

## 3C: Company System With Face Recognition To Log

### 1 Objetivo

Diseñar una solución computacional, con un enfoque transaccional-comercial, relacionada con conceptos de POO, recursividad, iteración, almacenamiento en disco y estructuras de datos, con el fin de aplicar la teoría en problemas específicos, considerando las buenas prácticas de programación. Además, de interactuar con conceptos matemáticos de manera general y con dispositivos periféricos (cámara) del computador.

### 2 Motivación

El campo de acción de Ingeniería en Computadores abarca software de alto y bajo nivel, así como aquellas aplicaciones que involucren conceptos matemáticos. En este caso en específico, el reconocimiento de patrones dentro una imagen. Cabe resaltar que el proyecto tiene un enfoque comercial, con el fin de que el estudiantado se familiarice con el diseño y modelado de aplicaciones que necesitan las empresas en el mercado laboral.

La idea fundamental del proyecto es crear pensamiento abstracto en el estudiantado, con el fin de que puedan modelar e implementar soluciones computacionales de alta calidad, tomando en consideración los requerimientos técnicos del problema.

Adicionalmente, el Reconocimiento de Patrones y la Inteligencia Artificial son áreas de estudio para un Ingeniero en Computadores y con gran auge en la actualidad; por lo tanto, este proyecto pretende dar una introducción al reconocimiento de patrones a través del reconocimiento de un rostro en una foto.

Finalmente, puede visitar la página de experimentos de Google para conocer algunos avances hechos en el campo de la Inteligencia Artificial. Por ejemplo, el experimento llamado *Semiconductor* intenta reconocer los gestos hechos por un director de orquesta y permite dirigir los músicos utilizando la cámara de su computadora.

### 3 Descripción de la aplicación

La empresa 3C se dedica a dar servicios de jardinería a otras empresas y clientes particularidades. Con la digitalización de los procesos, 3C detecta la necesidad de crear una plataforma digital para controlar la parte contable y generar las facturas como un archivo digital. La idea fundamental de los gerentes de 3C es colocar todas las acciones que se hacían en físico en una aplicación de escritorio.

La plataforma digital deberá contar con las siguientes características:

#### 3.1 Login con *Face Recognition*

La aplicación deberá tener un login que se realizará por medio de reconocimiento facial, esta sección consiste en tener la capacidad de tomar una foto desde la cámara y validar si la persona que está ingresando tiene acceso a la plataforma. Debe existir una carpeta con fotografías de las caras de todas las personas que tienen acceso a la aplicación. Si la persona que está intentando ingresar no se encuentra en la carpeta de personas autorizadas, no lo debe dejar entrar a la aplicación.

La biblioteca **face\_recognition** facilita el procesamiento de las imágenes. También se recomienda el uso de OpenCV para la captura de imágenes. Puede referirse al siguiente ejemplo donde se explica cómo reconocer una persona en una foto.

### 3.2 Registro

Se debe de proveer una sección para registrar a un usuario. Esta consiste en proveer  $n$  imágenes para entrenar el modelo, así como obtener la información básica de la persona tal como: nombre completo, edad, cédula, email y residencia.

### 3.3 Pantalla principal

Una vez validado los datos de entrada, deberá aparecer la pantalla principal, la cual será la encargada de obtener todas las funciones que se pueden realizar (estilo menú). Esta ventana será un “orquestador” entre las otras ventanas. Las funciones que debe de realizar la aplicación están:

#### 3.3.1 Realizar factura

Esta ventana proporcionará los medios para generar una factura, se deberá tomar todos los datos del cliente, así como los servicios brindados. El usuario podrá agregar tantos servicios como desee (ingresados desde una lista, desplegada), en cada servicio se debe de ingresar el porcentaje de descuento, en caso de que aplique. La figura 1 muestra un ejemplo de lo que se espera en esta sección. Además, al finalizar de agregar todos servicios brindados, se deberá generar la factura correspondiente en PDF y almacenarlo en un directoria llamado “invoices”. El sistema debe mantener un registro de cada factura de manera lógica, ya que se debe de hacer ciertas acciones contables. Cabe resaltar que el impuesto de venta agregado se debe calcular por aparte, también se resalta la importancia de la fecha creada y la fecha de vencimiento (tres días hábiles). Se recomienda el uso de archivos planos para el almacenamiento de los datos.

