



FACULTAD DE INGENIERÍA

**CARRERA DE INGENIERÍA DE SISTEMAS
COMPUTACIONALES**

INFORME DE PROYECTO FINAL

<<TÍTULO DEL PROYECTO>>

Autores:

CASTRO LUNA, NICK GONZALO(100%)
LÓPEZ OCHOA, JUAN MANUEL(100%)
SALAS GONZALES, VICTOR MARTÍN(100%)
VALDEZ CORNEJO, PEDRO JOSÉ LEANDRO (100%
participación)

Curso:

DESARROLLO DE APLICACIONES MÓVILES

Docente del Curso:

Apellidos y Nombres

LIMA – Perú

2025-1

Contenido

I. RESUMEN.....	4
II. INTRODUCCIÓN.....	5
2.1. Descripción de la problemática	5
2.2. Descripción del proyecto	6
III. ANÁLISIS DEL PROBLEMA	6
3.1. Antecedentes (<i>Mínimo de 5 antecedentes</i>) ;Error! Marcador no definido.	
3.2. Identificación y formulación del problema	7
3.3. Propuestas	8
3.3.1. Propuesta 01: ;Error! Marcador no definido.	
▪ Descripción ;Error! Marcador no definido.	
▪ Ventajas..... ;Error! Marcador no definido.	
▪ Desventajas..... ;Error! Marcador no definido.	
3.3.2. Propuesta 02: ;Error! Marcador no definido.	
▪ Descripción ;Error! Marcador no definido.	
▪ Ventajas..... ;Error! Marcador no definido.	
▪ Desventajas..... ;Error! Marcador no definido.	
3.3.3. Propuesta 03: ;Error! Marcador no definido.	
▪ Descripción ;Error! Marcador no definido.	
▪ Ventajas..... ;Error! Marcador no definido.	
▪ Desventajas..... ;Error! Marcador no definido.	
3.4. Definición de objetivos	;Error! Marcador no definido.
3.4.1. Objetivo general.....	10
3.4.2. Objetivos específicos	10
V. HERRAMIENTAS DE INGENIERÍA.....	11
VI. GENERACIÓN DE SOLUCIONES.....	12
6.1. Evaluación de tecnologías de aplicaciones móviles	12
Tecnologías.....	12
Ventajas.....	12
Desventajas.....	12

Por ejemplo: Android Studio	¡Error! Marcador no definido.
▪ Compilación rápida.....	¡Error! Marcador no definido.
▪ Ejecución de la app en tiempo real gracias al emulador.¡Error! Marcador no definido.	
▪	¡Error! Marcador no definido.
▪ no soporta el desarrollo para NKD.....	¡Error! Marcador no definido.
VII. METODOLOGÍA DE DESARROLLO A UTILIZAR	13
7.1. Aplicación móvil	¡Error! Marcador no definido.
7.1.1. Evidencias del funcionamiento	¡Error! Marcador no definido.
Colocar la captura de pantalla y descripción de la funcionalidad principal	¡Error! Marcador no definido.
7.2. Repositorios.....	¡Error! Marcador no definido.
Indicar el software de control de versiones usado y el repositorio, asimismo colocar el link de repositorio donde se encuentra los proyectos	¡Error! Marcador no definido.
VIII. COMPLEMENTO.....	14
8.1. Evidencia de Participación	14
8.2. Video de presentación del proyecto (Link).....	15
8.3. PPT sustentación.....	15
8.4. Link del repositorio donde se encuentran los proyectos (código)	15
IX. RESULTADOS.....	15
9.1. Resultados en base a los objetivos del proyecto	15
9.2. Discusión de resultados	15
- Comparar la postura a utilizar en el proyecto.	¡Error! Marcador no definido.
- Argumentar y elaborar la discusión en base a los elementos relacionados a la postura que interactúa en el proyecto... ¡Error! Marcador no definido.	
- Priorizar la información relevante relacionada a la postura seleccionada para el desarrollo del proyecto.	¡Error! Marcador no definido.
X. CONCLUSIONES	16
XI. REFERENCIAS O BIBLIOGRAFÍA	16
XII. ANEXOS	16
NOTAS ACLARATORIAS	19

