencv2/highgui/highgui.hpp>
#include<opencv2/video/background_segm.hpp>
#include<opencv2/videoio.hpp>
#include<opencv2/imgcodecs.hpp>
#include<qdebug.h>
#include<QTimer>
#include <QDateTime>
#include <QTime>
#include <QDate>
#include <QFile>
#include <QFileInfo>
//cv::VideoCapture camara(1);
using namespace cv;
QString nombre; //Guarda el cambio nombre asociado al lineEdit, campos que se guardan en el excel
QString empresa;
QString quien;
QString tiempoLocal;
QString tipoldentificacion;
QString numeroldentificacion;
QString nombreFoto;
int contador = 0;//Variable del LCD number el aumento
Mat IMAGEN;//Guarda la imagen de la camara Web
//cv::VideoCapture camara("http://192.168.43.243:8080/video");
//cv::VideoCapture camara("http://192.168.43.1:8080/video");
void MainWindow::fTimer(){
  ui->lcdNumber->display(contador);//Pone en el LCD number el numero que quedo guardado
anteriormente
   //Valida si existe el archivo de texto
   QFile conteo("contador.txt");
   QFileInfo existe("contador.txt");
   if(!existe.exists()){
```

```
conteo.open(QIODevice::ReadWrite);
      conteo.close();
   //Escribe el valor donde va el conteo
   if(existe.isFile() && existe.exists()){
      if(conteo.open(QIODevice::WriteOnly | QIODevice::Append)){
        QTextStream out(&conteo);
        conteo.resize(0);
        out << "<";
        out << contador;
        out << ">";
        conteo.close();
      }
   }
  //Actualizar la hora a la etiqueta correspondiente
  QDate fecha1 = QDate::currentDate();//Toma la fecha actual, no es string es una fecha
  QTime tiempo = QTime::currentTime();//Toma la hora actual
  QString fecha = fecha1.toString() + " " + tiempo.toString();//Convierte los valores a Qstring para que se
puedan mostrar
  tiempoLocal = fecha; //Campo que se guarda en el excel
  ui->label_6->setText(fecha);//Asigna a la label 6 la fecha actual
  Mat IMAGEN2;
 // if(camara.isOpened()){
  //camara >> IMAGEN;
  IMAGEN = cv::imread("../img/homer.jpg");
  cv::resize(IMAGEN,IMAGEN2,Size(138,177),0,0,0);
  QImage qImage = Mat2QImage(IMAGEN2);
  QPixmap pixmap = QPixmap::fromImage(qImage);
  ui->label->clear();
  ui->label->setPixmap(pixmap);
 // }
}
MainWindow::MainWindow(QWidget *parent):
  QMainWindow(parent),
  ui(new Ui::MainWindow)
{
  ui->setupUi(this);
    //Busca el numero donde se quedo el ultimo registro
    QString direction = "contador.txt";
    QFile archivo(direccion);
    archivo.open(QIODevice::ReadOnly);
    QTextStream estacosa(&archivo);
    QString contenidoArchivo = (estacosa.readAll().toUtf8().constData());
    int i2 = contenidoArchivo.indexOf("<");</pre>
    int i3 = contenidoArchivo.indexOf(">");
    QString contadorLeido = contenidoArchivo.mid(i2+1,(i3-i2-1));
    contador = contadorLeido.toInt();
```

```
ui->radioButton->setChecked(true);//Hacer uno de los radio button este activado default
  QTimer *cronometro=new QTimer(this);
  connect(cronometro, SIGNAL(timeout()), this, SLOT(fTimer()));
  cronometro->start(30);
  //if(!camara.isOpened()) camara("http://192.168.43.243:8080/video");
  // if(!camara.isOpened()) camara.open("http://192.168.1.243:8080/video");
             qDebug() << "No se abre la camara 22";
  //else
}
MainWindow::~MainWindow()
  delete ui;
}
void MainWindow::on_pushButton_clicked()
  contador++;//Ingrementar el contador del LCD Number
  ui->lcdNumber->display(contador);//Poner lo que hay de la variable contador
  nombre = ui->lineEdit->text(); //Guarda lo que hay en cada lineEdit en una variable
  quien = ui->lineEdit 2->text();
  empresa = ui->lineEdit_3->text();
  if(ui->radioButton->isChecked()) tipoIdentificacion = "INE"; //Seleccion de tipo de Indetificacion
  else if(ui->radioButton 2->isChecked()) tipoIdentificacion = "PASAPORTE";
  else if(ui->radioButton 3->isChecked()) tipoIdentificacion = "LICENCIA";
  else if(ui->radioButton 4->isChecked()) tipoIdentificacion = ui->lineEdit 4->text();
  numeroldentificacion = ui->lineEdit_5->text();
  QFile reporte("reporte.csv"); //Checa si existe un archivo
  QFileInfo existe("reporte.csv");
  //Sino existe abre el archivo y crea cada uno de los campos correspondientes
  if(!existe.exists()){
    reporte.open(QIODevice::WriteOnly);
         QTextStream out(&reporte);
         out << "Nombre" << ",";
         out << "A quien visita " << ",";
         out << "Empresa" << ",";
         out << "Tipo de Identificacion" << ",";
         out << "Numero de Identificacion" << ",";
         out << "NOmbre de imagen" << ",";
         out << "Fecha y Hora de ingreso \n";
         reporte.close();
  }
  if(existe.isFile() && existe.exists()){ //Verifica si el archivo existe
  if(reporte.open(QIODevice::WriteOnly | QIODevice::Append)){ //Escribir en la ultima fila del excel
    QTextStream out(&reporte);
    out << nombre.toUtf8().constData() << ",";
```

```
out << quien.toUtf8().constData() << ",";</pre>
    out << empresa.toUtf8().constData() << ",";
    out << tipoldentificacion.toUtf8().constData() << ",";
    out << numeroldentificacion.toUtf8().constData()<< ",";</pre>
    out << "imagen" << contador << ".jpg" << ",";
    out << tiempoLocal.toUtf8().constData() << "\n";
    reporte.close();
    QString nombreImagen = "imagen"+ QString::number(contador)+".jpg";
    cv::imwrite(nombreImagen.toUtf8().constData(),IMAGEN);
  }
  }
  else{
  }
   // if(!reporte.open(QIODevice::WriteOnly | QIODevice::Append)) qDebug() << "Error 3210";
}
*mainwindow.h
#ifndef MAINWINDOW H
#define MAINWINDOW_H
#include <QMainWindow>
namespace Ui {
class MainWindow;
}
class MainWindow: public QMainWindow
  Q_OBJECT
public:
  explicit MainWindow(QWidget *parent = 0);
  ~MainWindow();
public slots:
  void fTimer();
private slots:
  void on_pushButton_clicked();
private:
  Ui::MainWindow *ui;
};
#endif // MAINWINDOW_H
```