



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

**CARRERA DE INGENIERÍA EN ELECTRÓNICA Y REDES DE
COMUNICACIÓN**

**DESARROLLO DE UNA RED DE NODOS DISTRIBUIDOS EN ESTRELLA
BASADOS EN TECNOLOGÍA 802.11ax PARA LA ASISTENCIA INTELIGENTE DE
PEDIDOS EN UN RESTAURANTE**

**TRABAJO DE GRADO PREVIO A LA OBTENCION DEL TITULO DE
INGENIERO EN ELECTRÓNICA Y REDES DE COMUNICACIÓN**

AUTOR: PAÚL ALEJANDRO ZURITA ZAMBRANO

DIRECTOR: MSC. FABIÁN GEOVANNY CUZME RODRÍGUEZ

ASESOR: MSC. JAIME ROBERTO MICHILENA CALDERÓN

IBARRA-ECUADOR

2024



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN

A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
CÉDULA DE IDENTIDAD:	1003197306		
APELLIDOS Y NOMBRES:	Zurita Zambrano Paúl Alejandro		
DIRECCIÓN:	Calle Guayas y Av. 13 de abril		
EMAIL:	pazuritaz@utn.edu.ec		
TELÉFONO FIJO:	-	TELÉFONO MÓVIL:	0996000005

DATOS DE LA OBRA	
TÍTULO:	DESARROLLO DE UNA RED DE NODOS DISTRIBUIDOS EN ESTRELLA BASADOS EN TECNOLOGÍA 802.11ax PARA LA ASISTENCIA INTELIGENTE DE PEDIDOS EN UN RESTAURANTE
AUTOR (ES):	Zurita Zambrano Paúl Alejandro
FECHA: DD/MM/AAAA	11/04/2024
SOLO PARA TRABAJOS DE GRADO	
PROGRAMA:	<input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSGRADO
TITULO POR EL QUE OPTA:	Ingeniero en Electrónica Redes de Comunicación
ASESOR /DIRECTOR:	MSC. Fabián G. Cuzme R.

CONSTANCIAS

El autor (es) manifiesta (n) que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es (son) el (los) titular (es) de los derechos patrimoniales, por lo que asume (n) la responsabilidad sobre el contenido de la misma y saldrá (n) en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 11 días del mes de abril de 2024.

EL AUTOR:



Paúl Alejandro Zurita Zambrano




UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN:

MAGÍSTER FABIÁN GEOVANNY CUZME RODRÍGUEZ, DIRECTOR DEL PRESENTE TRABAJO DE TITULACIÓN CERTIFICA:

Que el presente trabajo de Titulación DESARROLLO DE UNA RED DE NODOS DISTRIBUIDOS EN ESTRELLA BASADOS EN TECNOLOGÍA 802.11ax PARA LA ASISTENCIA INTELIGENTE DE PEDIDOS EN UN RESTAURANTE, ha sido desarrollado por el señor Zurita Zambrano Paúl Alejandro bajo mi supervisión.

Es todo cuanto puedo certificar en honor a la verdad.


MSc. Fabián Geovanny Cuzme Rodríguez
DIRECTOR



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

DEDICATORIA

A mi familia y amigos, pero en especial a mis abuelos quienes no podrán estar conmigo de manera física y siempre fue su sueño verme culminar mi educación superior.

Zurita Zambrano Paúl Alejandro



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

AGRADECIMIENTO

Agradezco a mi familia por haberme apoyado en todo momento y recordarme cada día que las cosas se logran con paciencia, perseverancia y lo más importante, con amor a las cosas que uno hace. A mi novia, Nadia, quien me tuvo toda la paciencia del mundo y me apoyó para poder culminar esta etapa de mi vida. También a su madre, Vivi, por estar presente y presionarme un poco más para terminar este proceso.

También quiero agradecer a mis amigos, Juan T., Cristian R., Erick P., Didier V., Adriana H., Johanna Y. y Mateo T.. Gracias a ellos tuve la oportunidad de vivir muy buenos momentos y sobrellevar los malos brindándonos el apoyo que siempre tuvimos.

Finalmente, quiero agradecer a mi tutor del proyecto por su tiempo, dedicación, comprensión, paciencia y guía durante el desarrollo de este proyecto de titulación.

Gracias a todos.

Índice de Contenidos

CAPÍTULO I: ANTECEDENTES	1
1.1. DEFINICIÓN DEL PROBLEMA	1
1.2. JUSTIFICACIÓN DEL PROBLEMA	2
1.3. OBJETIVOS.....	4
1.3.1. <i>Objetivo General</i>	4
1.3.2. <i>Objetivos Específicos</i>	4
1.4. ALCANCE	4
CAPÍTULO II: FUNDAMENTO TEÓRICO.....	6
2.1. REDES INALÁMBRICAS	6
2.1.1. <i>WLAN</i>	6
2.1.2. <i>Wi-Fi</i>	7
2.1.3. <i>Estándar 802.11</i>	8
2.1.4. <i>Topologías de Red</i>	9
2.1.5. <i>Estándar 802.11ax</i>	13
2.2. MICROCONTROLADORES	15
2.2.1. <i>Software Libre</i>	15
2.2.2. <i>Arduino</i>	16
2.3. MINICOMPUTADORES	16
2.3.1. <i>Aplicación como servidores</i>	17
2.3.2. <i>Servidores Web</i>	17
2.4. RASPBIAN	17
2.4.1. <i>Raspberry Pi</i>	17
2.5. SISTEMAS INTELIGENTES	18

2.5.1. <i>Machine learning</i>	18
2.5.2. <i>Algoritmos de aprendizaje de máquina</i>	18
2.5.3. <i>Tipos de algoritmos de aprendizaje de máquina</i>	19
2.6. INTELIGENCIA ARTIFICIAL	20
2.6.1. <i>Tomas de decisiones en negocios aplicando la inteligencia artificial</i>	21
2.6.2. <i>Beneficio de aplicación de AI</i>	22
CAPÍTULO III: DISEÑO Y DESARROLLO	24
3.1. ANÁLISIS	24
3.1.1. <i>Situación Actual</i>	24
3.1.2. <i>Técnicas de Investigación</i>	26
3.1.3. <i>Descripción y Ámbito del Sistema</i>	27
3.1.4. <i>Descripción General del Sistema</i>	27
3.1.5. <i>Características del Sistema</i>	28
3.2. ESPECIFICACIÓN DE REQUERIMIENTOS.....	30
3.2.1. <i>Identificación de Stakeholders</i>	30
3.2.2. <i>Nomenclatura de Requerimientos</i>	31
3.2.3. <i>Requerimientos de Stakeholders</i>	31
3.2.4. <i>Requerimientos de Sistema</i>	32
3.2.5. <i>Requerimientos de Arquitectura</i>	33
3.3. RECURSOS	35
3.4. DISEÑO DEL SISTEMA	35
3.4.1. <i>Visualización de datos</i>	36
3.4.2. <i>Recepción de datos</i>	37
3.4.3. <i>Procesamiento de datos</i>	38
3.5. SELECCIÓN DE HARDWARE	39
3.5.1. <i>Selección de microcontrolador</i>	39

3.5.2. Selección de pantalla LCD	40
3.5.3. Selección del computador	40
3.5.4. Selección de enrutador.....	41
3.6. SELECCIÓN DE SOFTWARE	42
3.6.1. Sistema operativo para servidor	42
3.6.2. Selección de lenguaje de programación para nodos.....	42
3.6.3. Selección de Framework.....	43
3.7. HARDWARE DESTINADO PARA EL SISTEMA.....	44
3.7.1. Módulo ESP32	44
3.7.2. Pantalla OLED SSD1306	45
3.7.3. Raspberry Pi4	46
3.7.4. Netgear R6700AX.....	47
3.8. DISEÑO DEL SISTEMA	48
3.8.1. Diseño del Hardware.....	49
3.8.2. Diseño de Software	62
3.8.3. Machine Learning	80
CAPÍTULO IV: PRUEBAS DE FUNCIONAMIENTO Y ANÁLISIS DE RESULTADOS	89
4.1. PRUEBAS DE FUNCIONAMIENTO	89
4.1.1. Nodo cliente	89
4.1.2. Servidor	97
4.2. PRUEBAS DE FUNCIONAMIENTO EN AMBIENTE REAL	112
4.2.1. Prueba 1	113
4.2.2. Prueba 2	114
4.2.3. Prueba 3	115
4.2.4. Prueba 4	117
4.2.5. Prueba 5	118

4.2.6. Prueba 6.....	119
4.2.7. Prueba 7.....	121
4.3. ANÁLISIS DE COSTOS.....	125

Índice de Figuras

Figura 1 Topología tipo bus.....	10
Figura 2 Topología de Anillo.....	11
Figura 3 Topología Estrella.....	12
Figura 4 Topología de Árbol.....	12
Figura 5 Topología de malla.....	13
Figura 6 Diagrama de Bloques de sistema.....	36
Figura 7 Módulo ESP32.....	44
Figura 8 Display SSD1306 128x32px.....	45
Figura 9 Raspberry Pi4.....	46
Figura 10 Enrutador Netgear R6700AX.....	47
Figura 11 Arquitectura del sistema.....	48
Figura 12 Diagrama secuencial del nodo cliente.....	50
Figura 13 Diagrama de bloques de nodo.....	52
Figura 14 Diagrama de flujo de nodo para peticiones.....	56
Figura 15 Diagrama de conexión de circuito para nodo cliente.....	57
Figura 16 Diagrama PCB para nodo cliente.....	58
Figura 17 Placa de proyecto terminada y funcionando.....	60
Figura 18 Tapa del nodo cliente en 3D.....	61

Figura 19	Parte inferior del nodo cliente en 3D.....	61
Figura 20	Nodo cliente ensamblado y conectado	62
Figura 21	Diagrama secuencial del sistema.....	63
Figura 22	Opciones de instalación de Codeigniter 4	65
Figura 23	Carpeta de framework obtenida tras instalación	65
Figura 24	Asignación de URL base para proyecto	67
Figura 25	Configuración de archivo de base de datos	67
Figura 26	Rutas para sistema	68
Figura 27	Listado de controladores del proyecto.....	69
Figura 28	Controlador Admin: Función de ingreso.....	70
Figura 29	Controlador Admin.....	71
Figura 30	Controlador Clientes.....	71
Figura 31	Controlador Home.....	72
Figura 32	Controlador Login	73
Figura 33	Controlador Mesa	74
Figura 34	Controlador Orden.....	75
Figura 35	Controlador Producto	75
Figura 36	Controlador Websocket.....	76
Figura 37	Entidad Cliente	77
Figura 38	Lista de entidades del sistema	77
Figura 39	Modelo Cliente	78
Figura 40	Listado de modelos del sistema.....	78
Figura 41	Listado de vistas del sistema	80

Figura 42 Diagrama de flujo para entrenamiento de algoritmo	81
Figura 43 Creación de nuevo cuaderno en google colab	81
Figura 44 Importación de librerías en google colab	82
Figura 45 Datos de entrada para alimentación de sistema.....	82
Figura 46 Lectura de archivos de datos para entrenamiento de red neuronal.....	83
Figura 47 Definición de neuronas y capas para entrenamiento de modelo	83
Figura 48 Compilado de algoritmo	83
Figura 49 Entrenamiento de modelo.....	84
Figura 50 Curva de aprendizaje del modelo	84
Figura 51 Prueba de predicción	85
Figura 52 Guardado de modelo.....	85
Figura 53 Exportación de modelo para JavaScript	85
Figura 54 Diagrama de flujo para implementación de red neuronal	86
Figura 55 Importación de modelo de red neuronal en servidor.....	86
Figura 56 Función para obtener datos del sistema.....	87
Figura 57 Función para predecir venta y mostrar en pantalla.....	88
Figura 58 Prueba de conexión de nodo a red.....	90
Figura 59 Tramas de inicio de comunicación de nodo a servidor	90
Figura 60 Tramas de proceso de autenticación de nodo con servidor	91
Figura 61 Tramas de finalización de comunicación entre nodo y servidor	91
Figura 62 Trama de solicitud de código a servidor con código de autenticación.....	92
Figura 63 Tramas de respuesta con código de verificación para pedido de cliente.....	93
Figura 64 Solicitud de código de verificación para completar orden	93

Figura 65 Envío de petición de asistencia de nodo a servidor.....	94
Figura 66 Trama de ACK de servidor a nodo.....	94
Figura 67 Envío de solicitud de asistencia en mesa.....	95
Figura 68 Envío de notificación de entrega de pedido completo.....	96
Figura 69 Trama de notificación de pedido entregado entre nodo y servidor	96
Figura 70 Pantalla de bienvenida para clientes y administrador.....	98
Figura 71 Ventana de inicio en navegación de vista cliente.....	99
Figura 72 Ventana para realizar pedidos para clientes	100
Figura 73 Carrito de compras con items agregados.....	100
Figura 74 Carrito de compras de cliente.....	101
Figura 75 Sección de facturación para pedido.....	101
Figura 76 Cliente encontrado en base de datos de servidor.....	102
Figura 77 Cliente no encontrado en base de datos de servidor.....	102
Figura 78 Cliente registrado con éxito.....	103
Figura 79 Solicitud de código de verificación y finalización de pedido.....	103
Figura 80 Código de verificación incorrecto	104
Figura 81 Pedido recibido con éxito	104
Figura 82 Enlace a ventana de acceso de administración.....	105
Figura 83 Solicitud de credenciales para ingreso a panel de administración	105
Figura 84 Ingreso denegado a panel de control por credenciales erróneas.....	105
Figura 85 Ventana de inicio de panel de control	106
Figura 86 Creación de nuevo ítem en panel de control	106
Figura 87 Ítem creado en panel de control.....	107

Figura 88	Ítem editado en panel de control	107
Figura 89	Ítem editado y mostrado en panel de clientes.....	107
Figura 90	Panel de control de pedidos para administración	108
Figura 91	Notificación de solicitud de asistencia mostrada en panel de control	108
Figura 92	Observación de pedido realizado por cliente en panel de control	109
Figura 93	Pedido recibido sin procesar.....	110
Figura 94	Pedido en estado de procesamiento en panel de control	110
Figura 95	Pedido en estado de despacho en panel de control.....	111
Figura 96	Pedido redirigido a la ventana de atendidos en panel de control	111
Figura 97	Vista del establecimiento gastronómico del área de clientes	113
Figura 98	Ubicación de enrutador en counter de establecimiento	114
Figura 99	Predicción de venta para el día de prueba	115
Figura 100	Cliente observando pantalla de bienvenida para usuarios.....	115
Figura 101	Cliente realizando pedido	116
Figura 102	Cliente revisando pedido antes de confirmar orden	116
Figura 103	Cliente solicitando código de verificación	117
Figura 104	Cliente atendido tras haber solicitado asistencia con el nodo cliente.....	117
Figura 105	Solicitud de asistencia recibida por servidor	118
Figura 106	Imagen de referencia de distancia de la mesa de cliente	118
Figura 107	Pedido recibido en servidor	119
Figura 108	Cliente 2 navegando en pantalla de bienvenida del sistema.....	119
Figura 109	Cliente 2 siendo atendido por personal tras solicitar ayuda con el nodo.....	120
Figura 110	Cliente 2 realizando confirmación de pedido con código de verificación ..	120

Figura 111 Pedidos atendidos del día de prueba.....	120
Figura 112 Predicción de venta del día siguiente al día de prueba inicial.....	121
Figura 113 Tramas de análisis para tiempos de respuesta	122
Figura 114 Tiempo de llegada de petición de nodo a servidor	122
Figura 115 Tiempo de trama de respuesta a cliente.....	122

Índice de Tablas

Tabla 1 Generación WiFi, estándar relacionado y velocidad de transmisión.....	7
Tabla 2 Listado de Stakeholders	30
Tabla 3 Definición de Nomenclatura para Requerimientos.....	31
Tabla 4 Requerimientos de Stakeholders del Proyecto.....	31
Tabla 5 Requerimientos de Sistema del Proyecto.....	32
Tabla 6 Requerimientos de Arquitectura del Proyecto	33
Tabla 7 Recursos del Proyecto.....	35
Tabla 8 Selección de microcontrolador a implementar en los nodos de cada sensor	39
Tabla 9 Tabla comparativa entre pantallas para mostrar información en nodos cliente.	40
Tabla 10 Selección del minicomputador para implementación con aplicación de servidor	41
Tabla 11 Selección de enrutador para despliegue de red inalámbrica	41
Tabla 12 Selección de lenguaje de programación para nodos	42
Tabla 13 Selección de Framework.....	43
Tabla 14 Características Módulo ESP32	44
Tabla 15 Características de Display Seleccionado para Proyecto	45
Tabla 16 Características de Raspberry Pi4	46
Tabla 17 Características de Enrutador Netgear R6700AX.....	47

Tabla 18	Resultados de pruebas realizadas en ambiente de desarrollo.....	111
Tabla 19	Pruebas en ambiente real	123
Tabla 20	Desglose de costos de implementación del proyecto.....	125
Tabla 21	Etiquetas de prioridad para requerimientos del proyecto	157

Resumen

El presente trabajo de titulación desarrolla un sistema de toma de pedidos inteligente y además ofrece una metodología para poder tener una forma de comunicación en tiempo real con los clientes a través de nodos ubicados en cada mesa y también ofrece una herramienta de predicción de venta que permitirá a los administradores saber que cantidad de sus productos se prevé vender para el día de trabajo. Este dato predictivo no es afirmativo en su totalidad, es nada más una recomendación dado que en la industria de la gastronomía existen muchos factores que tomar en cuenta al momento de analizar una venta.