INDICE DE TABLAS

INDICE DE FIGURAS

INDICE DE ANEXOS

I. RESUMEN.

El proceso de toma de pedidos se lleva a cabo manualmente, usando lápiz y papel, o conversa verbalmente con la cocina en varios restaurantes. El servicio de los clientes, los pedidos, la pérdida de datos y las dificultades para monitorear la efectividad de los empleados en tiempo real son problemas típicos que resultan de esta práctica común. Además, la falta de un sistema centralizado ayuda a evitar que el administrador tenga un control completo y visibilidad sobre las operaciones de los restaurantes.

En el sector gastronómico, la necesidad de optimizar los tiempos de espera y mejorar la calidad del servicio al cliente es una prioridad. Por lo tanto, la creación de una aplicación para teléfonos inteligentes en Android Studio se sugiere que los jóvenes pueden registrar pedidos de una tableta a su cuenta personal, al tiempo que le da al administrador un dispositivo para monitorear y administrar el flujo de trabajo de manera efectiva. La organización interna de los restaurantes mejorará como resultado de esta capacidad de tecnología para reducir los tiempos de espera, minimizar los errores humanos y mejorar la organización interna de los restaurantes.

Palabras clave:

- **Pedidos digitales**
- **Optimización de tiempos**
- **Aplicación móvil**
- **Gestión de restaurante**
- **Automatización de procesos**

Abstract:

In many restaurants, the order-taking process is still carried out manually using pen and paper or through verbal communication with the kitchen. This traditional method often leads to delays in customer service, mistakes in orders, loss of information, and difficulty for managers to supervise staff performance in real time. The lack of a centralized system prevents administrators from having complete control and visibility over the restaurant's operations.

To address this issue, we propose the development of a mobile application in Android Studio that allows waiters to log in through a tablet and take customer orders efficiently using their individual accounts. Additionally, the application includes an administrator login for real-time monitoring and management of the restaurant workflow. This system aims to reduce waiting times, minimize human errors, and improve internal organization within the restaurant.

Keywords:

- Digital orders
- Time optimization
- Mobile application
- Restaurant management
- Process automation

II. INTRODUCCIÓN.

2.1. Descripción de la problemática

En un entorno de restaurante, la eficiencia de la gestión de la toma y el pedido es un factor crítico para garantizar una gran experiencia del cliente. Sin embargo, muchas instalaciones dependen de todas las adicciones tradicionales, como: B. Uso de papel y lino, o comunicación verbal entre el camarero y la cocina. Estas prácticas no solo aumentan la tasa de error al aceptar pedidos, sino que también causan retrasos en el servicio, interrupción, información y dificultades en las operaciones internas adecuadas. Rotación de la tabla que afecta negativamente la rentabilidad del negocio. Además, los administradores tienen herramientas limitadas para monitorear el rendimiento del personal y anali

zar los flujos de trabajo en tiempo real. Este proyecto está destinado a abordar este problema, una aplicación menor desarrollada en Android Studio que tiene como objetivo contribuir a la digitalización de los procesos, reducir la latencia y mejorar la organización de la empresa.

2.2. Descripción del proyecto

Este proyecto consiste en desarrollar pequeñas aplicaciones para dispositivos Android, particularmente para optimizar el proceso de solicitud de recompensa en los restaurantes. La aplicación le permite iniciar una sesión en una tableta con información de inicio de sesión personal, acceder al Menad digital y registrar los pedidos de los clientes. Tan pronto como se aprueben, su pedido se guardará y se enviará automáticamente a la cocina. Esto reduce significativamente los errores de comunicación y acelera los servicios. restaurante. Esta característica facilita la toma de decisiones e identifica posibles cuellos de botella en el proceso de atención. El objetivo principal de este proyecto es reducir los tiempos de espera, mejorar los pedidos con precisión y proporcionar al personal del restaurante herramientas de gestión efectivas.

