

**Universidad Politécnica de Quintana Roo**

**Ingeniería en software**

**Desarrollo de sistemas inteligentes**

**Manual de Programador**

**Proyecto de Sistema de Inicio de Sesión:**

**Login y Registro para Reconocimiento de Rostros**

**Profesor:**

**Jose León Borges**

**Alumnos:**

**Chuc ku Shary Yareli**

**Arrioja Zamudio José Luis**

**Presentación**

En el presente documento se recopilara o estructurar las medidas técnicas e informáticas dadas al proyecto de facturación conseguidas al análisis y a las conclusiones del equipo de trabajo.

Se busca dar a conocer todos los aspectos tomados en tiempo y forma de tal manera que no quede duda de que tipo de procedimientos fueron dados al armar y organizar todos y cada uno de los elementos de programación y diseño. Se consideraron muchas plataformas para desarrollar el proyecto y en la que se enfocó el equipo de trabajo fue la plataforma de java.

**Descripción del manual**

**Objetivo general**

El objetivo general del manual es darle al lector un poco de los cambios, direcciones y hechos ejecutados en el programa al compilarse. Observa cual fue el tipo de programación codificada y cuáles fueron las librerías y métodos para las acciones secundarias.

**Herramientas**

Se ocupó las librerías de OpenCv para la plataforma de java, con el propósito de modificaciones para imágenes.

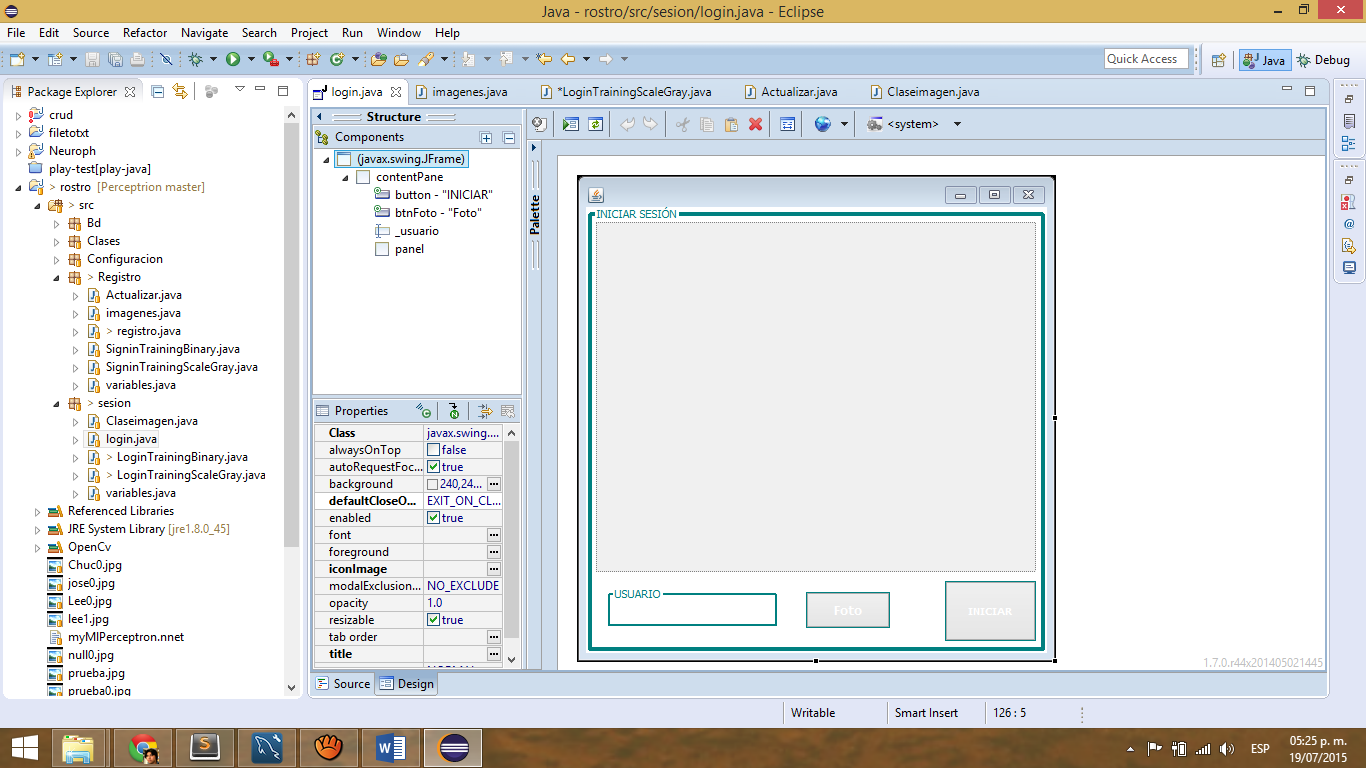
También se usó librerías de Mysql para la conexión con base de datos.