Utilizando la metodología de investigación en cascada se realizó el proceso de desarrollo de este proyecto, analizando diferentes sistemas gratuitos existentes que cuentan con opciones muy limitadas y algunos extras que requieren de una suscripción de paga para poder acceder a herramientas de predicción y análisis de venta. Además con esta metodología de cascada se desarrollan pruebas por etapas para confirmar el funcionamiento de cada sección del sistema que garantice la convergencia del sistema y sobre todo el sistema se entrega completamente funcional y adaptable para cualquier negocio o emprendimiento gastronómico que desee obtener un sistema de toma de pedidos inteligente y con la característica de que lo hace muy flexible gracias al diseño por etapas y sobre todo es gratuito y no depende de una conexión a internet.

Para el desarrollo de los nodos se establece un dispositivo que utiliza un módulo de desarrollo ESP32 los cuales cuentan con un módulo inalámbrico integrado compatible con la tecnología 802.11 la cual será la base para el despliegue de la red que será desplegada con un enrutador que permita la aplicación del estándar 802.11ax y aprovechar las características de este. En cuanto al sistema, se desarrolló en Linux como sistema operativo base y utilizando una Raspberry Pi4 con el framework llamado codeigniter para el desarrollo de la aplicación web. Para

las pruebas de ambiente real que permitirán comprobar el funcionamiento total del sistema, este se implementó dentro del restaurante Ilapa, ubicado en la ciudad de Ibarra. La implementación del sistema fue exitosa presentando una interfaz para clientes, una interfaz de administración para observar las predicciones de venta que el local podría tener cada día, pedidos y estados de asistencia de las mesas y todo esto con un tiempo de respuesta rápido por parte del sistema utilizando comunicación en tiempo real. Además, la cobertura de la red en todo el local fue adecuada para permitir la comunicación entre los nodos, clientes y usuarios de administración con el servidor de forma inalámbrica.

Abstract

This document develops a system for smart order taking in restaurants. It offers real time communication with the client using nodes located on each table and also has a tool for sales prediction that could be used by managers and owners. The prediction given by the system is not a hundred percent accurate, it's just a reference.

The waterfall methodology has been used for the development of this system with the evaluation of some free systems with limited features for customers and they have some paid plans to give full access to all the features of this systems. The test for each part of the system are designed using this methodology to prove the convergence of the system. This system is free, has the prediction tool, it's adaptable for every restaurant, flexible so people can use the features they want and the most important feature is that this system doesn't need an internet connection.

The nodes designed in this document use an ESP32 development kit compatible with the 802.11 standard. This standard is going to be the base technology for the network deployment. The main router for the network deployment is compatible with the latest standard 802.11ax. About the system, it's developed in Linux using a raspberry Pi4 with the Codeigniter framework for the web application.

The tests for a real environment case were executed in Ilapa's restaurant which is in Ibarra city. The interfaces for customers and managers with sales prediction, assistance status of the tables and orders were successfully implemented with a system with quick response using a real time communication. The network coverage with the router used in this document allows the communication between customers, nodes and managers with the server using the wireless network deployed.

Capítulo I: Antecedentes

En este capítulo se presentarán de manera detallada las bases para el desarrollo del presente trabajo de titulación para justificar la importancia de la ejecución de este proyecto.

1.1. Definición del Problema

Los avances tecnológicos han permitido la evolución de todos los sectores industriales en el mundo, obligando a mejorar a unos y dejando atrás a otros. El internet es uno de estos avances tecnológicos que ha tenido mucha más fuerza desde el inicio de la pandemia porque obligó a las personas a evolucionar y buscar otro medio para poder seguir vendiendo sus productos, dar clases, comunicarse con socios de trabajo, en otras palabras, a continuar permitiendo la comunicación entre seres humanos sin que corran riesgos de contagio (McKinsey, 2020). Uno de los sectores que fue forzado a evolucionar a causa de la pandemia es la industria gastronómica, algunos propietarios de establecimientos gastronómicos lograron digitalizarse e involucrarse en la venta de sus productos mediante de forma virtual, mientras que otros cerraron sus puertas al público (Lopez, 2021). Existen diferentes microcontroladores en el mercado y Raspberry es uno de ellos el cual permite ser programado para diferentes aplicaciones como lo es el caso de Ehsan Raman, quien realizó una aplicación para un restaurante (Hackster, 2017)

Muchas de las alternativas que se pueden encontrar en el mercado para evolucionar en el sector de la gastronomía tienen muchas limitaciones y en otros, su costo es excesivo por su desarrollo de software y adquisición de hardware (Hall, 2018). Además, en este tipo de sistemas existe la dependencia de una conexión a internet para poder tener acceso al servidor, por lo cual, existe el riesgo de perder la conexión con este y dejar el sistema inhabilitado. Otro de los problemas identificados es el acceso al menú por código QR, pese a que su uso en el mundo ha incrementado en un 780% (Lucas, 2021), en el Ecuador no todas las personas cuentan con un plan de datos para

poder acceder a estos enlaces, por lo que un servidor que cuente con un método de presentación de un menú de forma digital se presenta como una solución. Además, al momento de un cliente llegar a un restaurante, no siempre se tiene información o recomendaciones sobre ciertos platos, lo cual puede ser implementado utilizando los datos almacenados en una base de datos generada por el sistema. Por otra parte, se puede generar un análisis predictivo sobre el posible número de clientes que pueda tener un establecimiento acorde al número de platos que se ha vendido en un día. Esta información puede ser útil para generar una recomendación de inventario disponible para el encargado de producción del establecimiento. Estos datos también pueden ser utilizados para generar promociones bajo la autorización del o los propietarios del local.

Para el presente proyecto se plantea el desarrollo de una red que trabaje de manera local con 802.11ax y cuente con servidor que funcione local en conjunto con una red de nodos colocados en cada mesa los cuales estarán conectados con topología estrella utilizando bluetooth. Cada nodo cliente estará conectado con el nodo principal donde se encontrará corriendo el sistema de un restaurante y de esta forma poder ubicar de manera más eficiente cuando una mesa se encuentra ocupada, libre o requiera de alguna atención por parte del personal del establecimiento enviando esta información al servidor y poder tener un mejor control de los clientes. Sobre las recomendaciones de pedidos al ver una carta, existen pocos lugares que manejan una base de datos que permita analizar los diferentes platos que se han vendido y mediante ello poder generar recomendaciones a los clientes que visiten el establecimiento.

1.2. Justificación del Problema

Hoy en día el internet es una herramienta esencial para la comunicación y el entretenimiento de las personas, por lo que, en establecimientos públicos como un restaurante, es importante tener una conexión a internet estable para diferentes fines como el mantener los servicios de facturación electrónica habilitada o permitir a los clientes tener acceso a internet y que

su experiencia de navegación sea grata con una buena velocidad de navegación. Para esto existen equipos de telecomunicaciones que permiten una alta transferencia de datos. Incluso, el ofrecer una red WiFi llama la atención de los clientes y puede ser un punto a favor al momento de decidir si ir o no a un establecimiento gastronómico y más aún sabiendo que la velocidad de internet del lugar es excelente.(Social Hospitality, 2021)

Por otra parte, el tener un sistema inteligente que ayude en la interacción entre cliente-servicio de forma virtual es algo novedoso y debido a la pandemia es un tema que ha avanzado mucho tecnológicamente permitiendo a los usuarios que realicen sus pedidos ellos mismos sin la necesidad de tener la atención física de un miembro del personal del establecimiento. Esto ha permitido la evolución de la forma de interacción de clientes al ingresar a consumir en un restaurante (Hancock, 2021). En el mercado estos sistemas tienen costos mayores a los 1500 usd., pero con este proyecto se pretende realizar un sistema que no cueste más de 800 usd. y pueda presentar opciones de extensión en cuanto a nodos clientes.

Además, el estándar más actual para redes inalámbricas es IEEE 802.11ax, también conocido como WiFi 6. Este ha tenido una gran acogida en diferentes sectores industriales como en el sector turístico, permitiendo que los usuarios de hoteles tengan una experiencia de navegación agradable con una buena velocidad de internet (Binbaum, 2019). En este proyecto se realizará la implementación de una red basada en el estándar IEEE 802.11ax para brindar a los usuarios una mejor experiencia de navegación en internet.

Este documento al formar parte de la universidad será de libre acceso, lo cual permitirá servir como referencia para desarrollos de proyectos similares a este y a su vez permite dejar una guía de un proyecto funcionando que puede ser implementado por usuarios con conocimientos de electrónica y programación para poder aplicarlos en sus emprendimientos.

1.3. Objetivos

1.3.1. Objetivo General

Diseñar una red de nodos clientes basada en el estandar 802.11ax distribuidos en estrella que permita el acceso de manera local a un servidor de toma de pedidos inteligente basado en algoritmos de machine learning aplicado a un restaurante.

1.3.2. Objetivos Específicos

- Realizar recopilación de información bibliográfica sobre comunicaciones inalámbricas relacionadas con los estándares 802.11ax, programación de microcontroladores y sistemas de administración.
- Determinar los requerimientos del sistema acorde al modelo en cascada.
- Diseñar la red estableciendo todas las etapas de diseño del proyecto y comprobación del funcionamiento del sistema.
- Realizar pruebas de funcionamiento en ambientes de operación similares a un restaurante para el que se encuentra diseñado el sistema y que sirvan como respaldo para la demostración del funcionamiento total.

1.4. Alcance

Para el presente proyecto se realizará la recopilación de datos bibliográficos que sirvan de base para desarrollar el proyecto de manera correcta en cuanto a la información correspondiente de programación, comunicaciones inalámbricas, protocolo IEEE 802.11ax, programación de microcontroladores e información sobre sistemas de administración para el desarrollo de la aplicación web de clientes y administración del restaurante.

Por otra parte, la metodología en cascada será la utilizada para la definición de cada etapa que involucra el desarrollo del proyecto, empezando por la parte de análisis, en donde se

establecerán los requerimientos del sistema en base a los resultados de entrevistas realizadas a diferentes propietarios de locales de comida y así realizar la selección de los dispositivos necesarios para poder desarrollar el proyecto utilizando un análisis de benchmark.

Luego viene la etapa de diseño donde se desarrollará los diagramas para la parte de hardware y software del sistema. Para la implementación de software se implementará un algoritmo de machine learning orientado a restaurante y la producción que este necesita tener en base a los consumos que se tenga en dicho establecimiento. Estas estadísticas serán generadas por el sistema propio y se actualizarán periódicamente.

En la sección de implementación es donde se realizará las pruebas de funcionamiento del sistema para corroborar que cumpla con todos los requerimientos obtenidos previamente y para ello se realizará pruebas por etapas lo cual facilitará el descarte de errores.

Para continuar con las etapas finales, se encuentra la sección de verificación donde se demostrará la funcionalidad total del prototipo en un ambiente real de trabajo y se establecerá un manual de mantenimiento en caso de que sea necesario para el sistema desarrollado y garantizar que este funcione por un tiempo prolongado.

Las pruebas de funcionamiento serán documentadas de manera resumida para demostrar el total funcionamiento del sistema una vez que se haya finalizado la etapa de pruebas. Estas pruebas de funcionamiento deben ir cumpliendo los objetivos del proyecto y así satisfacer los requerimientos de los usuarios recopilados previamente. Los entregables del proyecto constan de un servidor que funcione de manera local, es decir, que no dependa de una conexión a internet para su funcionamiento y los nodos clientes que utilicen tecnología para la comunicación con el servidor.

Capítulo II: Fundamento Teórico

Dentro de este capítulo se aborda la base teórica que se ha utilizado para poder llevar a cabo el desarrollo de este trabajo de titulación con sus respectivas fuentes bibliográficas.

2.1. Redes inalámbricas

En 1864, James Clerk Maxwell escribió lo que se conoce como Las Ecuaciones de Maxwell, las cuales consisten en explicar la teoría del electromagnetismo, pero no fue hasta 1888 cuando Heinrich Hertz, físico alemán, logró demostrar estas ecuaciones utilizando una antena dipolo. Así empezaron los experimentos que permitieron desarrollar los sistemas de comunicación inalámbrica inspirando a otros científicos a indagar más sobre dicho tema. En 1895, la antena de Marconi permitió transmitir código morse a través de ondas electromagnéticas. Luego, el uso de las comunicaciones inalámbricas fue orientado para las milicias a nivel global. Tiempo después se permitió su uso comercial permitiendo la aparición de las radios y los canales de televisión. A finales del siglo 20 se establecieron las bandas ISM con el espectro radioeléctrico de 2.4 Gigahertzios (GHz) y la evolución continuó hasta el día de hoy (Itoh, 2013).

2.1.1. WLAN

Wireless Local Area Network por sus siglas en inglés o Red de Área Local Inalámbrica en español. El primer modelo de una red de área local inalámbrica fue establecido por la Universidad de Hawaii conectando a las islas utilizando frecuencias ultra altas (UHF). El nombre de esta red de envío de paquetes era ALOHAnet la cual utilizaba el protocolo Aloha para la transmisión de sus datos. Este protocolo operaba de manera muy sencilla donde, si un dispositivo tenía datos que enviar por la red, los enviaba y si este proceso era ejecutado de manera simultánea con otro envío,

existía una colisión de paquetes provocando la pérdida de estos y con un tiempo aleatorio en cada dispositivo se volvían a enviar los datos hasta que lleguen a su destino (Hans-Dieter, 2021).

2.1.2. Wi-Fi

Este grupo conocido a nivel internacional como Wi-Fi Alliance, se ha encargado desde 1999 en garantizar el funcionamiento de los dispositivos que utilizan tecnología inalámbrica certificando a cada prototipo que ingresa para poder obtener su certificación de que es compatible con la tecnología inalámbrica y con los estándares actuales que la rigen. De esta manera ayudan a garantizar la interoperabilidad de dispositivos en el mercado para una mejor experiencia de los usuarios (Wi-fi Alliance, 2019).

Al igual que los estándares que rigen la tecnología de redes locales inalámbricas, WiFi Alliance tiene sus versiones las cuales van de la mano acorde al estándar al que se homologan los dispositivos. Por ejemplo, el muy mencionado hoy en día Wifi6, se refiere a dispositivos que son capaces de trabajar en base a la norma IEEE 802.11ax que permite altas velocidades de transmisión de datos (Wi-fi Alliance, 2019).