III. ANÁLISIS DEL PROBLEMA

3.1. Antecedentes del sector de la hospitalidad

La digitalización se ha desarrollado significativamente en los últimos años, con varias cadenas implementando soluciones técnicas para mejorar la eficiencia y la experiencia del cliente. A continuación, se presentan cinco historias relacionadas.

Implementación de un quiosco digital con McDonald's: McDonald's ha comenzado a instalar quioscos digitales en restaurantes en los Estados Unidos. Precisión de pedido.

Uso de la inteligencia artificial en órdenes con Wendy: Wendy probó sistemas de autoservicio basados en la inteligencia artificial española en varios restaurantes.

Soluciones digitales que utilizan macros y soluciones digitales: las macros han lanzado carriles para platos que incluyen herramientas como pedidos de platos y punto de venta de platos, así como diseños para mejorar la gestión de clientes y la experiencia en bares y restaurantes a través de tecnología digital.

Amazon Go y la automatización en ventas minoristas: Amazon GO ha desarrollado una tienda sin un cajero automático que permita a los clientes obtener productos sin hacer cumplir la caja.

Restaurantes automáticos históricos: en el pasado, instituciones como Horn & Hardart en los Estados Unidos

3.1. Identificación y formulación del problema

- *Describir el problema a través de un diagrama de Ishikawa y a la formulación de la pregunta*

Tabla: Temas generales y específicos

Problemas generales

retrasar la atención en los restaurantes generados por el proceso manual y la falta de organización en la gestión del personal y los pedidos.

Problemas específicos

1. No existe un sistema centralizado, lo que resulta en demoras en la comunicación entre el joven y la cocina.
2. Falta de información sobre todos los manuales (papel/lámpara).
3. Falta de supervisión de tiempo real con respecto al flujo de pedidos y al personal del personal.
4. Errores de pedido debido a la confirmación digital o una visualización clara de Menwar.
5. Saturación personal de las horas pico debido a la distribución insuficiente de las tareas.

¿Cómo pueden los tiempos de espera para que los pedidos se registren y administren en los restaurantes ser reducidos por soluciones de tecnología médica que permitan una organización más eficiente y el monitoreo de pedidos en tiempo real?

3.2. Propuestas

Basado en el análisis de problemas y registros grabados, se han propuesto soluciones técnicas que modernizan el proceso de grabación de pedidos en restaurantes a través de aplicaciones globales. Esta herramienta permite a los hombres jóvenes registrar pedidos de sus tabletas y proporcionar un control central sobre sus gerentes en tiempo real. Esto reduce la latencia, los errores, los errores y las mejoras en la experiencia del cliente.

3.3.1. Sugerencia 01: Solicitud de Monville para tomar y gestión de lectores en restaurantes

Descripción:

Se recomienda visualizar el Menado digital, tomar pedidos con marcas de rosas y enviarlos directamente a la cocina. La aplicación incluye funciones como el estado del pedido, el historial de implementación y las actualizaciones automáticas de visualizaciones en tiempo real. Por otro lado, el administrador administró un panel de control que les permitió monitorear el personal de mitmers y configuraciones. sistema.

Ventajas:

Una reducción significativa en la latencia del cliente. del equipo.

Posibilidad de actualizaciones estrictas masculinas o campañas publicitarias de administradores.

Contras:

Necesita una inversión inicial en su tableta y una conexión estable con Internet o la red local.

Los empleados deben capacitar cuando usan la aplicación. o equipo).

Costos o actualizaciones de mantenimiento de software futuros.