**IDE**

El IDE ocupado fue Eclipse Luna, debido al uso mas practica en las librerías de OpenCV

**Estructura de programación**

**Login**

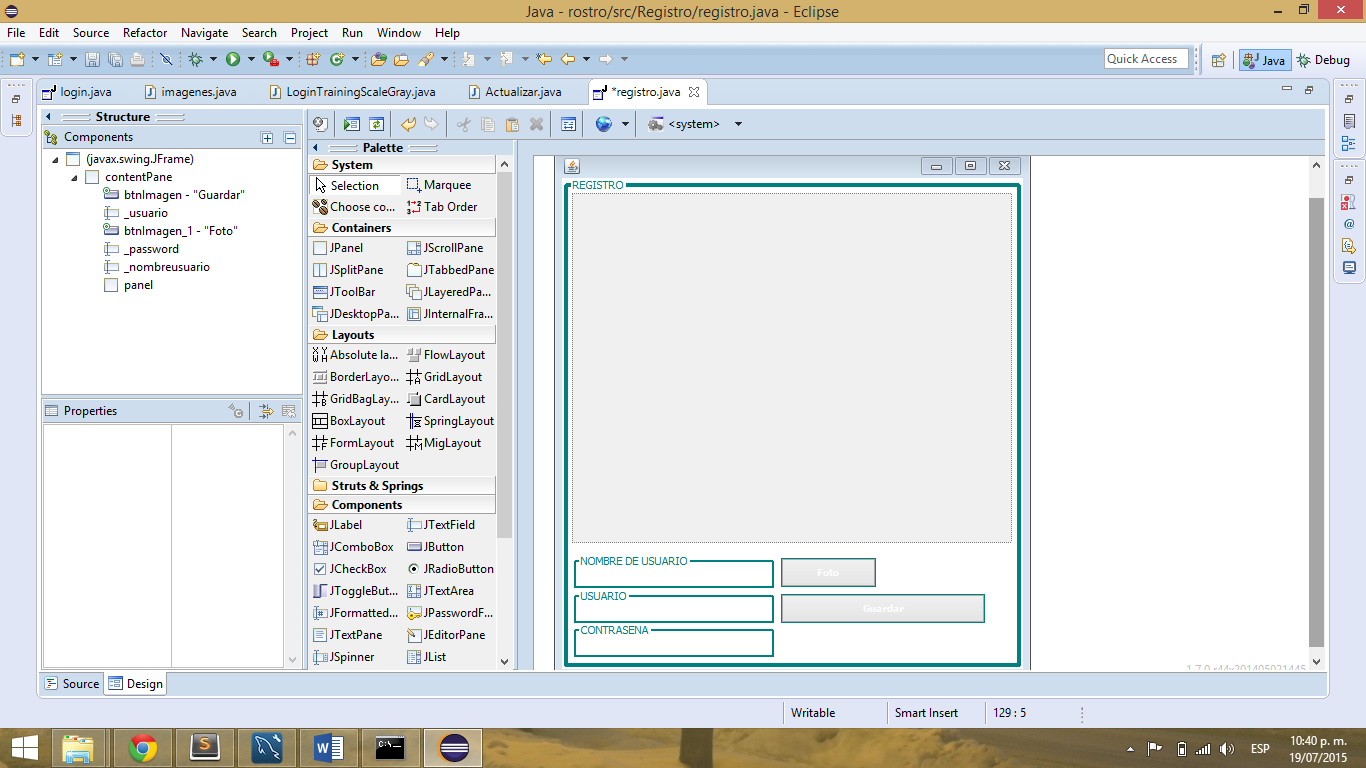


1. Las clases de los frames están separadas por carpetas, sin embargo cada carpeta de cada Jframe tiene su clase para ejecutar la base de datos y tiene su carpeta personal para recursos externos (Imágenes, estilos, colores).
2. Algunos de los componentes del Jframes estos son jTextfield, usado para la anterioridad del programa.
3. Los componentes del Jframes:
   * Jframe(ventana de compilación actual)
   * \_usuario [JTextField] (caja de texto)
   * button – “INICIAR” [JButton] (Botón para cargas las librerías de entrenamiento de imágenes
   * btnfoto – “foto” [JButton] (Botón para iniciar la conversión del rostro detectado en la camara
   * panel [Jpanel](panel de la cámara para visualizar los rostros detectados)

Clases:

* Claseimagen: Esta clase toma la captura de la cámara y se lleva los parámetros de los cuadrantes del rostro, con esto recorta la imagen del rostro como incentivo de que el fondo no alterara los datos que se ingresen, en ella hay 2 metodos:
  + Guardar: este ocupa la librerías de OpenCV para recortar la imagen convertirla en una imgen escala de grises y transformarla en una imagen de tamaño 100x100, al concluir, transforma cada pixel de la imagen en un numero especifico, se introduce este número en 2 vectores distintos, ya se binario 0 y 1 o escala de gris del 0 al 15.
  + getPixelData transforma el pixel tomado en 3 numeros de color, según la escala de colores RGB en su caso sea rojo, azul y verde (Red, Green and Blue).
* Variables: esta clase se ocupa de almacenar de manera temporal los datos mas relevantes e importantes de toda la secuencia de inicio de sesión, en su caso son:
  + \_id= id de usuario que quiere ingresar.
  + Bw= variable String donde irán los números en binario de la imagen.