En la figura a continuación se pueden observar las diferentes certificaciones que proporciona wifi Alliance y los estándares relacionados a estas versiones.

Tabla 1
Generación WiFi, estándar relacionado y velocidad de transmisión

Generación/Estándar IEEE	Velocidad de Transmisión
Wi-Fi 6 (802.11ax)	600 – 9608 Mbits/s
Wi-Fi 5 (802.11ac)	433 – 6933 Mbits/s
Wi-Fi 4 (802.11n)	72 – 600 Mbits/s
802.11g	3 – 54 Mbits/s
802.11a	1.5 – 54 Mbits/s

802.11b	1-11 Mbits/s
---------	--------------

Adaptado de: (Steve, 2019)

2.1.3. Estándar 802.11

En 1991, el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) comenzó el proceso de estandarización para las redes inalámbricas y no fue hasta 1997 que apareció la primera versión del estándar que rige hoy las redes inalámbricas IEEE 802.11.

Su primera versión publicada en 1999, IEEE 802.11, permitía una velocidad de transmisión de 2 Megabits por segundo (Mbps) (Hans-Dieter, 2021).

En 1999, apareció la actualización del estándar IEEE 802.11b que permite una velocidad hasta 11 Mbps en la frecuencia de 2,4 GHz. La introducción de dispositivos compatibles con esta tecnología apareció dentro del año 2000 y se pensaba que este sería el estándar definitivo, ya que la primera versión de esta tecnología tenía un alto costo de implementación por las tarjetas inalámbricas. El mismo año se estableció el estándar 802.11a que utilizaba una modulación OFDM para poder alcanzar velocidades de transmisión hasta 54 Mbps, pero la cobertura de la señal no era tan amplia causa de la frecuencia de 5.8 GHz en la cual trabajaba para evitar interferencias con otros dispositivos que utilizaban otra parte del espectro radioeléctrico (Cablefree, 2017).

En junio de 2003, donde se estableció el nuevo estándar IEEE 802.11g, el cual trabaja dentro de la misma frecuencia que su estándar predecesor (2.4 GHz). El mayor beneficio para este estándar es la compatibilidad de hardware con IEEE 802.11b y la aplicación de la modulación OFDM permitió que se alcance velocidades de transmisión de hasta 54 Mbps. Esta actualización permitió que, para el verano de 2003, los equipos que trabajan en doble banda, es decir, con IEEE 802.11a/b, también trabajen ahora en doble banda pero trimodo, en otras palabras, con 802.11a/b/g (Extreme, 2021).

Para octubre de 2009, se ratificó otra enmienda para el estándar IEEE 802.11n que permite velocidades de transmisión desde 54 Mbps hasta 600 Mbps con la implementación de antenas de múltiples accesos y múltiples salidas (MIMO) (Cablefree, 2017).

En finales de 2013, otra idea evolutiva para el documento de las redes inalámbricas de área local surgió. Este fue llamado estándar IEEE 802.11ac, este utiliza canales más amplios en la banda de 5 GHz, modulaciones de orden superior como 256 QAM y la mejora de implementación de múltiples usuarios MIMO (MU-MIMO). Esta actualización permitió alcanzar velocidades de transmisión de 1.3 Gbps en canales de 80 MHz dentro de la banda de 5 GHz (Cablefree, 2017).

El sucesor del estándar IEEE 802.11ac, es el estándar IEEE 802.11ax. Este documento es el más reciente y se encuentra todavía en fases de experimentación, pero ya se encuentra en el mercado y su meta principal es establecer que su tasa de transferencia efectiva, más conocida como throughput, sea de cuatro veces mayor que su predecesor.

2.1.4. Topologías de Red

Una topología de red es un conjunto de dispositivos conectados que cumplen ciertas condiciones para poder establecer esta conexión entre ellos de una forma específica como se detalla a continuación.

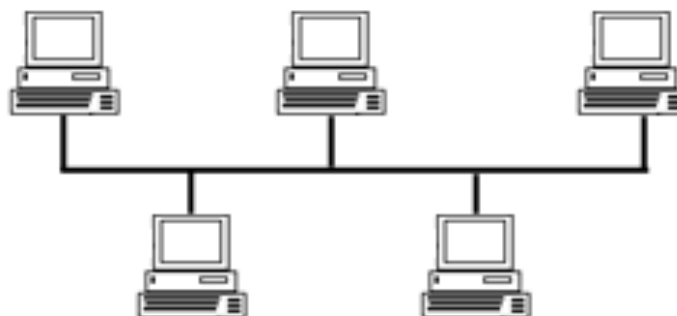
2.1.4.1. Topología de Bus

En este tipo de conexión, todos los dispositivos se encuentran conectados al mismo cable principal el cual será el canal que estarán escuchando constantemente para recibir o enviar datos, este método de acceso al medio se conoce como Carrier Sense Multiple Access.

Cuando un paquete es recibido por un dispositivo, este realiza el proceso de eliminación de cabeceras para confirmar que este paquete este en el destino correcto. En caso de que el paquete no corresponda al destino, el dispositivo descartará toda la información recibida. Si un paquete se llega a perder o llega a chocar con un paquete, existe un tiempo determinado de manera aleatoria

para que se vuelvan a enviar los paquetes por el medio en caso de no haber sido receptados por su destino.

Figura 1
Topología tipo bus



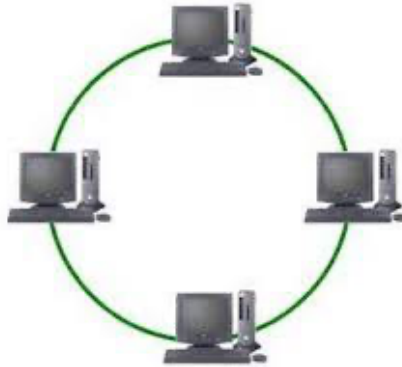
Fuente: (Computer Hope, 2018)

2.1.4.2. Topología de Anillo

Le información utiliza el sentido de reloj para hacer fluir los datos en la red, es decir, los datos del usuario viajan en una sola dirección. Esta topología se asimila a la de tipo bus al compartir el mismo medio de comunicación o canal de transmisión entre todos los dispositivos integrados en la red.

Para poder enviar los datos entre la red, los dispositivos reciben un token el cual cuenta con los datos de la dirección destino y al ser enviada la información, cada equipo revisa este token para comprobar que le corresponda y en caso de que no sea su destino, el paquete viaja de computador en computador hasta llegar a su destino y para confirmar su llegada, el equipo destino envía un paquete de acknowledgment (ACK) hacia su remitente. La mayor desventaja es que si un equipo se sale de la red, todo el anillo se rompe, por lo que esto puede afectar a todos los dispositivos posteriores y la comunicación se perdería en toda la red. (Mejía, 2015)

Figura 2
Topología de Anillo



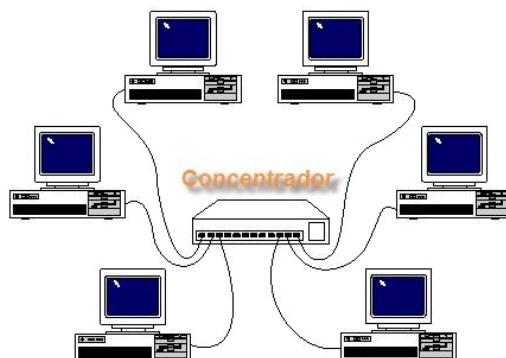
Fuente: (Mejía, 2015)

2.1.4.3. Topología de Estrella

Este método de conexión es el más utilizado en la actualidad al referirse a topologías de red físicas. En esta forma de interconexión de equipos se tiene un nodo concentrador o también conocido como servidor. Este equipo es un conmutador, es decir, su función es recibir todos los datos que envían los equipos y cada uno tiene un puerto asignado para conectarse a la red. De igual forma cada equipo tiene una identificación que es su dirección IP la cual será registrada en una tabla para conocer que equipo está conectado en cada puerto y si se encuentra disponible (encendido y conectado) o no en la red.

Esta topología de red permite actualizar cualquier cambio en ella de manera automática y el envío de paquetes entre dispositivos es mucho más eficiente. En la figura a continuación se puede observar un ejemplo gráfico de una topología en estrella.(BirthLH, 2018)

Figura 3
Topología Estrella



Fuente: (BirthLH, 2018)

2.1.4.4. Topología de árbol

Esta forma de conexión es una combinación entre la topología de bus y la de estrella. Dentro de la topología de árbol se tiene un nodo conocido como la raíz y todas las ramas o los otros nodos tienen un camino para poder llegar a este. Un ejemplo es un servidor de una empresa el cual utilizaría el nodo raíz y las ramas serían todos los concentradores que se encuentren controlando las redes de cada departamento de la empresa. Esta topología tiene la flexibilidad de extensión, detección de errores, división de segmentos y otros beneficios más, pero tienen un alto costo de implementación. (Toro, 2015)

Figura 4
Topología de Árbol

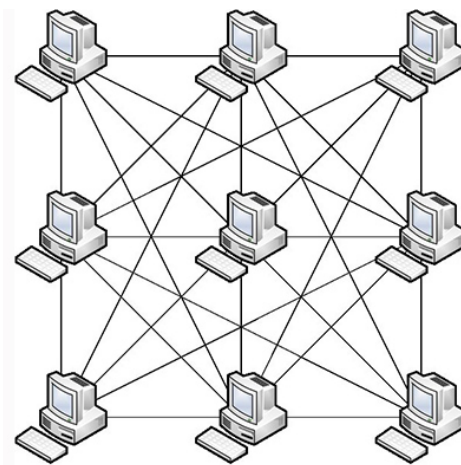


Fuente: (Toro, 2015)

2.1.4.5. Topología de Malla

El termino usado en inglés es Mesh Network. Su característica principal es que cada computador se encuentra conectado con cada dispositivo de la red. Esta conexión entre todos los dispositivos es la característica que la hace esencial para ser aplicada en redes de área extensa (WAN) porque si un nodo queda fuera de servicio, los nodos restantes pueden seguir conectados en la red gracias a los diferentes caminos que esta topología provee. Existe una fórmula que se puede revisar en la documentación técnica de este tipo de topología y permite a los administradores de red saber cuantos cables son necesarios para poder establecer una topología de malla.(Corvo, 2019)

Figura 5
Topología de malla



Fuente: (Corvo, 2019)

2.1.5. Estándar 802.11ax

El avance de la tecnología ha obligado a las personas a evolucionar y aprender a utilizar la tecnología para poder continuar con su vida. Uno de los ejemplos más claros de evolución e implementación de tecnología son los centros comerciales en donde trabajan cientos de empleados y estos requieren de una conexión de internet para mantenerse comunicado o para entretenerse. De igual forma, los visitantes de estos establecimientos pueden llegar a requerir de este servicio, por

lo que la capacidad de transmisión de los equipos debe soportar esta cantidad de datos emitidos por todos los usuarios. Además, hay que tomar en cuenta que con los dispositivos IoT, se suman más clientes y esto significa más consumo de recursos por lo que el Instituto de Ingenieros Eléctricos y Electrónicos junto con la alianza Wi-Fi pusieron en análisis el estándar 802.11ac para poder analizar la posibilidad de mejorarlo. Así nace 802.11ax en el 2018 o también conocido como WiFi6 como recientemente se lo ha nombrado (ArubaNetworks, 2019).

Las mejoras que ofrece este nuevo estándar son explicadas a continuación de manera resumida:

- Aplicando las tecnologías MU-MIMO y OFDMA, permite una transmisión hasta 8 dispositivos simultáneos permitiendo un mejor manejo del flujo de tráfico en la red y una mejor coexistencia con las aplicaciones de transmisión en tiempo real.
- Target Wake Time (TWT) es otra característica de 802.11ax configurable en determinados horarios que permite establecer una negociación con los puntos de acceso (APs) para permitir un ahorro de energía en dispositivos de portátiles mientras estos se encuentran en espera hasta que obtengan el turno de transmisión de sus datos.
- Dentro del internet de las cosas (IoT), 802.11ax establece un canal dedicado para estos, es decir, permite separar a los dispositivos como sensores, interruptores o dispositivos del área médica en una red que funciona con un canal de 20MHz dentro de la banda de 2.4 o 5 GHz según la compatibilidad de los dispositivos.

En un inicio, el propósito principal del desarrollo de esta tecnología fue enfocado a solucionar problemas de ambientes de alta densidad, es decir, de ambientes saturados (más de 20 dispositivos o un alto flujo de tráfico de datos) donde existen una gran cantidad de dispositivos conectados a una red y esta se satura (ArubaNetworks, 2019).

La mayoría de los dispositivos IoT soportan solo la banda de 2.4GHz debido al alto costo de actualización e implementación para estos dispositivos. De esta forma, 802.11ax permite la utilización de la banda de 5GHz para dispositivos que requieran una alta velocidad de transmisión de sus datos. Un par de ejemplos de ambientes de implementación para 802.11ax son:

- Oficinas inteligentes: Todos los dispositivos IoT utilizan la banda de 2.4GHz y la banda de alta velocidad de datos queda libre para el resto de los usuarios.
- Hospitales: Los dispositivos de salud inteligentes se mantienen en la banda compatible con estos mientras que, el hospital, al ser un área de alta densidad, permite tener una mejor navegación gracias a las características de 802.11ax.

Es importante mencionar que el estándar 802.11ax no ha sido lanzado de manera formal, pero el mercado ya ha utilizado y lanzado dispositivos que utilizan esta tecnología al cumplir con una modificación de la capa física de su estándar antecesor (ExtremeNetworks, 2020).

2.2. Microcontroladores

Un microcontrolador es un dispositivo electrónico que se utiliza hoy en día en la mayoría de los dispositivos electrónicos que rodean a todos los seres humanos gracias a la capacidad que tienen de poder realizar tareas específicas ordenadas por el algoritmo que se encuentra cargado en su almacenamiento. La mayoría de estos dispositivos consta de diferentes pines de entrada para poder leer datos y luego poder ser analizados para ejecutar una acción y también pines de salida para dar respuesta u órdenes a otros dispositivos electrónicos como: luces, válvulas, motores, entre otros (SMELPRO, 2020).

2.2.1. Software Libre

Existen diferentes tipos de sistemas operativos los cuales pueden permitir que se realicen cambios dentro del sistema o establecer acciones predeterminadas y limitadas dentro de su

programación. Los programas que permiten establecer cambios a gusto del usuario son conocidos como software libre (Techtarget, 2005).

Como se mencionó anteriormente, los microcontroladores utilizan algoritmos los cuales son programados por los usuarios y estos programas son realizados en plataformas de software libre lo cual permite enviar los códigos fuente para que otros usuarios puedan obtenerlos y editarlos para acoplar dichos algoritmos dentro de otros proyectos.

2.2.2. Arduino

Arduino es una plataforma de código abierto que permite crear a los usuarios algoritmos para implementarlos en microcontroladores. Tanto su software como hardware manejan la política que sea de uso sencillo. Esto ha permitido que aficionados y profesionales de diferentes áreas puedan vivir la experiencia de crear prototipos electrónicos a través de la programación en lenguaje C++ (Arduino, 2018).

El hardware que maneja esta compañía es de fácil acceso para el público y la plataforma de programación (Arduino IDE) es gratis y funciona los sistemas operativos más comunes como Linux, MacOS y Windows (Arduino, 2018).

2.3. Minicomputadores

Un minicomputador cumple con todas las características de una computadora normal pero en menor tamaño, manejando un número menor de microprocesadores a comparación de los computadores convencionales. Una de las aplicaciones principales es como servidores de tamaño medio aplicados en emprendimientos que no contengan muchos usuarios.(Techopedia, 2017)

Otras aplicaciones de estos dispositivos son en escuelas primarias para aprender computación básica o incluso en los dispositivos que utilizamos a diario como una tablet o un teléfono inteligente.