3. 3. Definición de objetivos,**3. 4. 1. Objetivo general**

En un restaurante, cree una aplicación para teléfonos inteligentes que ayude a acelerar el proceso de pedido, reducir los tiempos de espera y aumentar la organización interna mediante digitalización y supervisión en tiempo real.

3. 4. 2. Objetivos específicos

Cree un pedido fácil de registrar desde una tableta diseñando una interfaz web intuitiva para hombres jóvenes.

Para garantizar el acceso adecuado a las funciones de aplicación, implementa un esquema de inicio de sesión con diferentes roles (camarero y administrador).

Integre una base de datos en tiempo real que facilite la sincronización de los pedidos entre los jóvenes y la cocina.

Para el administrador, cree un panel de control que le permita ver métricas, órdenes actuales y el historial de intereses.

Al eliminar el uso de papel y lápiz del procesamiento de pedidos, podemos eliminar los errores en el procesamiento del orden.

Evalúe la usabilidad de las aplicaciones a través de estudios de usabilidad y actualizaciones iterativas basadas en comentarios de los empleados.

3.2.1. Objetivo general

Crear una app en Android que haga mejor tomar y manejar pedidos en un restaurante, para bajar tiempos de espera, reducir fallas humanas y ordenar mejor al personal con un sistema digital.

3.2.2. Objetivos específicos

Hacer un diseño fácil que deje a los meseros marcar pedidos en una tableta.

Un sistema de entrada que diferencia a los trabajadores de los jefes.

Unir el programa a una base de datos viva para manejar los pedidos y su estado.

Hacer una parte para administradores que deje ver las ordenes que están en proceso.

Cambiar el método a mano de tomar pedidos por un solución digital para evitar errores y pérdida de datos.

Verificar el rendimiento del sistema durante pruebas en el terreno con usuarios verdaderos en un lugar de comer.

V. HERRAMIENTAS DE INGENIERÍA

Metodología	Ventajas	Desventajas
Cascada	Fácil de entender, adecuado para proyectos pequeños con requisitos claros.	No es flexible, difícil de cambiar requisitos.
Scrum	Mayor interacción con el usuario, entregas rápidas, adaptabilidad.	Requiere experiencia, reuniones frecuentes.
Kanban	Flujo continuo de trabajo, visualización clara.	No define roles ni sprints, menos estructurado.

Selección: Scrum, ya que permite entregar incrementos funcionales en poco tiempo y adaptarse fácilmente a cambios del usuario.

2. Selección de herramientas hardware y software, fundamentado:

- **Hardware:**
 - PC con 8GB RAM mínimo para desarrollo y ejecución de emuladores.
 - Smartphone Android para pruebas reales.
- **Software:**
 - **Android Studio:** Desarrollo de app móvil.
 - **Python + Flask:** Backend ligero y fácil de usar.
 - **MySQL:** Base de datos estructurada y robusta.
 - **Volley (Android):** Comunicación HTTP con el backend.
 - **Postman:** Pruebas de endpoints del servidor.

3. Uso correcto de herramientas:

- **Recolección de datos:** App Android registra platos, mozo y mesa.
- **Análisis de datos:** Se almacenan en MySQL y se muestran en una interfaz HTML.
- **Procesamiento de datos:** El servidor procesa pedidos y los registra con validación.

4. No olvidar requisitos de usuario final, servidor:

- Registro rápido de pedidos por parte del mozo.
- Visualización clara de pedidos en el sistema.

- Función de reinicio de pedidos por mesa.
- Conexión servidor estable (Flask sobre localhost o red local).

