

```

import org.opencv.core.Core;
import org.opencv.core.Mat;
import org.opencv.core.MatOfRect;
import org.opencv.core.Rect;
import org.opencv.core.Scalar;
import org.opencv.core.Size;
import org.opencv.imgcodecs.Imgcodecs;
import org.opencv.imgproc.Imgproc;
import org.opencv.objdetect.CascadeClassifier;

public class Condominio {

    private CascadeClassifier faceCascade;
    private boolean visitanteDesordeiro;
    private boolean visitanteSupervisionado;
    private boolean multaEnviada;

    public Condominio() {
        // Carrega o classificador facial
        faceCascade = new CascadeClassifier();
        faceCascade.load("haarcascade_frontalface_default.xml"); // Caminho para o arquivo XML
        do classificador facial

        // Inicializa variáveis de controle
        visitanteDesordeiro = false;
        visitanteSupervisionado = false;
        multaEnviada = false;
    }

    // Método para analisar uma imagem e tomar ações com base no reconhecimento facial
    public void analisarImagem(String morador, String visitante, String imagemPath) {

```

```

Mat imagem = Imgcodecs.imread(imagemPath); // Carrega a imagem

// Converte a imagem para escala de cinza para melhorar o desempenho do
reconhecimento facial

Mat imagemCinza = new Mat();

Imgproc.cvtColor(imagem, imagemCinza, Imgproc.COLOR_BGR2GRAY);

// Redimensiona a imagem para acelerar o processo de detecção facial

Imgproc.resize(imagemCinza, imagemCinza, new Size(640, 480));

// Detecta faces na imagem

MatOfRect faces = new MatOfRect();

faceCascade.detectMultiScale(imagemCinza, faces);

if (faces.toArray().length > 0) {
    // Face detectada

    if (visitanteDesordeiro && !visitanteSupervisionado) {
        System.out.println("ALERTA: O visitante " + visitante + " é desordeiro! Necessita
assinatura do morador para liberar.");

        visitanteSupervisionado = true;
    } else if (visitanteDesordeiro && visitanteSupervisionado) {
        System.out.println("O visitante " + visitante + " está sob supervisão do sistema de
segurança.");
    }

    // Processa ações do visitante

    // ...

    // Atualiza os status de desordeiro, supervisionado e multa enviada

    visitanteDesordeiro = true;

    visitanteSupervisionado = true;

    multaEnviada = false;

```

```

    } else {

        // Nenhuma face detectada

        visitanteDesordeiro = false;

        visitanteSupervisionado = false;

    }
}

// Método para enviar uma multa ao morador
public void enviarMulta(String morador) {

    if (visitanteDesordeiro && visitanteSupervisionado && !multaEnviada) {

        System.out.println("O morador " + morador + " foi multado.");

        multaEnviada = true;

    }

}

public static void main(String[] args) {

    System.loadLibrary(Core.NATIVE_LIBRARY_NAME);

    Condominio condominio = new Condominio();

    // Exemplo de uso

    String morador = "Morador1";

    String visitante = "Visitante1";

    String imagemPath = "path_da_imagem.jpg";

    // Teste 1: Detecção Facial Correta

    condominio.analisarImagem(morador, visitante, imagemPath);

    // Teste 2: Alerta de Visitante Desordeiro

    condominio.analisarImagem(morador, visitante, imagemPath);

```

```
// Teste 3: Supervisão do Sistema  
condominio.analisarImagem(morador, visitante, imagemPath);  
condominio.analisarImagem(morador, visitante, imagemPath);  
  
// Teste 4: Envio de Multa  
condominio.analisarImagem(morador, visitante, imagemPath);  
condominio.enviarMulta(morador);  
}  
}
```