2.3.1. Aplicación como servidores

Los servidores son computadores que permiten ofrecer un servicio adicional para sus clientes y los más comunes son: servicios de almacenamiento, virtualización de servicios, servicios web y entretenimiento.(Rehman, 2018)

2.3.2. Servidores Web

Estos servidores utilizan un protocolo muy aplicado hoy en día para la comunicación final con sus usuarios y es el conocido Hypertext Transfer Protocol (HTTP). Pueden ser expuestos al público con el fin de ser un sitio web informativo o también pueden convertirse en un servicio local para diferentes aplicaciones como el mostrar un menú en un restaurante dentro de su propia red. (MDN Contributors, 2021)

Este protocolo permite mostrar información estática o dinámica, es decir, intercambiar o almacenar archivos (utilizando otros protocolos de red convirtiendo que estos servidores no solo sean un sitio web informativo, sino, una aplicación web. (MDN Contributors, 2021)

Comentar sobre sitios web dinámicos

2.4. Raspbian

Es un software desarrollado basado en Debian y contiene más de 35 000 paquetes de librerías especiales para utilizar en el hardware de Raspberry. Esto hace que el hardware sea aprovechado casi en su totalidad y a su vez permite la compatibilidad con diversos dispositivos y según el modelo de hardware, diferentes aplicaciones como servidores o controladores principales.(Raspbian, 2022)

2.4.1. Raspberry Pi

Su tamaño de hardware es casi de una tarjeta de crédito pero tiene suficiente poder para poder ser utilizado como un servidor pequeño o para jugar. Esta minicomputadora ha sido implementada en todo el mundo con diferentes propósitos como el enseñar a los niños

programación para que entiendan el funcionamiento de una computadora. También se han implementado estos dispositivos como concentradores de datos en redes neuronales gracias a su capacidad de procesamiento y también se han utilizado dentro de sistemas de cámaras y otros proyectos.(Raspberry PI foundation, 2022)

2.5. Sistemas inteligentes

Este tipo de sistemas tienen la característica de poder analizar los datos para poder almacenarlos en variables en una base de datos y así generar un tipo de entrenamiento que permitirá al equipo poder tomar decisiones en ciertos escenarios diseñados por los desarrolladores.

2.5.1. Machine learning

Las sugerencias que aparecen en redes sociales, plataformas de streaming como Netflix o Spotify, son fruto de toda la información que proporcionan los usuarios al navegar en estas plataformas. Las computadoras analizan estos datos y establecen los gustos de los usuarios en las pantallas para captar más su atención. A esto se le conoce como aprendizaje de máquina o machine learning. Es una derivación de la inteligencia artificial y se aplica en diferentes propósitos como: pilotos automáticos, predictores de datos, sugerencias de contenido en plataformas de streaming, domótica, entre otros (A, 2021).

2.5.2. Algoritmos de aprendizaje de máquina

Los algoritmos de aprendizaje de máquina son códigos de programación que permiten realizar ciertos análisis de datos para que los sistemas puedan tomar estos resultados como entrenamiento para su “aprendizaje” y así tomar decisiones en los diferentes escenarios que hayan sido establecidos para su implementación.

2.5.2.1. Aprendizaje supervisado

Este método de aprendizaje utiliza bases de datos como entradas para poder analizarlos y poder ejecutar una tarea en base a esta base de datos ingresada por el usuario estableciendo así los parámetros de trabajo para el algoritmo. Los datos que se utilizan en este tipo de aprendizaje llevan etiquetas para poder los datos de entrada y analizarlos con la base de datos de entrenamiento (IBM, 2021a).

2.5.2.2. Aprendizaje no supervisado

A diferencia del aprendizaje supervisado, utiliza datos sin etiquetar porque la información que se analiza es masiva. Se puede decir que se asemeja a cómo los seres humanos procesan la información para poder reconocer ciertos objetos o también en la parte de correos electrónicos, se implementa el análisis de datos para identificar los correos no deseados (IBM, 2021a).

2.5.2.3. Aprendizaje reforzado

Su metodología consiste en aprender por medio de una prueba y error para poder analizar los datos y en base a esto crea su propia base de datos la cual le servirá para poder identificar qué camino tomar cuando el sistema se encuentre en una situación similar analizada, es decir, no utiliza una base de entrenamiento estática. (BBVA, 2019)

2.5.2.4. Aprendizaje profundo

También es conocido como Deep Learning y es bastante utilizado en las redes neuronales para poder realizar la simulación del trabajo del cerebro humano para poder identificar diferentes situaciones y que el sistema esté preparado para poder tomar decisiones y aprender de estas. Un claro ejemplo del aprendizaje profundo es el reconocimiento de imágenes o de voz implementados en diferentes dispositivos móviles que se encuentran día a día en los usuarios (IBM, 2021a).

2.5.3. Tipos de algoritmos de aprendizaje de máquina

- Algoritmos de Regresión

Los algoritmos de regresión utilizan datos para poder analizarlos y establecer una relación entre ellos. Existen dos clasificaciones para estos y son las siguientes:

- Lineal

La regresión lineal utiliza dos tipos de variables, una dependiente y una independiente para poder predecir el valor de una variable. El ejemplo más básico que se puede brindar de estos algoritmos de regresión lineal es poder predecir una venta anual de un restaurante (IBM, 2021a).

- Logística

Utiliza la base de datos de entrenamiento para poder predecir la presencia o ausencia de una característica similar a una variable del sistema. En otras palabras, utiliza todos los valores de la base de datos para poder establecer una predicción y la condición es que las variables dependientes deben estar relacionadas (IBM, 2021b).

- Árbol de decisiones

Se caracteriza por tener puntos de cruce. En cada uno de estos se establece una prueba para el sistema y en sus subramas se encuentran los posibles resultados de la prueba. Es muy similar a un diagrama de flujo (Grapheverywhere, 2021).

- Redes Neuronales

Esta forma de analizar los datos tiene el fin de trabajar similar al cerebro y las variables que se encuentran en este tipo de aprendizaje de máquina están relacionadas entre ellas a través de sus capas, lo que permite analizar variables más complejas o para desarrollar modelos de relaciones no lineales (Grapheverywhere, 2021).

2.6. Inteligencia Artificial

El ser humano ha ido evolucionando con el paso del tiempo y a su vez su deseo de crear máquinas que faciliten su vida tomando decisiones o ejecutando acciones en diferentes ambientes para eliminar la necesidad de mano de obra humana.

Un ejemplo claro son las líneas de ensamblaje de automóviles se realizaba antes con muchas personas que manejaban diferentes áreas de trabajo, unos se encargaban del chasis, otros de la parte electrónica, un equipo específico para la parte mecánica, etc. Este proceso tardaba demasiado y en tiempos de guerra, los países necesitaban producción de vehículos a gran velocidad para ofrecer transporte de sus tropas, por lo que empezó el desarrollo de máquinas para acelerar el proceso de producción. Después de la guerra, estos procesos fueron adoptados y mejorados por distintas casas de automóviles para acelerar sus procesos de fabricación y reducir la mano de obra humana porque el proceso de soldadura, pintura y transporte de chasis es realizado por máquinas coordinadas entre ellas (GONCALVES, 2016).

Este es un ejemplo de los tantos beneficios que ha proporcionado el desarrollo de inteligencia artificial y se ha aplicado en muchas industrias para mejorar procesos de todo tipo, hasta quirúrgicos.

2.6.1. Tomas de decisiones en negocios aplicando la inteligencia artificial

Los científicos al observar que es lo que pasaba con cada máquina y lo que conlleva su desarrollo, llegaron a la conclusión de que toda actividad se puede traducir a datos y estos datos pueden ser innumerables que un humano no los puede analizar con tanta facilidad o puede que se requiera de muchas personas para poder estar al pendiente de estos datos. En cuanto a la velocidad de cálculo de una computadora es incomparable contra la velocidad de cálculo de un humano, por esta razón son mucho más eficientes.

Acciones tan simples como el encender o apagar ciertos dispositivos según la necesidad del ambiente de trabajo requerían antes de un humano para poder ejecutarlas. Ahora con sensores, una programación en un controlador y algunos elementos adicionales (en caso de ser necesarios), las acciones son más eficientes con una respuesta más rápida.

Unos ejemplos adicionales sobre la aplicación de tomas de decisiones en diferentes negocios son los siguientes:

- La aplicación de chatbots o robots de respuesta automática de soporte para diferentes negocios ayudan a los clientes a solucionar problemas o responder preguntas frecuentes según sea el caso y cuando en realidad se detecte la ayuda de una atención personalizada, dirige el contacto a una persona del equipo de soporte.
- La detección de voz mediante filtros ha permitido a los desarrolladores que ciertas acciones sean ejecutadas si las frecuencias vocales de la persona que brinda el comando concuerdan con las registradas en las bases de datos.
- El análisis biométrico ha permitido generar nuevos mecanismos de seguridad para no permitir el robo de identidad y así garantizar que las operaciones a realizar sean con las personas adecuadas.
- La automatización de casas y edificios es uno de los temas más destacados en cuanto a AI.
- Las recomendaciones en plataformas de streaming de video y audio o también en redes sociales son otro campo donde se encuentra muy desarrollado la IA a causa de la demanda de entretenimiento que existe en el mundo.

2.6.2. Beneficio de aplicación de AI

En la mayoría de los casos, las razones principales para implementar la inteligencia artificial son mejorar la eficiencia y reducir costos operacionales en una empresa o negocio. Además, los campos de aplicación de inteligencia artificial en diferentes ramas empresariales

siguen aumentando por la demanda que existe en el mundo y eso hace más fácil su metodología de implementación en las diferentes ramas de los negocios a nivel global. (NIBUSINESS, 2020)

Algunos de los ejemplos de los beneficios que se puede obtener mediante la implementación de inteligencia artificial son los siguientes:

- Con la automatización de servicios se ahorra tiempo y dinero para realizar rutinas o procesos específicos que no requieren gran complejidad u operación obligatoria de la mano humana.
- En las fábricas permite realizar procesos mucho más rápidos y eficientes permitiendo así la producción y masa de diferentes productos.
- El análisis rápido que ofrecen las computadoras mediante sus procesadores, ayuda a evitar el error que podría ocasionar un empleado humano y gracias a la inteligencia artificial y el aprendizaje de máquina se logra tener una mejor toma de decisiones gracias al análisis a profundidad de diferentes variables mediante los diferentes algoritmos AI.
- Con el uso de los datos proporcionados por los clientes o las personas que se vean involucradas en el ambiente de implementación de AI, se pueden ofrecer estadísticas exactas (monitoreo) que permiten llevar un control de un negocio y además permite observar nuevas oportunidades para continuar con la mejora y desarrollo de nuevas ramas de aplicación para la inteligencia artificial (Pratt, 2021).

Capítulo III: Diseño y Desarrollo

En este capítulo se detalla el proceso realizado para poder desarrollar e implementar el presente proyecto desde los análisis iniciales realizados para obtener los requerimientos de este, la instalación de programas, desarrollo de software para cada etapa del proyecto y las etapas de implementación para comprobar su funcionamiento en diferentes ambientes de prueba.

3.1. Análisis

Previo a la definición de requerimientos, es importante tener un análisis del ambiente donde se desea implementar el proyecto y obtener datos sobre la población seleccionada para el uso del proyecto, establecer las técnicas de investigación a usar para obtener información relevante y así generar los requerimientos mínimos para la ejecución del sistema.

3.1.1. Situación Actual

El desarrollo del trabajo de titulación tendrá como fin el desarrollar un sistema de pedidos inteligente que no requiera de internet, esto permitirá que los propietarios de este sistema trabajar en un entorno local y en caso de perder la conexión internet, el servicio de toma de pedidos no se verá afectado.

Actualmente, los establecimientos gastronómicos que tienen un sistema similar a este corresponden a las cadenas de comida como KFC o McDonald's. Estas cadenas de comida cuentan con equipos de trabajo a nivel global que permiten desarrollar aplicaciones similares a la que se plantea en el presente proyecto. Las características esenciales de estos sistemas son:

- Tener una interfaz amigable y sencilla para los clientes.
- La interacción de dicha interfaz tiene que ser lo bastante clara para evitar el cometer errores.
- Las vistas de la aplicación deben visualizarse sin problema en dispositivos móviles como en computadores.

Estos sistemas han permitido que en países desarrollados el requerimiento de personal para atención al cliente se vea reducido dado que estas técnicas permiten realizar la misma tarea de manera directa, eficiente y a su vez generando un ahorro de gastos al reducir una persona en el equipo de trabajo.

En un estudio realizado en Ambato-Ecuador, se obtuvo como resultado que el 81% de la población ambateña lleva su teléfono celular siempre y obtienen una conclusión de que algunos de ellos no lo llevan por la inseguridad que existe en el país (Luque & Pavlov Raul, 2020).

La información mencionada anteriormente es un respaldo para el desarrollo del proyecto al demostrar que la mayoría de una muestra de la población ecuatoriana lleva siempre su teléfono celular y este dispositivo será el utilizado por los clientes para la solicitud de pedidos de manera directa con el establecimiento gastronómico donde se encuentren.

El ambiente de prueba será un restaurante al aire libre con una amplia extensión. el nombre de este establecimiento gastronómico es Ilapa. Abrió sus puertas al público en el año 2020 y desde un inicio tuvo la idea de implementar tecnología innovadora, pero por falta de recursos y conocimiento sobre el tema recurrieron por la adquisición de un sistema de toma de pedidos gratuito el cual funciona con acceso a internet.

Este sistema de toma de pedidos ha colapsado algunas veces incluso ha generado algunos errores en la toma de pedidos de los clientes y por consecuencia generó discordia entre los usuarios, los propietarios y personal del local. En otras ocasiones, el servicio de internet llegaba a fallar a causa de que los equipos implementados para brindar acceso a internet son para uso doméstico, no para uso público. Esta caída de servicio de internet generaba la pérdida de conexión con el sistema de toma de pedidos y por tanto se tenía que cambiar la forma de tomar la orden de los visitantes. Esto ocupa personal y tiempo se servicio por lo que se generaron algunas quejas, según comenta

el Sr. Israel Muñoz, propietario del establecimiento en una conversación ocasional con el autor del proyecto.

3.1.2. Técnicas de Investigación

3.1.2.1. Investigación Aplicada Tecnológica

Con este método se pretende desarrollar la aplicación del proyecto basada en la observación del funcionamiento de otras aplicaciones similares implementadas o en desarrollo a nivel nacional o internacional y buscando la forma de adaptar este sistema a la realidad de un establecimiento gastronómico.

3.1.2.2. Encuesta

Con la implementación de esta técnica de investigación se espera el recopilar información que permita desarrollar la aplicación de manera correcta y que esta sea adaptable a la necesidad de un establecimiento gastronómico mediante diferentes variables que se identifiquen a lo largo de la investigación.

Los resultados esperados de las encuestas son los requerimientos para el desarrollo del proyecto y entre ellos se lista a continuación algunos de los requisitos para el proyecto que se espera obtener:

- Tiempo de funcionamiento mínimo del sistema
- Conocer sobre la existencia de servicios de conexión a internet contratados por los propietarios de los establecimientos gastronómicos.
- En caso de que exista un contrato con un proveedor de internet, conocer sobre la existencia de problemas de conexión a internet y el periodo de problemas de conexión así como las consecuencias del corte de este servicio.

- También se busca obtener los datos sobre la distancia mínima y máxima entre el punto de venta y las mesas del establecimiento gastronómico. Este dato será relevante para obtener el rango de cobertura máximo de la red a desplegar.
- Otros datos que se busca conocer son la adquisición de sistemas de toma de pedido y así mismo los problemas que se hayan tenido con estos (en caso de que hayan existido).
- Por último se ejecutan preguntas sobre algunas características que creen los encuestados que debe tener los nodos planteados para el diseño del sistema y también las interfaces de usuario cliente y administrador.