VI. GENERACIÓN DE SOLUCIONES

6.1. Evaluación de tecnologías de aplicaciones móviles

En una tabla de doble entrada debe analizar las tecnologías asociadas en cuanto a ventajas y desventajas de las tecnologías existentes

Tecnologías	Ventajas	Desventajas
<ul style="list-style-type: none"> - <i>Android studio</i> - <i>Flutter (dart)</i> - <i>Kotlin multiplatform</i> - <i>React native</i> 	<ul style="list-style-type: none"> ▪ <i>Compilación rápida, emulador integrado, soporte para java/kotlin</i> ▪ <i>Ui moderna, multiplataforma (iOS/Android)</i> ▪ <i>Código compartido entre plataformas</i> ▪ <i>Multiplataforma, gran comunidad</i> 	<ul style="list-style-type: none"> ▪ Requiere PC potente, no soporta NDK ▪ Lenguaje menos conocido, curva de aprendizaje ▪ Menor comunidad que flutter o android

		<ul style="list-style-type: none"> ▪ Problemas con rendimiento o nativo

VII. METODOLOGÍA DE DESARROLLO A UTILIZAR

7.1 Aplicación móvil:

- **Metodología:** *Scrum*
- **Justificación:** Permite avanzar por etapas funcionales (pedidos, reinicio, visualización), con retroalimentación constante del usuario.
- **Cronograma:**

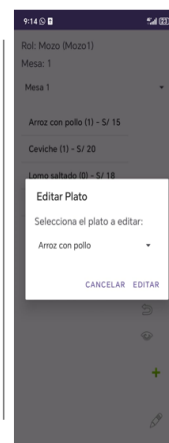
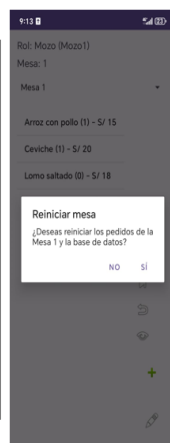
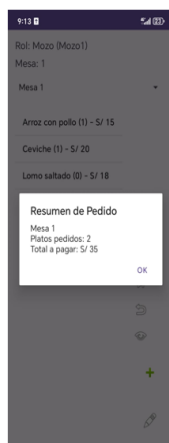
Etapas	Actividades
Planificación	Definir requerimientos, tablas SQL, interfaz
Análisis	Identificar lógica de flujo entre cliente-servidor
Desarrollo	Programar app Android y servidor Flask
Pruebas	Validar pedidos, conexión, reinicio, errores
Implementación	Desplegar en red local y presentar funcionalidad

- **Protección de datos:** No se recolecta información sensible, pero se evita mal uso del servidor con logs, validaciones y control de errores.

7.1.1 Evidencias del funcionamiento:

Capturas de pantalla y descripción:

- **Pantalla principal de pedidos** con selección de mesa, platos y botón de resumen.



- **Visualización en servidor:** Listado de pedidos en HTML con campos (mozo, mesa, precio).

Pedidos Registrados

pollo

Mozo: Mozo1

Mesa: 1

Precio: S/ 114.00

Comentario: Pedido por Mozo1

Fecha: 2025-05-13 22:45:02

lentejas

Mozo: Mozo1

Mesa: 1

Precio: S/ 48.00

Comentario: Ped

Fecha: 2025-05-

Actualizado al cargar la página.

USE restaurante;

SELECT * FROM pedidos;

CREATE TABLE platos (

id INT AUTO_INCREMENT PRIMARY KEY,

nombre VARCHAR(100) NOT NULL,

precio DECIMAL(10,2) NOT NULL

);

id	mozo	mesa	plato	comentario	precio	fecha
30	Mozo1	1	Lomo saltado	Pedido por Mozo1	18.00	2025-05-13
31	Mozo1	1	lentejas	Pedido por Mozo1	48.00	2025-05-13
32	Mozo1	1	pollo	Pedido por Mozo1	114.00	2025-05-13

Automatic content disabled. Use the manually get help current caret position toggle automatic