Item	Quantity	Price	Amount

Subtotal

Tax

Figure 1: Ejemplo de agregar servicio

#### 3.3.2 Buscar facturas por fechas

Esta sección consiste en proporcionar un mecanismo para visualizar todas las facturas en un rango de fechas (inicial-final). Se debe de hacer un listado que muestre el número de consecutivo, el nombre del cliente, el monto a pagar y la fecha de vencimiento. Así como la suma de los totales de dichas facturas.

#### 3.3.3 Eliminar factura

Esta funcionalidad consiste en listar todas las facturas en orden ascendente (número de factura) y proporcionar un mecanismo de eliminación de una o más facturas. También debe eliminarse el archivo PDF generado.

Cuando se listan las facturas, se debe visualizar el nombre del cliente, el número de consecutivo y el monto a pagar.

#### **3.3.4 Generar Informe**

Esta sección consiste en generar cuánto es el monto total de las facturas y el monto total de los impuestos de todas las facturas generadas. También se debe proporcionar un mecanismo que haga lo mismo, pero con un rango de fechas.

#### **3.3.5 Agregar servicio**


Esta funcionalidad permite agregar servicios nuevos para que puedan brindarse. Los datos del servicio son el identificador secuencial(generado automáticamente), la descripción y el costo del mismo.

### **3.4 Actualizar servicio**

Esta sección permite actualizar la descripción o el costo del servicio. Para actualizar el servicio se debe de mostrar todos los servicios y el usuario selecciona el que quiere actualizar.

### **3.5 Mostar PDF de la factura**

Esta sección debe proporcionar el mecanismo para elegir una factura en específico y mostrar el PDF de la misma. Cuando se desea listar las facturas debe aparecer el número de consecutivo (generado automáticamente cuando se crea) y el nombre del cliente de dicha factura. En la figura 2 se muestra un ejemplo de como debe de verse la factura.



**+ Your  
Logo**

# INVOICE

**Your Company Name**

Phone: (999) 999-9999  
 Your email address  
 Your street address  
 Your City, State Zip

Invoice # 0001  
 Invoice date 99/99/0000  
 Due date 99/99/0000

**Bill To:**

Customer name  
 Customer email  
 Billing address

**Ship To:**

Customer name  
 Shipping address

Description	Quantity	Price	Amount
Item 1	1	\$0.00	\$0.00
Description 1			
Item 2	1	\$0.00	\$0.00
Description 2			
Item 3	1	\$0.00	\$0.00
Description 3			

	Subtotal	\$0.00
	Discount	-\$0.00
	Shipping	\$0.00
	Tax total	\$0.00
	Other	\$0.00
	<b>Total</b>	<b>\$0.00</b>

**Notes**

Thank you for your business.

This invoice is provided by PayPal




Figure 2: Ejemplo de agregar servicio

## 4 Aspectos generales

A continuación se presentan los aspectos generales del proyecto:

- El proyecto será desarrollado en grupos de tres personas. Cualquier intento de fraude será castigado de acuerdo con reglamento.
- La fecha de entrega será jueves 13 de agosto a media noche.
- Debe entregar el código fuente y la documentación que explique cómo funciona su proyecto. Las partes que debe contener son: Introducción, Marco teórico (explicación de bibliotecas utilizadas), Diagrama de arquitectura básico, Diagrama de clases, Secciones de código importante y Conclusiones. Para la documentación es obligatorio el uso de  $\text{\LaTeX}$  y debe tener un formato de **artículo científico** con una extensión no mayor a 3 páginas. Finalmente, la documentación no tendrá peso sobre la nota si no logra implementar al menos el 50% del programa.
- Se programará una defensa de su proyecto luego de la entrega del mismo, en el cual demostrará personalmente el funcionamiento del proyecto.
- En la solución puede utilizar iteración o recursividad indistintamente.
- Se recomienda el uso de Tkinter o Pygame para desarrollar la interfaz gráfica.
- Es obligatorio el uso de *GitHub* y se debe mostrar el aporte de cada uno de los miembros del grupo en la historia del repositorio.

## 4.1 Evaluación

- Login con reconocimiento de rostro - 40%
- Funcionalidad del programa - 30%.
- Documentación externa e interna - 30%.
- **Total - 100%.**