Las preguntas se pueden observar dentro del primer anexo en la sección final del documento junto con los respaldos de las encuestas respondidas por los stakeholders seleccionados por el autor del proyecto.

3.1.3. Descripción y Ámbito del Sistema

El proyecto presentado en la propuesta utilizará una red de nodos distribuidos en estrella, es decir, se comunicarán con el servidor mediante un enlace inalámbrico basado en el estándar 802.11ax.

Esta red estará disponible para los clientes y así permitir una conexión de estos con el servidor para poder visualizar el menú del establecimiento gastronómico y de igual forma realizar su pedido.

3.1.4. Descripción General del Sistema

El sistema consta del conjunto de un servidor instalado en un computador o un minicomputador, según sean los recursos disponibles mientras que los nodos utilizarán un

microcontrolador que cumpla con los requerimientos del sistema para poder establecer la conectividad con el servidor.

De igual forma se encontrará habilitada una red inalámbrica para los clientes en donde ellos podrán navegar a través de internet si el establecimiento cuenta con este servicio contratado, caso contrario, los clientes podrán acceder a la aplicación web del restaurante y realizar su pedido una vez que hayan visualizado y analizado el menú de este. Al enviar su requerimiento hacia el servidor, este analizará el estado de la cola de espera de pedidos y definirá un tiempo acorde a la capacidad de preparación de platos que tenga el restaurante.

Los datos que los clientes generen al realizar un pedido serán almacenados en una base de datos para luego analizar cada ítem solicitado y acorde a la demanda que exista de cada plato o bebida generar una estadística de venta para días similares a futuro y de igual forma un pronóstico de venta de platos según la demanda de estos.

3.1.5. Características del Sistema

El proyecto en desarrollo es una red de nodos que se encuentran distribuidos en topología de estrella y se comunican con un servidor de forma inalámbrica, misma comunicación que se habilitará para la comunicación entre clientes con el servidor en donde se encuentra corriendo un sistema que funciona de manera local, es decir, no depende de internet para funcionar y permite que los clientes tengan acceso al menú para poder realizar el pedido en el establecimiento gastronómico que se encuentre.

3.1.5.1. Alcance

- El sistema deberá funcionar por un mínimo de 8 horas sin ningún tipo de caída de servicio.
- La comunicación entre los nodos ubicados en cada mesa y el servidor deberá ser estrictamente inalámbrica utilizando el estándar 802.11.

- La alimentación de los nodos de cada mesa será mediante la implementación de baterías o de una fuente de voltaje con la capacidad suficiente para poder alimentar al circuito por el tiempo mínimo requerido de servicio.
- Los nodos en cada mesa deberán tener un mecanismo que permita mantener una comunicación con el cliente sobre el estado de preparación de su pedido o el tiempo restante para la entrega de este.
- El rango de cobertura mínimo debe ser de 20 metros de distancia entre nodo y servidor.

3.1.5.2. Riesgos

- Algunos módulos son importados directamente para poder reducir un poco de costos, esto puede ser riesgoso dado que si un dispositivo viene dañado de fábrica, la garantía de este no podrá ser ejecutada y en caso de que se pueda realizar, este proceso terminaría demasiado tiempo.
- El enlace de comunicación entre nodos y servidor puede verse afectada en caso de que aumenten los obstáculos de cada establecimiento, lo cual significa tener que comprar más dispositivos que permitan ampliar el rango de red y por tanto, es un aumento de presupuesto para cada cliente.
- Si llega a existir un corte del servicio eléctrico, todo el sistema se vería complicado a menos que se tenga baterías de reserva con las especificaciones necesarias para poder alimentar a un enrutador y al servidor, pero esto significa más inversión para cada propietario.
- El diseño de los nodos ubicados en cada mesa no es a prueba de derrame de líquidos, por lo que si en un ambiente de prueba, se llegara a derramar algo sobre el dispositivo, este tiene el riesgo de dañarse.

3.1.5.3. Restricciones

- El sistema no va a entregar un sistema de inventario para los establecimientos gastronómicos.
- El sistema no estará diseñado para funcionar en caso de que la energía eléctrica total se vaya.
- La sección de facturación no estará implementada en este sistema.

3.2. Especificación de Requerimientos

Los dos factores más importantes del proyecto son: el servidor y los nodos. Por esta razón, es necesario seleccionar los dispositivos que permitan el funcionamiento del proyecto utilizando el estándar 802.11ax. Para esto se debe llevar a cabo un análisis de requerimientos de usuario, sistema y arquitectura para poder llevar a cabo el desarrollo de la propuesta planteada.

3.2.1. Identificación de Stakeholders

La definición de stakeholder dentro de un proyecto es un usuario o entidad que puede ser un beneficiado o afectado de forma directa o indirecta por las actividades que se realicen para cumplir con el desarrollo del proyecto presentado.

Tabla 2
Listado de Stakeholders

Stakeholders		
ID	Nombre	Institución
1	Sr. Paúl Zurita Z.	Universidad Técnica del Norte
2	MSC. Fabián Cuzme R.	Universidad Técnica del Norte
3	MSC. Jaime Michilena C.	Universidad Técnica del Norte
4	Sr. Israel Muñoz	Restaurante Ilapa Bar & Grill

5	Chef Bismark Paspuel A.	Restaurante El Patio Pizza, Bar & Grill
6	MSC. Pablo Proaño P.	Asesor Gastronómico
7	Lic. Nadia Garzón	Asesora y Administradora Gastronómica

3.2.2. Nomenclatura de Requerimientos

Como se había mencionado en el punto 3.2, los requerimientos se basan en usuario, sistema y arquitectura para lo cual se establecen nomenclaturas para identificar cada requerimiento.

Tabla 3
Definición de Nomenclatura para Requerimientos

Requerimiento	Nomenclatura
Stakeholders	HRS
Sistema	SRS
Arquitectura	ARS

3.2.3. Requerimientos de Stakeholders

Es importante tomar en cuenta la opinión de las personas y entidades que se vean afectadas, beneficiadas o puedan brindar una recomendación para el proyecto

Tabla 4
Requerimientos de Stakeholders del Proyecto

Requerimientos HRS					
#	Requerimiento	Prioridad			Relación
		Alta	Media	Baja	
Requerimientos Operacionales					
HRS1	El tiempo mínimo de funcionamiento del sistema debe ser 8 horas.		X		
HRS2	La comunicación entre los nodos clientes y el servidor deben ser de forma inalámbrica.		X		

HRS3	Los nodos utilizarán una fuente de voltaje externa para su alimentación.	X
HRS4	El rango de cobertura entre servidor y nodo debe ser máximo 20 metros.	X
HRS5	La conexión entre el servidor y el enrutador debe ser fiable posible.	X

Requerimientos de Usuario

HRS7	Los nodos deben contar con un método de envío de solicitud de servicio personalizado al servidor.	X
HRS8	Los nodos deben soportar condiciones externas causadas por los usuarios.	X
HRS9	La interfaz para clientes debe ser fácil de manejar	X

3.2.4. Requerimientos de Sistema

Los requerimientos funcionales son un punto de partida para definir los requerimientos de sistema dado que estos son más detallados. La siguiente tabla enlista los requerimientos de sistema para el desarrollo del proyecto.

Tabla 5
Requerimientos de Sistema del Proyecto

Requerimientos SRS					
#	Requerimiento	Prioridad			Relación
		Alta	Media	Baja	
Requerimientos de Sistema					
SRS1	Debe ser apto para trabajar por un periodo mayor a la jornada laboral común	X			HRS1
SRS2	La alimentación de los nodos debe manejar un voltaje de 3.3V a 5V	X			

SRS3	Los datos de clientes y nodos deben ser guardados en una base de datos.	X	
SRS4	La conexión de los nodos al servidor debe utilizar el estándar 802.11xx	X	HRS2
Requerimientos de Interfaz			
SRS5	Sencilla y amigable para los usuarios	X	HRS9
Requerimientos de Uso			
SRS6	El servidor debe ser implementado en un sitio con flujo de aire para una correcta ventilación y fácil acceso para mantenimiento.	X	
Requerimientos Físicos			
SRS7	Los nodos deben ser compactos, estéticos, soportar salpicaduras de líquido.	X	HRS8

3.2.5. Requerimientos de Arquitectura

El hardware y software deben estar en armonía para obtener un correcto funcionamiento del sistema, por esta razón se listan a continuación los requerimientos de arquitectura para la propuesta planteada.

Tabla 6
Requerimientos de Arquitectura del Proyecto

Requerimientos ARS					
#	Requerimiento	Prioridad			Relación
		Alta	Media	Baja	
Requerimientos Lógicos					
ARS1	La comunicación de los nodos debe ser estrictamente inalámbrica para enviar y recibir datos del servidor.	X			HRS2
Requerimientos de Diseño					

ARS2 El costo para desarrollo del proyecto debe mantenerse en un valor menor a 1000 usd. X

Requerimientos de Hardware

ARS3 El servidor deber tener una memoria RAM mínima de 4GB. X

ARS4 El servidor debe contar con al menos 2 puertos USB para conectar dispositivos periféricos. X

ARS5 El servidor debe tener puertos de conexión ethernet y módulo Wi-Fi para adaptarse a distintos ambientes de implementación X

ARS6 Los nodos deben contar con módulo inalámbrico Wi-Fi. X HRS2

ARS7 Disponibilidad de los dispositivos en el mercado con un precio accesible. X

Requerimientos de Eléctricos

ARS8 El voltaje de trabajo para los nodos debe ser entre los 3.3V a 5V. X SRS2

Requerimientos de Software

ARS9 Deben existir librerías y documentación que sirvan de soporte para el desarrollo del proyecto X

ARS10 Las plataformas de programación deben ser de código abierto. X ARS9

ARS11 El sistema debe ofrecer datos de predicción de venta para ofrecer una estadística al administrador del establecimiento gastronómico. X

3.3. Recursos

Dentro del desarrollo del trabajo de titulación se debe tomar en cuenta las personas involucradas dentro de la toma de decisiones que permitan llegar a cumplir los objetivos planteados en la propuesta del proyecto.

En la siguiente tabla se enlistan las personas relacionadas al desarrollo del proyecto de titulación.

Tabla 7
Recursos del Proyecto

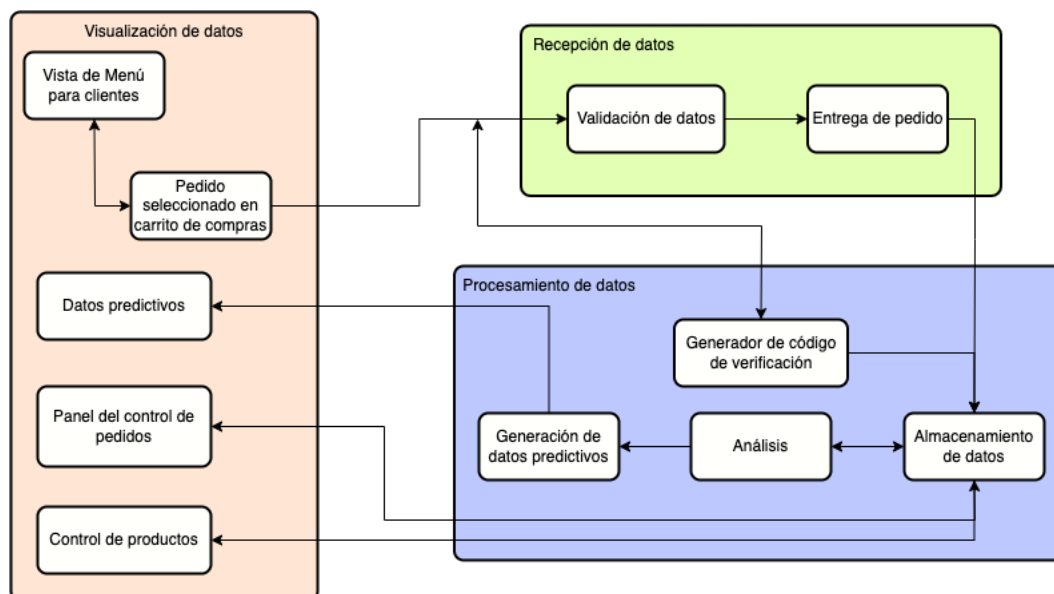
HUMANOS	
Tutor	MSC. Fabián Cuzme R.
Desarrollador	Sr. Paúl Zurita Z.

3.4. Diseño del Sistema

Después de haber realizado el análisis de los requerimientos para poder desarrollar el proyecto, los cuales tienen su respaldo dentro de la sección de anexos, se procede con la fase de diseño del sistema. Esta sección consiste en detallar en este documento el proceso que se realiza para poder definir las etapas del proyecto previo a su implementación.

Se puede observar en la siguiente figura un diagrama de bloques general de cómo funciona el sistema para luego entrar más en detalle sobre cómo funciona cada bloque.

Figura 6
Diagrama de Bloques de sistema



3.4.1. Visualización de datos

Este bloque permite visualizar la información para los clientes y administradores en diferentes ventanas según sea la pantalla seleccionada

Para los clientes, los campos de visualización serán:

- Vista de menú para clientes: Página de inicio para observar detalles del local gastronómico, menú, historia e imágenes referenciales de platillos.
- Pedido seleccionado en carrito de compras: En otra ventana se podrá observar el menú pero esta vez se podrá seleccionar los ítems del pedido para agregarlos a un carrito de compras.

El siguiente paso es enviar el pedido pero para ello se complementa con el bloque de Generador de código de verificación para continuar con el bloque de recepción de datos que se explicará más adelante.

En cuanto al panel de administración se pueden observar los bloques:

- Datos predictivos: Esta información se recibe por parte del bloque de generación de datos predictivos. Este se explicará más adelante.
- Panel de control de pedidos: Esta información mantiene la conexión con el pedido que se realice por parte de los clientes y se puede controlar el estado de este pedido para colocarlo en estado de preparación, despachado o cancelarlo. Mantiene una comunicación con la base de datos para actualizar los pedidos en ella. La acumulación de pedidos generará un dato de venta total que se almacenará en la base de datos para analizar a futuro y generar las predicciones de venta.
- Control de productos: Dentro de la industria gastronómica los precios pueden llegar a variar y para ello esta pestaña ofrece la opción de mantener un control de estos para editar, agregar o eliminar. Este proceso mantiene una conexión con la base de datos constante para actualizar de forma automática el menú para realizar el pedido.

3.4.2. Recepción de datos

El primer paso es dado por el cliente quien con su teléfono móvil ingresará al servidor del sistema en donde se encuentra la pantalla de bienvenida con información del local así como el menú para realizar el pedido.

Una vez que se haya visualizado el menú, el cliente procede a la ventana de pedidos para agregar los platos o bebidas que desee para ese momento y una vez finalizado esto procede a confirmar su compra en el carrito de compras para una última revisión de su pedido.

Para finalizar el envío del pedido, el sistema solicitará al cliente que presione un botón del nodo ubicado en su mesa para poder generar un código de verificación que generará el servidor de manera automática y aleatoria con un tiempo de caducidad, esto es para prevenir ataques y la generación de falsos pedidos.

Una vez ingresado el código de verificación que mostrará el nodo en la mesa, el servidor recibirá este dato y lo confirmará con su base de datos para dar el paso final de envío de datos al sistema, almacenarlo en la base de datos y mostrar el pedido en la sección de panel de control de administración.

3.4.3. Procesamiento de datos

En la sección del servidor se cuenta con una base de datos, la cual cumple con la función de almacenar toda la información permanente y temporal del sistema.

Los datos permanentes son:

- Productos ofrecidos por el establecimiento
- Registro de ventas
- Registro de nodos suscritos al sistema
- Usuarios y contraseñas para administradores
- Registro de órdenes

Los datos temporales son:

- Códigos de autenticación de nodos
- Códigos de verificación de pedidos según cada nodo.