Tu Restaurante

Ver Pedidos

Listado de pedidos realizados

Ver Logs

Historial de actividad del servidor

Pedidos Registrados

lentejas

Mozo: Mozo1

Mesa: 1

Precio: S/ 48.00

Comentario: Pedido por Mozo1

Fecha: 2025-05-13 22:44:15

Lomo sa

Mozo: Mo

Mesa: 1

Precio: S/

Comentar

Fecha: 202

VIII. COMPLEMENTO

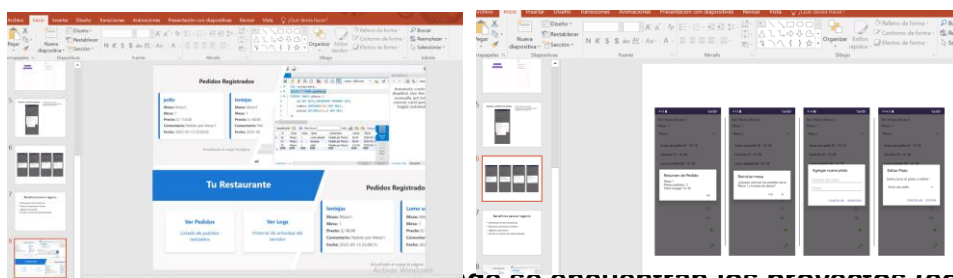
8.1. Evidencia de Participación

(Captura de la participación en debates en el aula virtual y evidencia de haber presentado los avances del proyecto solicitados)

8.2. Video de presentación del proyecto (Link)

<https://files.fm/u/ts4w59aggd>

8.3. PPT sustentación



8.3. Link del repositorio donde se encuentran los proyectos (codigo)

<https://github.com/leandrovaldezcornejo/UPN>

IX. RESULTADOS

9.1. Resultados en base a los objetivos del proyecto

Objetivo	Resultado obtenido
Registrar pedidos por mesa	✅ Funciona correctamente en Android y servidor Flask
Agregar platos nuevos	✅ Desde la app se agregan y editan platos
Ver pedidos y resumen	✅ Función implementada y visible en HTML del servidor
Reiniciar pedidos y base de datos	✅ Función ejecutada desde app Android con efecto en MySQL

9.2. Discusión de resultados

- Se eligió una arquitectura sencilla cliente-servidor que garantiza bajo consumo de recursos.
- La integración entre Android y Flask vía HTTP permitió validar la solución localmente.
- La interfaz es amigable y adaptable a usuarios no técnicos (mozos).
- Se recomienda a futuro implementar autenticación por usuario/mozo.

X. CONCLUSIONES

Conclusiones por objetivo:

1. Se logró implementar una app funcional para toma de pedidos en un restaurante.
2. El backend con Flask y MySQL soporta correctamente las operaciones.
3. La interfaz permite agregar platos y visualizar pedidos.
4. El sistema es escalable y puede evolucionar a una versión online.

Recomendaciones:

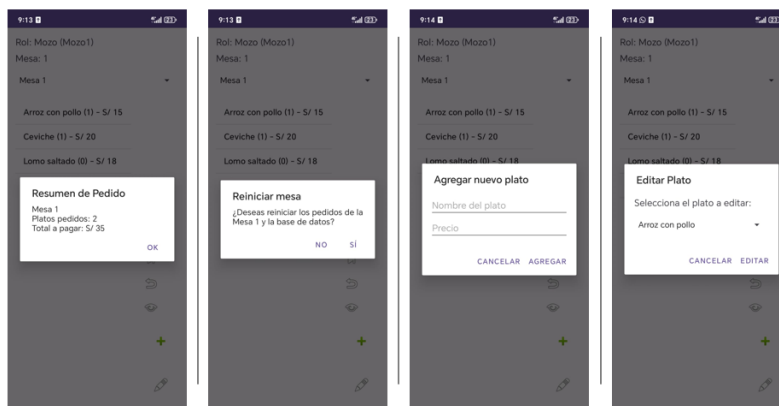
- Migrar el servidor a una IP externa para producción.
- Proteger el acceso con autenticación y cifrado.
- Automatizar la carga de platos al iniciar la app.