  + Gray= variable String donde irán los números en escala de grises de la imagen.
* LoginTrainingBinary: Esta clase constituye el entrenamiento de imágenes en binario, en este hay muchos métodos, se trataran de nombrar los más importantes debido a la complejidad de esta clase.
  + Constructor: este método genera los vectores donde estarán las imágenes en binarios de los usuarios ya registrados y del que quiere ingresar, así como también delimita cuantas entradas y salidas tendrá el método.
  + Principal: este es el método principal que llama a todos los métodos del conjunto del algoritmo de perceptron, desde el cálculo de pesos hasta la iniciación de datos.
  + MuestraResultados: este método desglosa todos los pesos de las imágenes en binario, tal y como es previsto, los pesos son muy parecidos, debido a que en binario solo puede haber un reconocimientos de objetos.
* LoginTrainingScaleGray: Esta clase constituye el entrenamiento de imágenes en escala de grises, en este hay muchos métodos, se trataran de nombrar los más importantes debido a la complejidad de esta clase.
  + Constructor: este método genera los vectores donde estarán las imágenes en escala de grises de los usuarios ya registrados y del que quiere ingresar, así como también delimita cuantas entradas y salidas tendrá el método.
  + Principal: este es el método principal que llama a todos los métodos del conjunto del algoritmo de perceptron, desde el cálculo de pesos hasta la iniciación de datos.
  + MuestraResultados: este método es muy complejo, en este se realiza el cálculo de distancias entre pesos, tomando en cuenta la salida de cada entrada se genera un promedio de pesos, en base a estos, se compara las distancias que tienen estos con el peso de la última entrada, en este caso el usuario que desea ingresar. Al delimitar esto se hace una comparativa de si es realmente un usuario ingresado y cual es o si no está de manera común en el sistema.

