

UNIVERSIDAD INTERAMERICANA DE PANAMA
FACULTAD DE INGENIERIA

Carrera:

ING. En sistemas computacionales

Materia:

Base de datos II.

Estudiantes:

Efraín Fox 3-749-47

Hector Campos 8-994-273

Adrian Batista 8-954-2031

Anthony Delgado 3-751-779

Profesor:

Maryon Torres

25 de abril 2024

1. Introducción

- **Título del proyecto:** PharmaTech
- **Autores:** Efrain Fox, Anthony Delgado, Hector Campos, Adrian Batista.
- **Fecha:** 25 de Abril de 2024
- **Resumen:** El Sistema de Gestión de Farmacia "PharmaTech" es una aplicación informática diseñada para ayudar a las farmacias a gestionar sus operaciones de manera eficiente y eficaz.

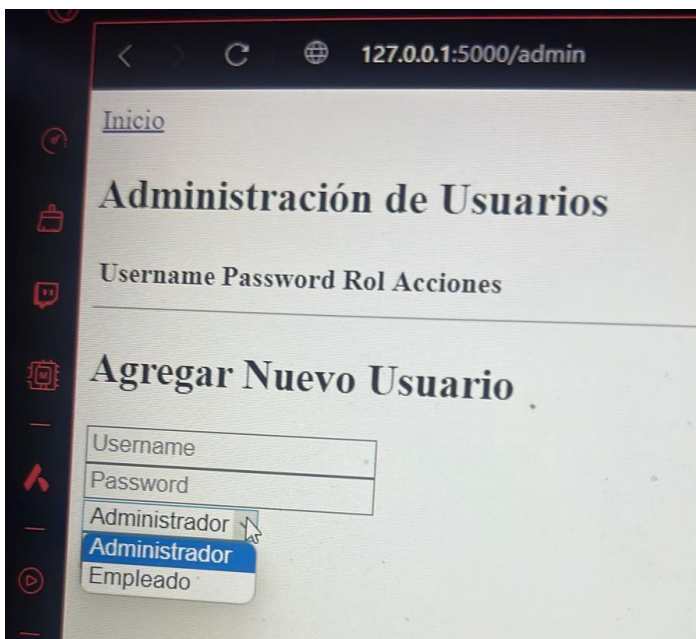
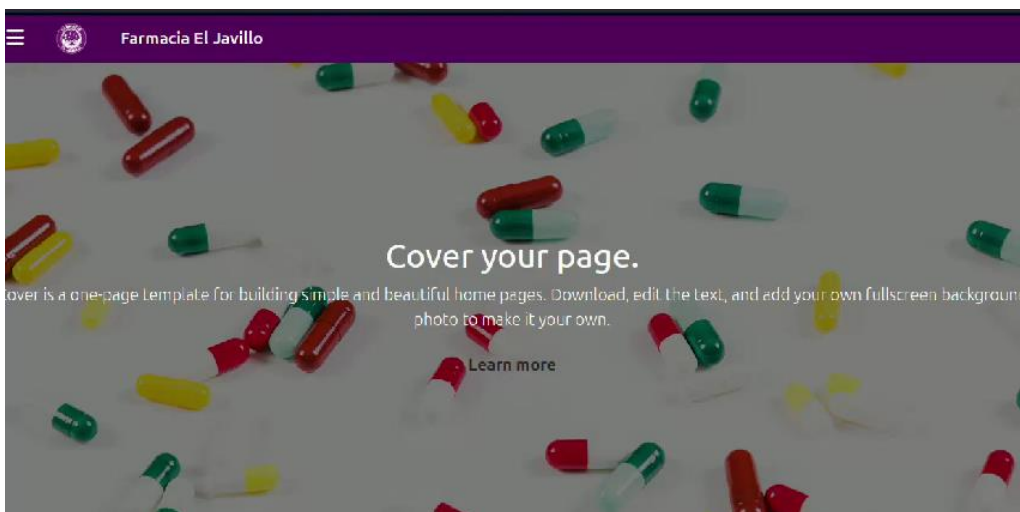
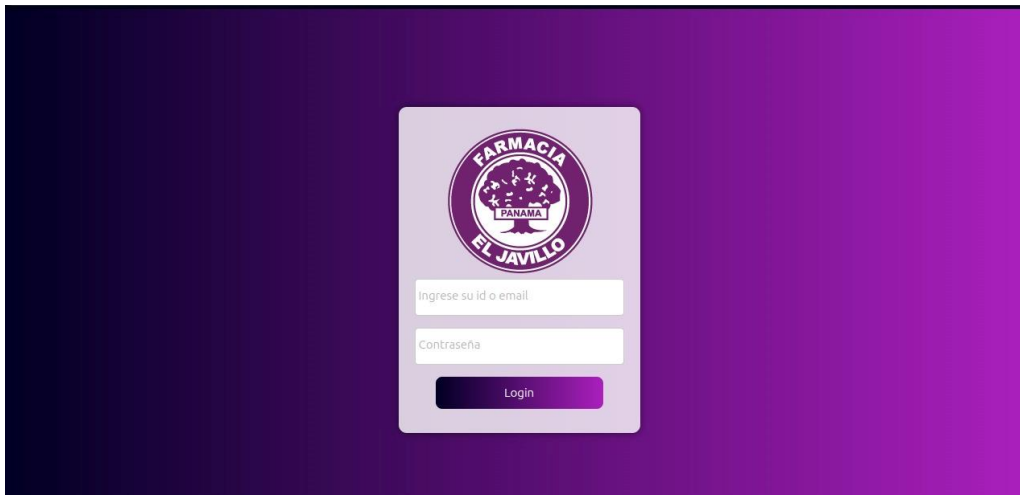
2. Definición del problema