En esta sección también se cuenta con el generador de código de verificación. Este es un número aleatorio de 6 dígitos generado por el servidor y no puede repetirse a menos que haya caducado su tiempo de vida para poder ser utilizado nuevamente. Este código generado será el complemento para el bloque de validación de datos.

El bloque de análisis tomará la información de la base de datos con respecto a los registros de venta para poder ejecutar el algoritmo de aprendizaje y así complementarse con el bloque de generación de datos predictivos para poder entregar al administrador estos datos obtenidos.

3.5. Selección de Hardware

En esta sección se determinará que dispositivos son los mejores para el desarrollo del proyecto basados en los requerimientos obtenidos en la sección 3.5. Esta selección se realizará estableciendo tablas comparativas entre diferentes equipos que cumplan con ciertas características de los requerimientos y así mismo el que cumpla todas será el seleccionado. Cuando un requerimiento sea cumplido se llenará la casilla con un 1 y con 0 si el requerimiento no es cumplido.

3.5.1. Selección de microcontrolador

El microcontrolador seleccionado será el encargado de procesar los datos de transmisión y recepción con el servidor para funciones específicas del sistema como el entregar un código de verificación y solicitar ayuda al servidor para que un miembro del personal se acerque hacia la mesa de los clientes.

Tabla 8
Selección de microcontrolador a implementar en los nodos de cada sensor

Dispositivo	Requerimientos							Valoración
	HRS4	SRS2	SRS4	SRS7	ARS6	ARS7	ASR8	
ESP32	1	1	1	1	1	1	1	7
Arduino NANO	0	1	0	1	0	1	1	4

Cumple: "1" No Cumple: "0"

Elección: ESP32

En la tabla anterior se observa la comparativa realizada entre dos microcontroladores, ESP32 de la empresa ESPRESSIF y Arduino NANO, los 2 dispositivos son utilizados en proyectos como microcontroladores por su buena capacidad de procesamiento y sobre todo por su tamaño

compacto. De estos dos dispositivos, el que cumple todos los requerimientos es el ESP32 al tener un módulo Wi-Fi integrado. Las características adicionales de este dispositivo se muestran en la siguiente sección.

3.5.2. Selección de pantalla LCD

Esta pantalla se implementará en el módulo del nodo ubicado en cada mesa para mostrar el código de verificación del pedido que deberá ingresar el cliente para finalizar su orden. La tabla a continuación detalla la comparativa entre los dispositivos considerados para poder realizar esta etapa del proyecto.

Tabla 9

Tabla comparativa entre pantallas para mostrar información en nodos cliente

Dispositivo	Requerimientos							Valoración
	HRS9	SRS2	SRS5	SRS7	ARS9	ARS7	ASR8	
LCD 16X2	0	0	1	0	1	1	1	4
OLED SSD1306	1	1	1	1	1	1	1	7
Cumple: “1” No Cumple: “0”								
Elección: OLED SSD1306								

La pantalla seleccionada cumple con la función de ser lo más estética y amigable para la presentación de datos a los clientes en las mesas. Los detalles de este dispositivo se mencionan en la sección 3.8 del documento.

3.5.3. Selección del computador

En caso de que el lugar de implementación no cuente con un computador propio para instalar todos los paquetes del servidor, se ha propuesto el implementar un computador o minicomputador para la instalación del servidor que correrá de forma local en el sistema.

La tabla a continuación expone las opciones de dispositivos considerados para el desarrollo del sistema.

Tabla 10
Selección del minicomputador para implementación con aplicación de servidor

Requerimientos								
Dispositivo	HRS1	SRS1	ARS2	ARS3	ARS4	ARS7	ASR8	Valoración
Laptop corei5	1	1	0	1	1	0	0	4
Raspberry PI4	1	1	1	1	1	1	1	7
Onion Omega2+	1	1	1	0	1	0	1	5
Cumple: “1” No Cumple: “0”								
Elección: Raspberry Pi4								

El raspberry pi4 es un minicomputador que cumple con todos los requisitos propuestos en el proyecto por lo que lo hace la mejor opción para aplicación de servidor. Las características de este dispositivo se muestran en la sección 3.8.

3.5.4. Selección de enrutador

Es importante que el dispositivo que se seleccione en este caso cumpla con todas las características mínimas y que sobre todo tenga la capacidad de trabajar con el estándar 802.11ax.

La tabla a continuación muestra los dispositivos a considerar y su selección.

Tabla 11
Selección de enrutador para despliegue de red inalámbrica

Requerimientos							
Dispositivo	HRS1	SRS1	SRS4	ARS2	ARS7	HRS4	Valoración
Netgear R6700AX	1	1	1	1	1	1	6

TP-link	0	1	1	1	1	0	4
AX1500							
Reeyee	1	1	1	0	0	1	4
AX3200							

Cumple: “1” No Cumple: “0”

Elección: Netgear R6700AX

Realizando una revisión un poco más extensa sobre los comentarios de los usuarios, se llegó a la conclusión de que el enrutador seleccionado es el más apto en cuanto a la relación costo-beneficio. Las características de este dispositivo se presentan en la sección 3.8 correspondiente.

3.6. Selección de Software

Para la selección de software se ha realizado un análisis de algunos parámetros como el software a utilizar para la implementación del servidor, el software de programación para los nodos y también el framework que se va a utilizar para el desarrollo del servidor.

3.6.1. Sistema operativo para servidor

El computador seleccionado en la tabla 10 tiene ya un sistema operativo Debian instalado por defecto y este es conocido como Raspbian. Este sistema está basado en Linux, es decir, es de código abierto y permite realizar todas las funciones necesarias para el desarrollo del proyecto.

3.6.2. Selección de lenguaje de programación para nodos

La tabla a continuación muestra la comparación de los lenguajes de programación posibles para configurar los nodos clientes que se ubicarán en cada mesa.

Tabla 12
Selección de lenguaje de programación para nodos

Requerimientos			
Tipo	ARS9	ARS10	Valoración

C++(Arduino IDE)	1	1	2
Python	1	1	2
Cumple: “1” No Cumple: “0”			
Elección: C++(Arduino IDE)			

El nodo ESP32 soporta los dos lenguajes de programación, pero en cuanto a respaldo de documentación y proyectos relacionados, existe más información referente al IDE de Arduino, es decir, el lenguaje seleccionado de programación es C++.

3.6.3. Selección de Framework

Un framework es una plantilla de una aplicación prediseñada que cuenta con librerías instaladas y permiten acelerar el proceso de desarrollo de aplicaciones. La tabla siguiente indica la comparativa entre dos frameworks populares los cuales son los más usados para el desarrollo de aplicaciones web y la selección de uno de estos para el desarrollo del proyecto.

Tabla 13
Selección de Framework

Requerimientos				
Tipo	SRS5	ARS9	ARS10	Valoración
Angular	0	1	1	2
Codeigniter	1	1	1	3
Cumple: “1” No Cumple: “0”				
Elección: Codeigniter				

Codeigniter es un framework completo que permite utilizar la base de datos directamente, además, contiene un controlador específicamente para realizar la conexión directa hacia una base

de datos. Por otra parte, Angular posee es un framework más para diseño pero para utilizar base de datos se requiere de una extensión adicional. Por esta razón el framework seleccionado es Codeigniter.

3.7. Hardware Destinado Para el Sistema

Este apartado describe los elementos seleccionados en la sección 3.7 para la implementación y desarrollo del sistema propuesto en este proyecto. A continuación se enlista el hardware seleccionado previamente con sus características principales.

3.7.1. Módulo ESP32

Figura 7
Módulo ESP32



Fuente: (Electrostore, 2021)

La figura anterior es una ilustración de ejemplo de cómo se ve el microcontrolador ESP32. Este cuenta con diferentes versiones de presentación, pero la que se va a usar para este proyecto es la que se observa en la figura anterior. La tabla a continuación muestra las características principales de funcionamiento de este dispositivo.

Tabla 14
Características Módulo ESP32

Parámetro	Características
Tensión de alimentación	3.3-5V
Protocolos soportados	802.11 b/g/n - 802.15

Soporte de red	2.4 GHz
Banda	2400 MHz
Procesador	32 bits
Potencia de salida	0.15 W
Consumo en modo de baja energía	10 uA
Modos de funcionamiento	AP, STA, STA + AP
Comunicación interfaz	SPI, Serial, UART
Tamaño	51mm x 23mm x 8mm

Fuente: (Electrostore, 2021)

3.7.2. Pantalla OLED SSD1306

Figura 8
Display SSD1306 128x32px



Fuente: (Ashraf Said, 2022)

La figura anterior muestra la ilustración de referencia de la pantalla que se va a implementar dentro de los nodos para mostrar datos como: logotipo del establecimiento, códigos de verificación de pedido y mensajes de configuración inicial.

La tabla siguiente detalla las características principales de este dispositivo.

Tabla 15
Características de Display Seleccionado para Proyecto

Parámetro	Características
Tensión de alimentación	1,6-3.3V
Resolución	128x32 pixels
Pines de transmisión y recepción	2
Pines de alimentación	Vcc y GND
Protocolo Soportado	I2C

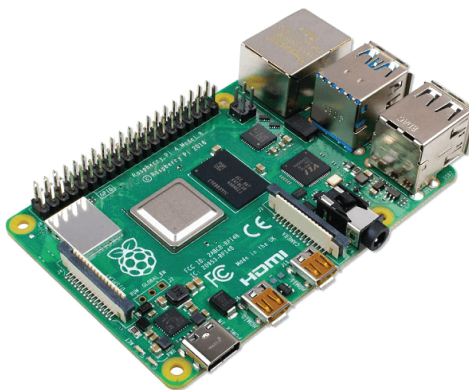
Tamaño

51mm x 23mm x 8mm

Fuente: (Ashraf Said, 2022)

3.7.3. Raspberry Pi4

Figura 9
Raspberry Pi4



Fuente:(RaspberryPi, 2022)

La imagen anterior muestra de manera gráfica cómo se ve la placa del minicomputador seleccionado para el alojamiento y desarrollo del sistema propuesto en este documento y la tabla a continuación detalla las características esenciales de este dispositivo.

Tabla 16
Características de Raspberry Pi4

Parámetro	Características
Procesador	BCM2711/ Quad Core ARMv8
Módulo Inalámbrico	802.11 b/g/n/ac - 802.15 BLE
Puertos USB	2 USB 2.0 y 2 USB 3.0
Puertos extra	2 puertos HDMI
Procesador	64 bits
Tensión de alimentación	5V
Temperatura de trabajo	0-50 grados Celsius

 Fuente:(RasperryPi, 2022)

3.7.4. Netgear R6700AX

Figura 10
Enrutador Netgear R6700AX



Fuente:(Netgear, 2021)

En la figura anterior se puede observar una ilustración de ejemplo de cómo es del enrutador seleccionado para en el despliegue de la red de este sistema y en la tabla a continuación se pueden observar las características que destacan de este dispositivo.

Tabla 17
Características de Enrutador Netgear R6700AX

Parámetro	Características
Protocolo de Red	802.11ax – Compatible con anteriores
Seguridad	Control de Acceso y control parental,
Área de cobertura	130 metros cuadrados
Velocidad Máxima de Transmisión	1.8 Gbps
Puerto de red	Gigabit Ethernet

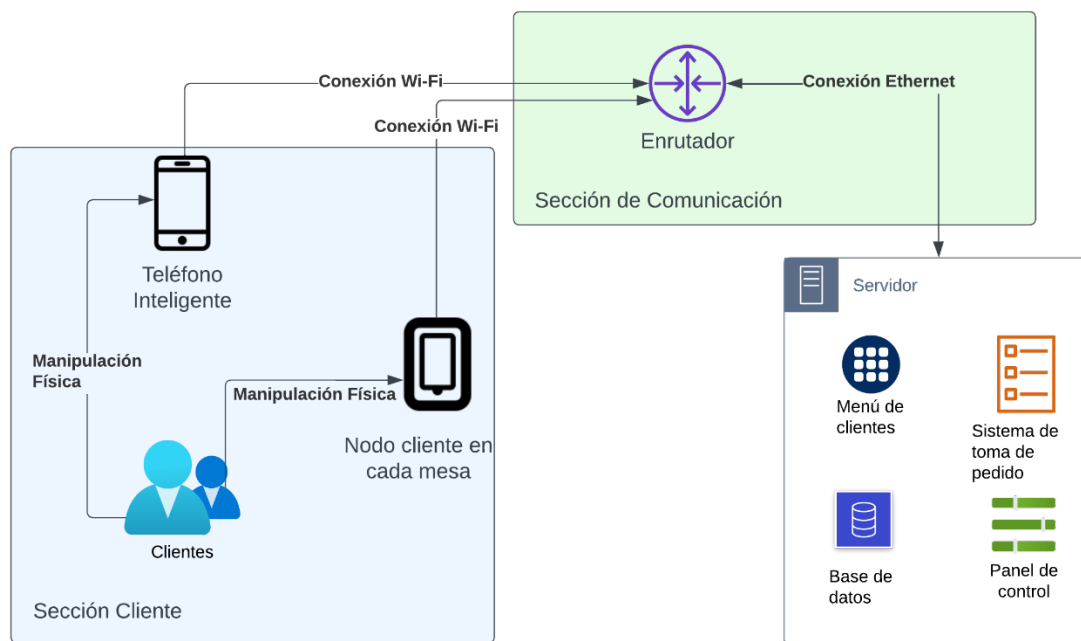
Fuente:(Netgear, 2021)

3.8. Diseño del Sistema

Luego de haber seleccionado y también haber analizado el hardware y software disponibles y compatibles con los requerimientos recolectados del proyecto se procede con la etapa de diseño en donde se tiene como objetivo el construir el sistema cumpliendo todos los requerimientos planteados previamente para garantizar el funcionamiento total del sistema.

En la figura a continuación se muestra la arquitectura del sistema y posteriormente una breve descripción de esta.

Figura 11
Arquitectura del sistema



Como se observa en la imagen anterior, existen una serie de procesos y todos apuntan hacia el servidor como punto final para realizar distintas acciones. Dentro de la sección 3.4 se encuentra el diagrama de bloques del funcionamiento del sistema.

En esta figura, se puede observar 3 secciones correspondientes a cliente, comunicación y servidor.

Dentro de la sección cliente se tiene la manipulación física por parte de los usuarios hacia el teléfono inteligente con el cual podrán acceder ha sido el servidor mediante una conexión inalámbrica a una red Wi-Fi. De igual forma los clientes podrán realizar una manipulación hacia los nodos ubicados en cada mesa para poder solicitar el código de verificación para completar el pedido. Estos nodos utilizan una conexión inalámbrica Wi-Fi con el servidor.

En la sección de comunicación se encuentra el enrutador que funciona con el protocolo 802.11ax y todas sus versiones anteriores para dispositivos antiguos. Este dispositivo cumplirá con una función de permitir la conexión entre los clientes y nodos con el servidor. El enlace de comunicación entre este equipo y el servidor será cableado para mantener la fiabilidad máxima de comunicación.

Finalmente se tiene al servidor el cual cumplirá con la recepción de todos los datos enviados por los clientes, nodos y usuarios de administración que tengan acceso para manipular la información almacenada dentro de este.

3.8.1. Diseño del Hardware

Para el desarrollo de los nodos cliente que se van a ubicar en las mesas se utiliza el microcontrolador ESP32 con el armazón para desarrollo (development kit) junto con el display que utiliza el protocolo I2C para la comunicación con la placa y como extras se implementarán 3 pulsadores en configuración pull-up con resistencias de 330 Ohms para evitar la lectura de rebotes en el pin de entrada del microcontrolador.