**Registro de usuarios**



1. Estos Jtextfield son para registrar a los usuarios nuevos, los primordiales son el usuario y contraseña.
2. Botón para registrar 5 fotos del rostro que se detecte, de manera consecutivo, la clase DaemonThread estará constantemente tomando imágenes de la cámara pero este botón permite que se lleve solo las 5 que son necesarias.
3. Este botón inicia la secuencia de registro del nuevo usuario, como el usuario, la contraseña y la imagen en binario y escala de grises.
4. Componentes del Jframe
   * Jframe(ventana de compilación actual)
   * \_usuario [JTextField] (caja de texto)
   * \_password [JTextField] (caja de texto)
   * \_nombreusuario [JTextField] (caja de texto)
   * btnImagen – “Guardar” [JButton] (Botón para cargas las librerías de entrenamiento de imágenes
   * btnImagen\_1 – “foto” [JButton] (Botón para iniciar la conversión del rostro detectado en la camara
   * panel [Jpanel](panel de la cámara para visualizar los rostros detectados)

Clases:

* imagenes: Esta clase toma las 5 capturas de la cámara y se lleva los parámetros de los cuadrantes del rostro, con esto recorta las imágenes del rostro como incentivo de que el fondo no alterara los datos que se ingresen, en ella hay 2 metodos:
  + Guardar: este ocupa la librerías de OpenCV para recortar las imágenes, convertirlas en imágenes en escala de grises y transformarlas en imágenes de tamaño 100x100, al concluir, transforma cada pixel de las imágenes en un numero especifico, se introduce este número en 2 vectores distintos, ya se binario 0 y 1 o escala de gris del 0 al 15.
  + getPixelData transforma el pixel tomado en 3 numeros de color, según la escala de colores RGB en su caso sea rojo, azul y verde (Red, Green and Blue).
* Variables: esta clase se ocupa de almacenar de manera temporal los datos más relevantes e importantes de toda la secuencia de registro, en su caso son:
  + \_idusuario= id de usuario que quiere registrarse.
  + \_id= id de imagen.
  + Bw= vector String donde irán los números en binario de las 5 imágenes.
  + Gray= vector String donde irán los números en escala de grises de las 5 imágenes.
* SigningTrainingBinary: Esta clase constituye el entrenamiento de imágenes en binario, en este hay muchos métodos, se trataran de nombrar los más importantes debido a la complejidad de esta clase.
  + Constructor: este método genera los vectores donde estarán las imágenes en binarios de los usuarios ya registrados y del usuario nuevo por registrarse, así como también delimita cuantas entradas y salidas tendrá el método.
  + Principal: este es el método principal que llama a todos los métodos del conjunto del algoritmo de perceptron, desde el cálculo de pesos hasta la iniciación de datos.
  + MuestraResultados: este método desglosa todos los pesos de las imágenes en binario, tal y como es previsto, los pesos son muy parecidos, debido a que en binario solo puede haber un reconocimientos de objetos, una vez delimitados los pesos, se actualizan en la base de datos, con sus respectivos id de usuario.
* SigningTrainingScaleGray: Esta clase constituye el entrenamiento de imágenes en escala de grises, en este hay muchos métodos, se trataran de nombrar los más importantes debido a la complejidad de esta clase.
  + Constructor: este método genera los vectores donde estarán las imágenes en escala de grises de los usuarios ya registrados y del usuario nuevo por registrarse, así como también delimita cuantas entradas y salidas tendrá el método.
  + Principal: este es el método principal que llama a todos los métodos del conjunto del algoritmo de perceptron, desde el cálculo de pesos hasta la iniciación de datos.
  + MuestraResultados: este método desglosa todos los pesos de las imágenes en escala de grises, una vez delimitados los pesos, se actualizan en la base de datos, con sus respectivos id de usuario.