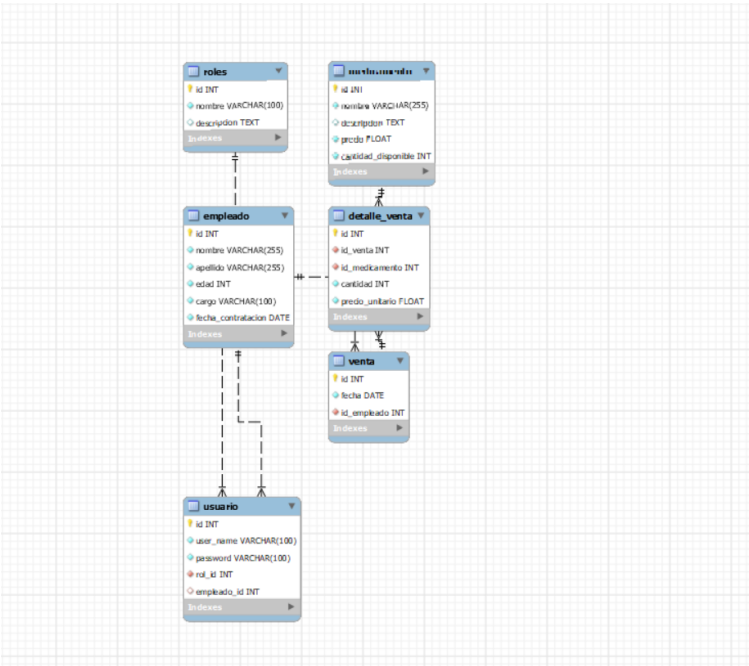
Las problemáticas que enfrenta una farmacia incluyen la gestión de inventario, el cumplimiento normativo, la seguridad y confidencialidad de los datos, la competencia en el mercado, la atención al cliente, la integración tecnológica, las relaciones con proveedores y la rentabilidad del negocio.

3. Diseño del software

Para la gestión de la base de datos, optamos por utilizar MySQL, una base de datos relacional, la cual decidimos alojar en AWS mediante el servicio RDS para garantizar su disponibilidad y seguridad. Configuramos un servidor utilizando Flask, una herramienta ligera y flexible para el desarrollo web en Python. Para el diseño de la interfaz de usuario (UI), aprovechamos Bootstrap 5, una biblioteca de código abierto que nos proporcionó componentes y estilos predefinidos para agilizar el proceso de desarrollo y asegurar una experiencia de usuario consistente.

En cuanto a las funcionalidades de la aplicación, diseñamos una interfaz intuitiva que permite a los usuarios acceder y visualizar información relevante de manera rápida y eficiente. Los empleados pueden acceder a funciones como la visualización de ventas realizadas y sus detalles, así como la gestión de empleados, generación de reportes y control de inventario de medicamentos disponibles. Implementamos un sistema de autenticación mediante un login, donde cada empleado puede ingresar con sus credenciales para acceder a las funciones específicas asignadas a su rol dentro de la farmacia.





| Farmacia El Javillo | | | | | |
|--|---|--------|---------------------|----------|--|
| <div><div></div><div></div><div></div><div></div><div></div></div> | | | | | |
| <div><div></div><div></div><div></div><div></div><div></div></div> | | | | | |
| Nombre | Descripción | Precio | Cantidad Disponible | Imágenes | |
| Panadol | Alivia dolor | 2.7 | 9 | | |
| Paracetamol | Analgesico y antipiretico | 5.5 | 100 | | |
| Ibuprofeno | Antiinflamatorio no esteroideo | 7.75 | 80 | | |
| Amoxicilina | Antibiotico | 10.25 | 50 | | |
| Aspirina | Analgésico, antiinflamatorio y antipirético | 4.2 | 120 | | |

Menu de Opciones

Inicio

Inventario

Empleados

Reportes

Dashboard

REPORTES GUARDADOS

Mes actual

Ultimo trimestre






Anual

Configuración

Cerrar Sesión

nacia El Javillo

InventarioEmpleadosReportesDashboardInicio

| Descripción | Precio | Cantidad Disponible | Imágenes |
|---|--------|---------------------|---|
| Alivia dolor | 2.7 | 9 |  |
| Analgesico y antipirético | 5.5 | 100 |  |
| Antiinflamatorio no esteroideo | 7.75 | 80 |  |
| Antibiotico | 10.25 | 50 |  |
| Analgésico, antiinflamatorio y antipirético | 4.2 | 120 |  |