XI. REFERENCIAS O BIBLIOGRAFÍA

- Android Developers. (s.f.). *Volley overview*. <https://developer.android.com/training/volley>
- Flask Documentation. (s.f.). *Flask Web Framework*. <https://flask.palletsprojects.com/>
- Oracle. (s.f.). *MySQL Reference Manual*. <https://dev.mysql.com/doc/>
- GitHub Inc. (s.f.). *GitHub Docs*. <https://docs.github.com/>

XII. ANEXOS

XIII. Anexo 1: Captura de funcionamiento de la aplicación móvil (pantalla de pedidos, resumen y edición de platos).



XIV. Anexo 2: Visualización de los pedidos registrados desde el servidor Flask en el navegador (ruta /pedidos).

```

PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL  PUERTOS

* Serving Flask app 'server'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:3000
* Running on http://192.168.0.12:3000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 908-624-549
127.0.0.1 - - [14/May/2025 21:09:21] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [14/May/2025 21:09:22] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [14/May/2025 21:09:34] "GET /pedidos HTTP/1.1" 200 -
  
```

XV. Anexo 3: Script SQL utilizado para crear las tablas platos y pedidos.

Pedidos Registrados

pollo

Mozo: Mozo1
Mesa: 1
Precio: S/ 114.00
Comentario: Pedido por Mozo1
Fecha: 2025-05-13 22:45:02

lentejas

Mozo: Mozo1
Mesa: 1
Precio: S/ 48.00
Comentario: Pedido por Mozo1
Fecha: 2025-05-13 22:44:15

Actualizado al cargar la página.

Tu Restaurante

Ver Pedidos

Listado de pedidos realizados

Ver Logs

Historial de actividad del servidor

Pedidos Registrados

lentejas

Mozo: Mozo1
Mesa: 1
Precio: S/ 48.00
Comentario: Pedido por Mozo1
Fecha: 2025-05-13 22:44:15

Lomo saltado

Mozo: Mozo1
Mesa: 1
Precio: S/ 114.00
Comentario: Pedido por Mozo1
Fecha: 2025-05-13 22:45:02

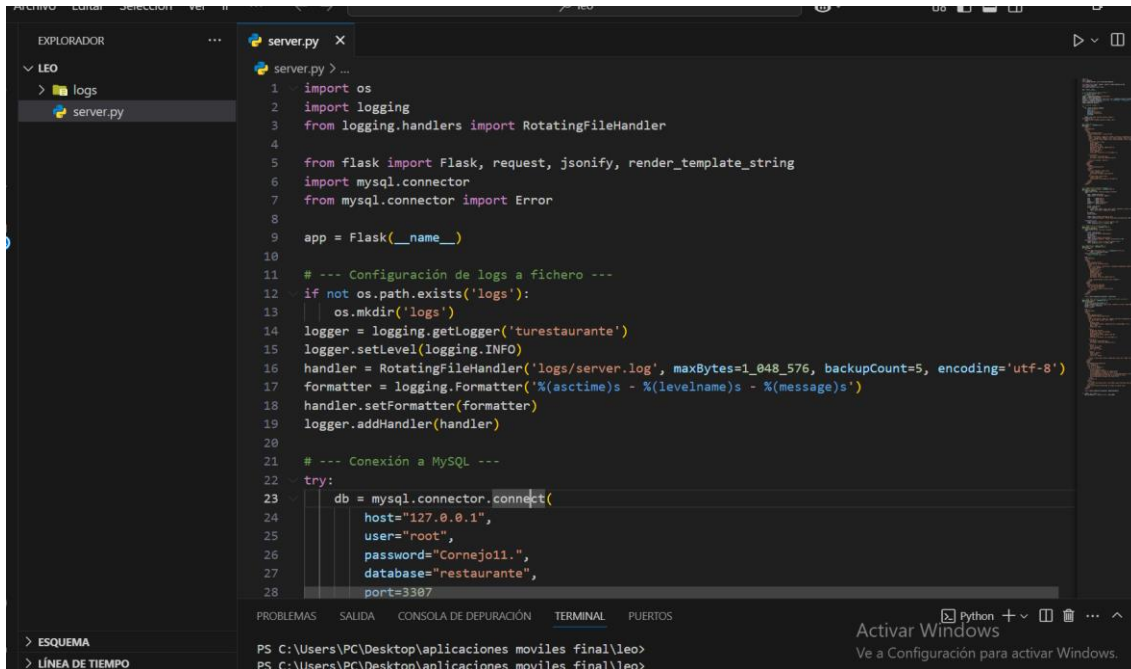
SQL Editor:

```

1 USE restaurante;
2 SELECT * FROM pedidos;
3 CREATE TABLE platos (
4   id INT AUTO_INCREMENT PRIMARY KEY,
5   nombre VARCHAR(100) NOT NULL,
6   precio DECIMAL(10,2) NOT NULL
7 );
  