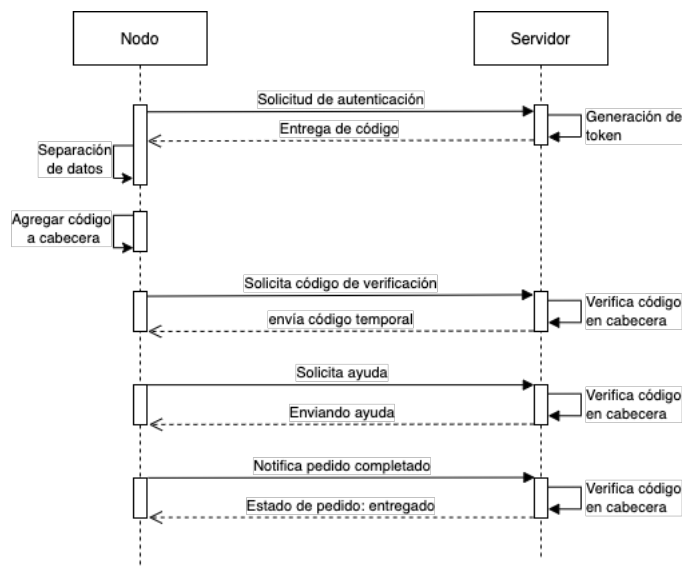
El protocolo I2C o también conocido como interfaz de 2 cables (TWI) utiliza dos pines nada más para establecer una comunicación entre dos dispositivos. El primero se encarga de mantener la señal de reloj para la comunicación serial (SCL) Y el segundo se encarga de transferir los datos por la comunicación serial (SDA). este protocolo cuenta con dos modos de operación los cuales pueden ser maestro o esclavo y el envío de los pulsos se producen a través de la señal de

reloj configurada en el pin SCL, Pero si por alguna razón el pin de la señal de reloj se encuentra alto la información enviada por el pin de datos SDA no tendrá lectura. Este protocolo maneja transmisiones de tipo half-duplex, es decir, envío y recepción de datos pero no de forma simultánea y una de sus desventajas es que solo puede usarse dentro de comunicaciones de baja velocidad. (Ajitgupta Prerna, 2021)

3.8.1.1. Diagramas del Nodo Cliente

El nodo cliente tiene un diagrama secuencial para la ejecución de sus acciones. Este diagrama se observa en la figura siguiente y servirá de referencia para realizar las pruebas de funcionamiento del nodo.

Figura 12
Diagrama secuencial del nodo cliente



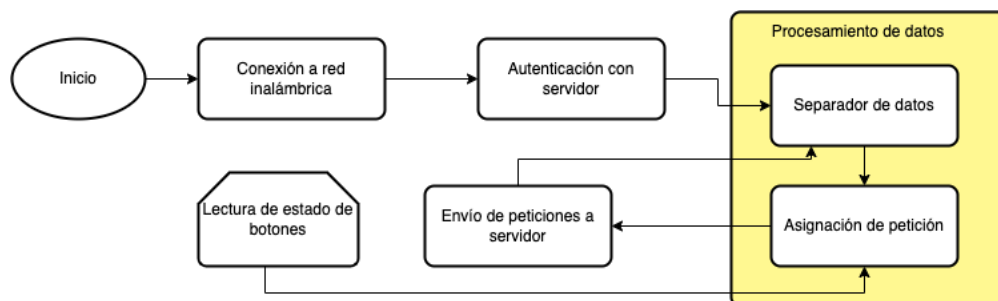
- El nodo al inicializarse y tras haber logrado conectarse a la red inalámbrica, solicitará autenticarse al servidor, el cual generará un código único (token) para entregarle al nodo y que este tenga el identificativo de que mesa es exactamente. Tras recibir los datos de respuesta, el nodo realiza un proceso de separación de datos.

- Al separar los datos, toma el código enviado por el servidor y lo utiliza para agregarlo a la cabecera de la trama en cada comunicación con el servidor para saber el nodo y la mesa correspondiente que está enviando peticiones al servidor.
- En el primer caso, cuando el cliente solicita un código de verificación para confirmar su pedido, el nodo envía una petición al servidor para que le entregue un código generado por el sistema y así confirmar la orden. Este código tendrá un tiempo de vida de un minuto para poder ingreso y enviar el código recibido por el nodo.
- En caso de que el cliente necesite asistencia en la mesa, el nodo tiene un botón para poder enviar una petición al servidor y que este muestre en la pantalla de administración que la mesa X requiere asistencia. Una vez finalizada la atención, el personal del establecimiento volverá a presionar el botón para notificar al servidor que el requerimiento ha sido cumplido.
- Cuando el pedido ya se haya entregado en su totalidad, el mesero deberá realizar la confirmación de entrega utilizando otro botón del nodo el cual adjuntará la petición del cliente junto con el código de verificación para saber que pedido es el que se ha entregado y de esta forma notificará al servidor que el pedido está completo para que este pase en la pantalla de administración a la sección de pedidos atendidos.

El diagrama de bloques expresa de manera resumida los procesos que se van a realizar para obtener un resultado. El diagrama de flujo es importante para conocer el orden de las acciones de los dispositivos para su funcionamiento.

A continuación se presenta la figura con el diagrama de bloques del funcionamiento del nodo cliente.

Figura 13
Diagrama de bloques de nodo



El nodo cliente a implementar en este proyecto iniciará con la importación de las librerías necesarias para trabajar y comunicarse con el servidor. Para ello utilizará la librería HTTP la cual importará funciones que permitan actuar acorde al protocolo de transferencia de hipertexto (HTTP). También utilizan un conjunto de librerías para poder habilitar el protocolo I2C y establecer la comunicación con la pantalla del nodo. Luego se importa la librería Wi-Fi para definir el modo de trabajo del microcontrolador como cliente y poder asociarlo a la red Wi-Fi correspondiente. Finalmente se incluye la librería JSON (JavaScript Object Notation) la cual permitirá tener una extracción de cualidades de objetos JSON y poder utilizarlas en el código de programación del nodo cliente. Luego se inician las variables que se van a utilizar de manera global en el programa y dentro de estas variables se tiene:

- Credenciales para poder conectarse a la red Wi-Fi (SSID y contraseña)
- Variables para almacenamiento de la información recibida por parte del servidor
- Variables auxiliares para restringir la ejecución de ciertas funciones
- Declaración del número de pin que se va a utilizar para los botones y comunicación serial con la pantalla.

- Direcciones URL para comunicación que el nodo cliente con el servidor según la operación a ejecutar.

Para finalizar el primer bloque, se inician las comunicaciones con los dispositivos conectados, declarar el modo de trabajo de los pines seleccionados.

En el segundo bloque se inicia el proceso de asociación del nodo con la red inalámbrica. Si esta conexión es exitosa procederá con el resto de las funciones para mostrar el logotipo del establecimiento gastronómico, de lo contrario seguirá intentando la conexión y no avanzará.

El tercer bloque utiliza la dirección IP del servidor para establecer un enlace y solicitarle mediante una petición que le otorgue un código de autenticación. El servidor recibirá la petición en una estructura indicada y enviará su respuesta.

Dentro del bloque de procesamiento de datos se tiene dos bloques:

- Separador de datos: Se encarga de recibir las respuestas de peticiones enviadas el servidor para ver si la información es importante para el nodo o descartarla. Este procesamiento depende del valor de la variable auxiliar dentro del sistema del nodo.
- Asignación de petición: Según el valor asignado a la variable auxiliar del sistema, se realiza una asignación de URL para enviar peticiones al servidor posteriormente.

El valor de la variable auxiliar depende de la lectura de los botones del nodo.

El quinto bloque corresponde al de envío de peticiones, este bloque se encarga de utilizar los datos obtenidos en el proceso de autenticación para poder agregar a la cabecera de la petición a enviar al servidor y así pasar el filtro de autenticación y a su vez dejarle saber al servidor con que nodo y la mesa que está enviando la petición.

Una vez iniciado el sistema, se inicia con la petición del nodo para autenticarse al sistema. Si llegase a tener una respuesta el nodo por parte del servidor continua con el siguiente paso para

poder procesar los datos recibidos (función “sepkey”) y separar el código de autenticación que le permitirá al nodo interactuar con el servidor. Si no tiene una respuesta, el nodo arrojará un error por no poder establecer una conexión con el servidor.

Luego de la separación de los datos, la cual se realizó almacenando la información en una variable temporal, se extrae el código entregado por el servidor y se lo almacena en una variable adicional para poder pasar a la siguiente etapa.

En la siguiente parte, el microcontrolador toma el código extraído y lo agrega dentro de la cabecera de peticiones que se van a enviar al servidor y para ello se realiza una prueba por primera vez con el valor del auxiliar en 0 y se solicita un código de verificación que será de prueba para demostrar que el servidor si logró autenticarse. Para esto continúa con el proceso de primera vez y envía la petición al servidor. Al recibir respuesta, almacena la información en otra variable para luego poder separar los datos y poder separar el código obtenido. Al ser primera vez, el sistema del nodo pasará a mostrar el logotipo de la empresa de forma directa y no el código obtenido en la respuesta del servidor. Después de esto el nodo estará iniciado y autenticado esperando acciones en el loop para poder interactuar con el servidor.

Una vez encendido el dispositivo completamente, este estará censando los pines de los botones y en caso de que un botón sea presionado ejecutará una acción:

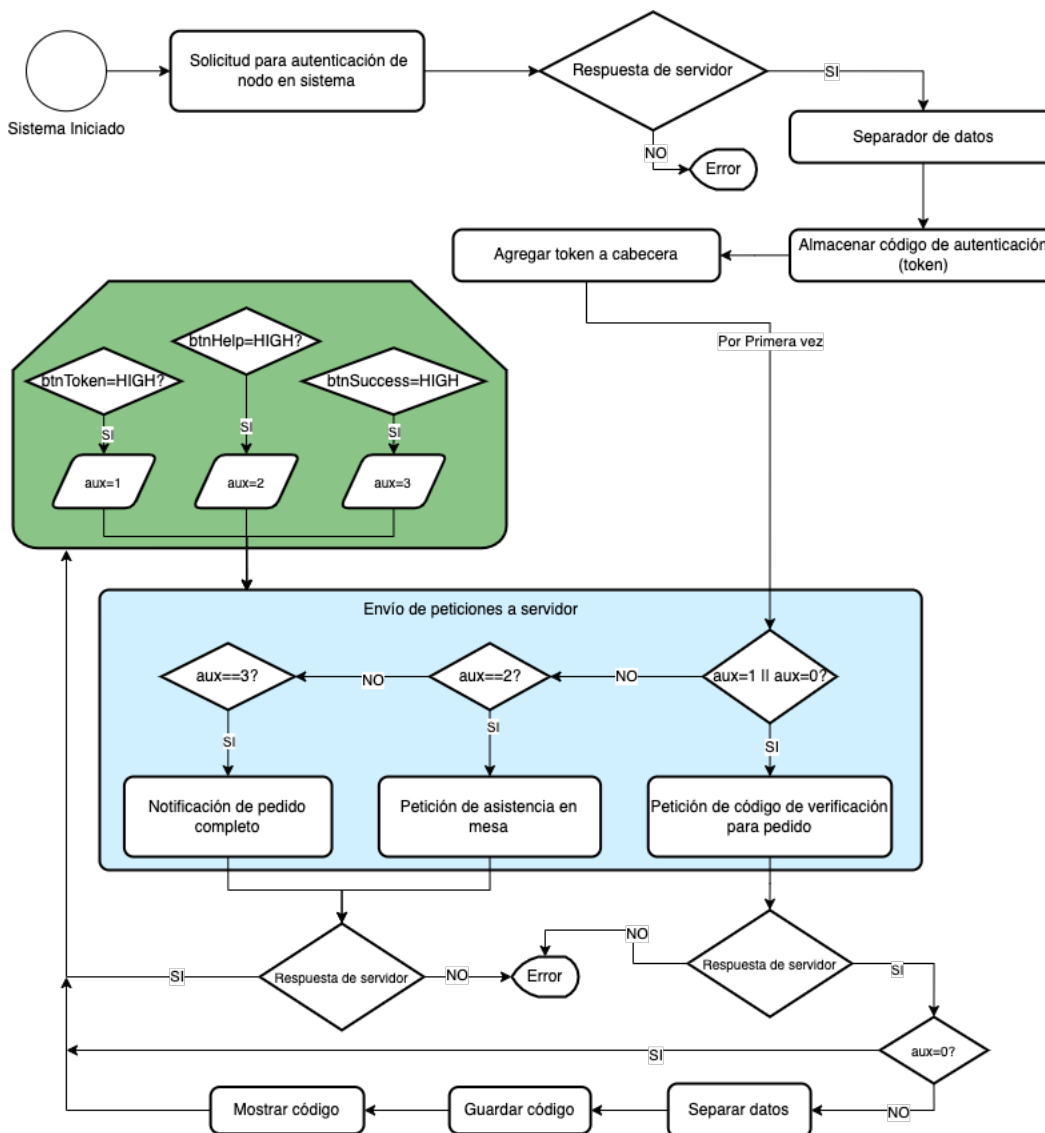
- Botón de token: Al presionar este botón (btn) se cambia el valor del auxiliar a 1 y se llama nuevamente al bloque de peticiones para el servidor y envía una solicitud hacia este para obtener un código de verificación adicional, pero el envío de la solicitud hacia el servidor deberá incluir el código de autenticación dentro de la cabecera de la solicitud HTTP para poder pasar el filtro de seguridad. La respuesta obtenida en este mensaje se almacenará en la variable “info2” y pasará a otra

función llamada “sepcode” en donde se realiza la separación de datos para obtener el nuevo token de verificación. Este código cumplirá con la función de permitir una verificación del pedido enviado por parte del cliente hacia el servidor. Si la información existe se almacena en la variable “token2” y se mostrará en la pantalla por un minuto y luego se desaparecerá para volver a mostrar el logotipo del restaurante.

- Botón de asistencia: Definido como btnHelp, si este es presionado se asigna el valor de 2 al auxiliar y se envía nuevamente una petición de requerimiento de asistencia en la mesa hacia la pantalla del administrador utilizando el enlace respectivo. A su vez utiliza el método POST para poder cambiar el estado de asistencia en la mesa. Si se vuelve a presionar el botón, el requerimiento será tomado como atendido.
- Botón de entrega: Definido como btnSuccess, si este es presionado se asigna el valor de 3 al auxiliar y se envía la notificación de pedido entregado para poder registrar la orden entregada hacia el cliente en la pantalla de administración.

Si ninguno de los botones es presionado, se mantiene en estado de lectura de los pines el nodo hasta que alguna acción cambie. La figura a continuación contiene el flujograma del nodo cliente junto con los pasos mencionados en esta sección.

Figura 14
Diagrama de flujo de nodo para peticiones



3.8.1.2. Diagrama de Conexión y Esquemático del Nodo Cliente

- **Diagrama de conexión**

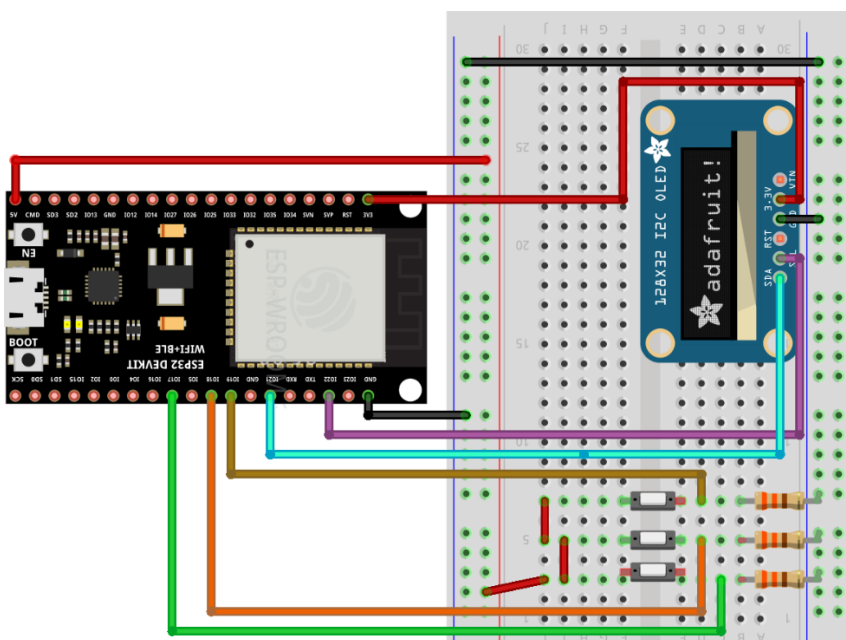
En cuanto a las conexiones de los nodos cliente, se encuentran distribuidas de la siguiente manera:

- Pin 17: lectura de botón para generación de código de verificación de pedido.
- Pin 18: lectura de botón para envío de solicitud de asistencia en mesa.