Amazon RDS

Panel

Bases de datos

Editor de consultas

Información sobre rendimiento

Instantáneas de

Exportaciones en Amazon S3

Copias de seguridad automatizadas

Instancias reservadas

Proxies

Grupos de subredes

Grupos de parámetros

Grupos de opciones

Versiones de motor personalizadas

Integraciones sin extracción, transformación y carga (ETL)

Nuevo

RDS > Bases de datos > pharmatech

pharmatech

Resumen

Identificador de base de datos
pharmatech

CPU
-

Estado
Disponible

Clase
db.t3.micro

Rol
Instancia

Actividad actual

Motor
MySQL Community

Región y AZ
us-east-1a

Recomendaciones
1 Informativo

Conectividad y seguridad

Supervisión

Registros y eventos

Configuración

Integraciones sin extracción, transformación y carga (ETL)

Conectividad y seguridad

Punto de enlace y puerto

Redes

Seguridad

Punto de enlace
pharmatech.c1084u60fuv.us-eas
t-1.rds.amazonaws.com

Zona de disponibilidad
us-east-1a
VPC

Grupos de seguridad de la VPC
default (sg-091f8da63af0e8af8)
Activo
myyvcnew (sg-007a53af1d7cca80c)

Comentarios

© 2024, Amazon Web Services, Inc. o sus filiales. Privacidad Términos Preferencias de cookies

```

from datetime import date

from flask import Flask, render_template, redirect, request, url_for, flash
from flask_login import LoginManager
app = Flask(__name__)
app.config[
    'SQLALCHEMY_DATABASE_URI'] = 'mysql://admin:Bd2_123456789@pharmatech.c1084u60ifuv.us-east-1.rds.amazonaws.com/farmacia'
app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False # Desactiva el seguimiento de modificaciones
app.secret_key = 'pepito1334'
# Importa los modelos después de inicializar db
from models import Medicamento, Empleado, Venta, DetalleVenta, db, UsuarioRoles

db.init_app(app)
login_manager = LoginManager(app)
login_manager.login_view = 'login'
# Crea las tablas en la base de datos
with app.app_context():
    db.create_all()

new *
@app.route('/logout', methods=['POST'])
def logout():
    if request.method == 'POST':
        return redirect(url_for('login')) # Redireccionar al usuario al formulario de inicio de sesión

new *
@login_manager.user_loader
def cargar_usuario(id_usuario):
    return Usuario.query.get(id_usuario)

```

```

1 import sqlalchemy
2 from flask_sqlalchemy import SQLAlchemy
3
4 db = SQLAlchemy()
5
6
7 9 usages new *
8 class Usuario(db.Model):
9     id = db.Column(db.Integer, primary_key=True)
10    user_name = db.Column(db.String(100), nullable=False, unique=True)
11    password = db.Column(db.String(100), nullable=False)
12    rol_id = db.Column(db.Integer, db.ForeignKey('roles.id'), nullable=False)
13    is_active = db.Column(db.Boolean, default=True)
14    empleado_id = db.Column(db.Integer, db.ForeignKey('empleado.id'))
15    empleado = db.relationship('Empleado', backref=db.backref('usuario', uselist=False), lazy=True)
16
17 2 usages new *
18 class Roles(db.Model):
19     id = db.Column(db.Integer, primary_key=True)
20     nombre = db.Column(db.String(100), unique=True, nullable=False)
21     descripcion = db.Column(db.Text)
22
23

```

4. Conclusión

El Sistema de Gestión de Farmacia "PharmaTech" es una aplicación informática que utiliza una base de datos MySQL para almacenar información sobre medicamentos, usuario, ventas, empleados y más. Con el objetivo general de automatizar y optimizar los procesos operativos de las farmacias pequeñas y medianas, este sistema ofrece funciones clave como seguimiento de inventario, procesamiento de ventas, gestión de clientes y proveedores, y generación de informes detallados sobre el rendimiento del negocio. Sus objetivos específicos incluyen reducir el tiempo dedicado a la gestión manual, mejorar la precisión del control de stocks, facilitar el seguimiento del historial de compras de los clientes y generar informes para la toma de decisiones.

5. Enseñanzas del proyecto

En este proyecto, adquirimos una valiosa experiencia en trabajo en equipo, comprendiendo las dinámicas y desafíos que implica. Exploramos diferentes metodologías de trabajo y aprendimos a adaptarnos a ellas para maximizar nuestra eficiencia y colaboración. Además, adquirimos habilidades técnicas como el hosting de una base de datos, permitiendo el acceso compartido para todos los miembros del equipo. También exploramos la creación de dashboards para visualizar y analizar datos, lo que nos permitió extraer conclusiones significativas. Aprendimos sobre el desarrollo tanto del backend como del frontend de la aplicación web, lo que amplió nuestro conocimiento y nos proporcionó una comprensión integral del proceso de desarrollo de software.

6. Anexos

Link para visualizar el proyecto en GitHub: https://github.com/Emek459/Base-de-datos-2/tree/main/Proyecto_Final