```

id	mozo	mesa	plato	comentario	precio	fecha
30	Mozo1	1	Lomo saltado	Pedido por Mozo1	114.00	2025-05-13 22:45:02
31	Mozo1	1	lentejas	Pedido por Mozo1	48.00	2025-05-13 22:44:15
32	Mozo1	1	pollo	Pedido por Mozo1	114.00	2025-05-13 22:45:02

XVI. Anexo 4: Código del servidor en Python (server.py)



```

server.py
1 import os
2 import logging
3 from logging.handlers import RotatingFileHandler
4
5 from flask import Flask, request, jsonify, render_template_string
6 import mysql.connector
7 from mysql.connector import Error
8
9 app = Flask(__name__)
10
11 # --- Configuración de logs a fichero ---
12 if not os.path.exists('logs'):
13     os.mkdir('logs')
14 logger = logging.getLogger('trestaurante')
15 logger.setLevel(logging.INFO)
16 handler = RotatingFileHandler('logs/server.log', maxBytes=1_048_576, backupCount=5, encoding='utf-8')
17 formatter = logging.Formatter('%(asctime)s - %(levelname)s - %(message)s')
18 handler.setFormatter(formatter)
19 logger.addHandler(handler)
20
21 # --- Conexión a MySQL ---
22 try:
23     db = mysql.connector.connect(
24         host="127.0.0.1",
25         user="root",
26         password="Cornejo11.",
27         database="restaurante",
28         port=3307
    
```

NOTAS ACLARATORIAS

Consideraciones para formación de equipos

- El número de integrantes de cada equipo debe ser 3 a 5 persona (lo ideal es que la cantidad de integrantes de los equipos sean impares)

Consideraciones para el desarrollo del proyecto

- El proyecto debe terminar con un producto software y/o hardware **implementado alineado al perfil de egreso.**

Consideraciones para el informe y video

1. El Informe de proyecto de fin de curso será redactado con procesador de textos en fuente Arial, tamaño 12, alineación justificada, interlineado 1,5, hoja A4. Utilizar formato APA 7ma edición.
2. Sobre el tema de proyecto
 - ✓ Integración con el aprendizaje previo (temas aprendidos en otros cursos)
 - ✓ Promueve un nuevo aprendizaje
 - ✓ Brinda experiencias realistas
 - ✓ Balance entre complejidad y carga de trabajo
3. Comunicación
 - ✓ Lenguaje profesional
 - ✓ Organización de la presentación
 - ✓ Presentación efectiva
 - ✓ Figuras y formato
 - ✓ Redacción y gramática
4. Video
 - ✓ La presentación de los estudiantes debe ser formal.
 - ✓ Usar office 365 para alojar su video
 - ✓ El video presentado debe contener los puntos indicados en el informe, considerando que se debe cumplir con las indicaciones sobre el tema del proyecto y la comunicación.
 - ✓ El video debe tener una duración máxima de 10 minutos (3min parte teórica y 7min de producto).