- Pin19: lectura de botón para envío de confirmación de entrega de pedido a clientes.
- Pin 21: pin SDA de comunicación con pantalla
- Pin22: pin SCL de comunicación con pantalla
- 3.3Vout: alimentador de energía para pantalla

En la figura a continuación la descripción gráfica de cómo se encuentra conectado el circuito de cada nodo.

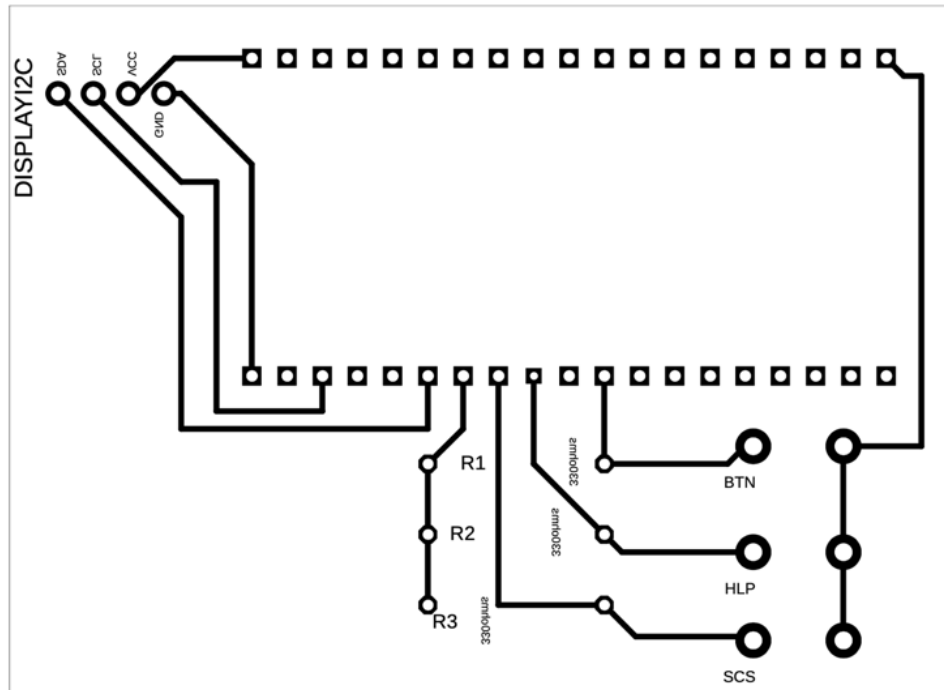
Figura 15
Diagrama de conexión de circuito para nodo cliente



- Diagrama PCB

La figura a continuación muestra el diagrama de la placa para poder proceder con el diseño e implementación de los nodos clientes en cada mesa.

Figura 16
Diagrama PCB para nodo cliente



- Fabricación de placa

El proceso de fabricación de placa para el diseño del prototipo de los nodos cumplió los pasos explicados a continuación:

- Realizar una impresión laser en papel bond, couche de 125 o papel termotransferible del diagrama PCB. Esta imagen debe estar en escala real y en modo espejo para poder realizar el traspaso hacia la placa
- Lo siguiente es colocar la impresión en la placa de cobre de tal forma que tenga el tamaño de la imagen.
- Utilizando la plancha de ropa se procede a calentar sobre el papel y la placa de cobre con movimientos verticales u horizontales para permitir que la tinta laser abandone el papel y se pegue en la placa de cobre. Este proceso se recomienda realizarlo por 10 minutos mínimo.

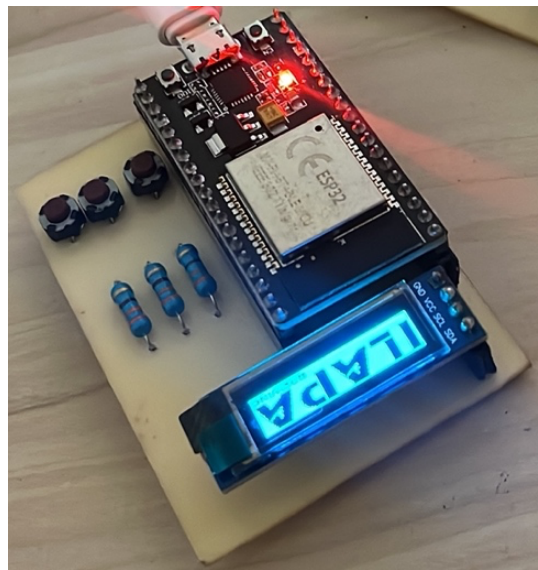
- Una vez realizado el planchado, se procede a confirmar que el papel no se desprende de la placa de cobre, caso contrario se debe repetir el paso anterior hasta confirmar que la tinta se haya quedado en su mayoría en el papel.
- Con un poco de agua se procede a humedecer el papel para que este se debilite y poder removerlo dejando la tinta impregnada en la placa de cobre.
- Se confirma que las pistas dibujadas estén correctas, si alguna de estas le falta un poco de imagen, con un poco de corrector de bolígrafo se puede completar la pista de contacto para evitar que el ácido destruya el camino del circuito.
- Luego se procede a colocar la placa con un poco de ácido para placas PCB y en un recipiente plástico se coloca una cantidad moderada de agua hirviendo para acelerar el proceso de quemado del cobre que no se encuentra cubierto por la tinta para así generar los caminos del circuito. Este proceso puede variar entre 10 a 30 minutos. Se recomienda una observación constante de este proceso para evitar daños en las pistas de cobre.
- Una vez que se compruebe que el cobre sin cubrir ha sido eliminado, se debe enjuagar con abundante agua. Un truco para saber si el cobre sin cubrir se ha eliminado es colocar una lámpara debajo de la placa, si la luz logra atravesar sin problema, la placa se quemó con éxito. Otra opción es comprobar que las pistas tengan contacto solo entre ellas con el multímetro. Si este proceso tiene contacto con el resto de la placa quiere decir que falta un poco de tiempo sumergir la placa en ácido.

- Una vez finalizada la placa se procede con la perforación de los puntos para contacto de los elementos electrónicos.
- Para finalizar, se procede a la colocación y soldadura de los dispositivos a la placa para probarla.

Una vez finalizada la placa se debe probar y si existe algún error se debe verificar que las conexiones sean iguales a los del prototipo y confirmar el contacto entre los puntos de inicio y fin del diseño de la placa.

A continuación se muestra una figura de la placa diseñada para este proyecto finalizada.

Figura 17
Placa de proyecto terminada y funcionando



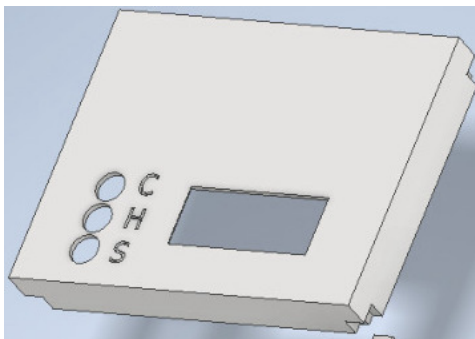
- Diseño de caja 3D para prototipo

Para este proceso se ha utilizado la aplicación Inventor de la empresa Autodesk la cual permite realizar diseños en tres dimensiones para exportarlos e imprimirlos en una impresora 3D.

La figura a continuación muestra el diseño de la parte superior o tapa del nodo cliente. Esta muestra 3 letras las cuales son para: Código (C), solicitar ayuda (H) y notificar entrega de pedido completa (S). Junto a estas letras se encuentra un agujero para poder colocar la pantalla del nodo

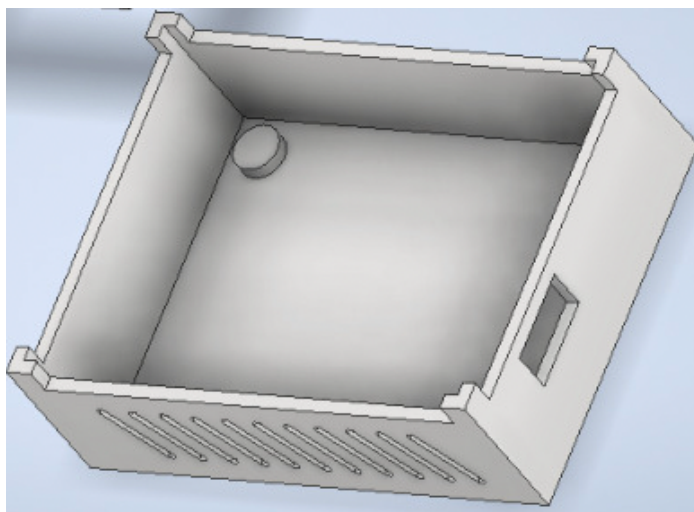
y en las esquinas se observa unas cavidades las cuales cumplirán con la función de recibir a otras pronunciaciones y generar el sello del nodo.

Figura 18
Tapa del nodo cliente en 3D



A continuación se muestra el diseño de la parte inferior en donde se colocará la placa fabricada previamente. En esta figura se observa en el interior una protuberancia la cual será el apoyo para la placa, de estas hay 4 en total en el interior. Al lateral se observa un orificio el cual permitirá el ingreso de un cable micro-USB para alimentar al nodo cliente. En la parte frontal se observan unos agujeros diagonales los cuales permitirán una ventilación para el dispositivo. Finalmente en la parte superior se observan otras protuberancias las cuales encajan a medida con las que se observan en la tapa del nodo (figura anterior).

Figura 19
Parte inferior del nodo cliente en 3D



Finalmente se exporta el archivo en el formato que la impresora 3D lo permita y se empieza la impresión de este. Este proceso puede tardar entre 2-4 horas por parte, dependiendo de la calidad, materiales e impresora que se utilice. La imagen a continuación tiene el dispositivo finalizado y ensamblado.

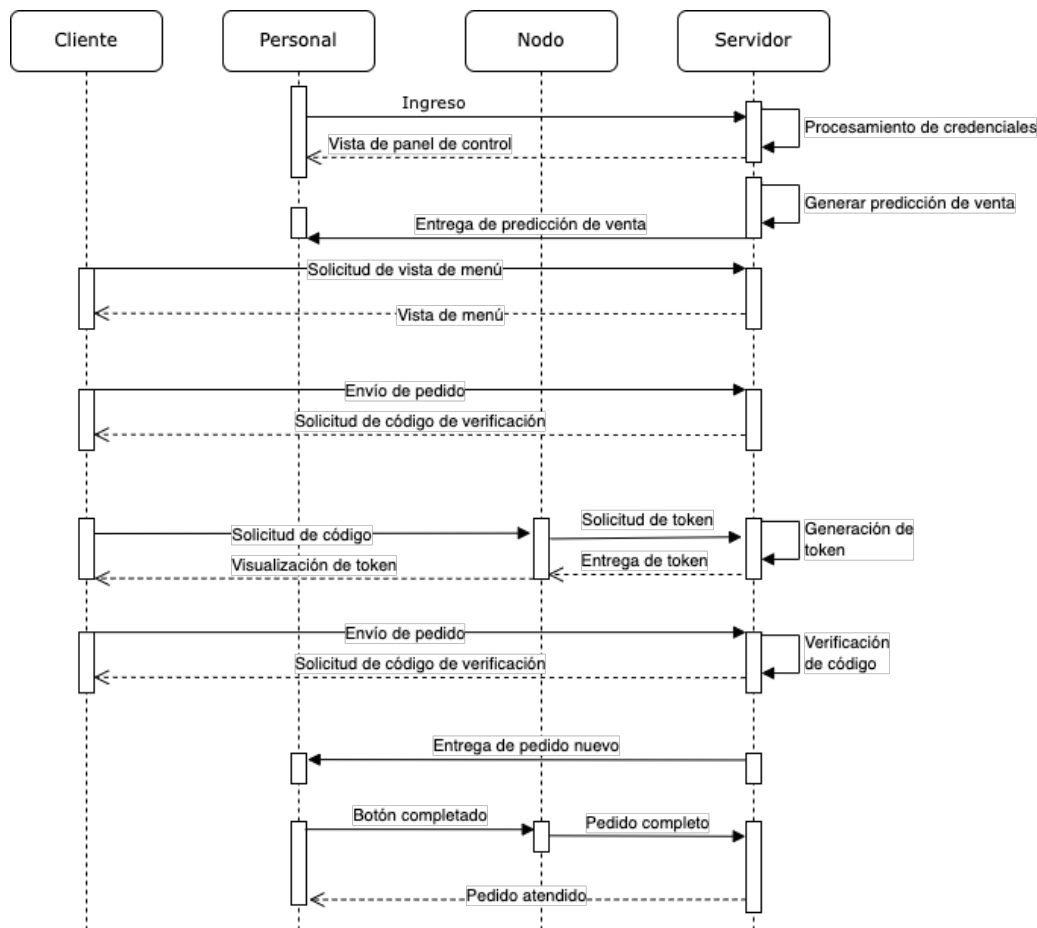
Figura 20
Nodo cliente ensamblado y conectado



3.8.2. Diseño de Software

El diagrama que se observa en la figura a continuación establece las acciones en orden desde el arranque del sistema y el usuario de administración ingresa a su panel respectivo. Cada acción viene a ser una prueba a comprobar en la sección de resultados con las pruebas de funcionamiento del sistema.

Figura 21
Diagrama secuencial del sistema



- El primer escenario es el ingreso del usuario de administración hacia el sistema en donde se solicitarán las credenciales para poder acceder, si estas son correctas el sistema redireccionará al usuario hacia el panel de administración.
- Una vez que el usuario administrador haya ingresado, el servidor realizará una tarea interna de generar la predicción de venta del sistema que servirá de apoyo para los usuarios administradores del establecimiento para tener las cantidades de los productos disponibles para los clientes.
- Cuando un usuario cliente se conecta al servidor, este realiza la petición de ingreso hacia la pantalla de bienvenida. El servidor recibe esta petición y devuelve como

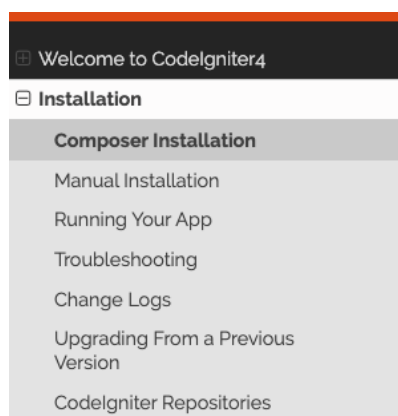
respuesta la pantalla de productos para agregar al carrito y poder realizar el pedido de manera automática.

- Una vez que se haya completado la selección de los ítems a ordenar junto con los datos de facturación, el cliente procede a confirmar el pedido y el servidor responderá con una solicitud de un código de verificación el cual debe ser solicitado a través del nodo cliente ubicado en las mesas.
- Para obtener el código de confirmación, el cliente debe seguir las instrucciones mostradas en pantalla y presionar un botón en el dispositivo, el nodo enviará una solicitud al servidor para que genere un código aleatorio de 6 dígitos y este será entregado al nodo, este código será visualizado por el cliente a través de la pantalla del nodo y así podrá ingresar el código en la interfaz de usuario para finalizar el pedido.
- Después de ingresar el código en la interfaz y confirmar el envío del pedido, el servidor recibe esta petición, realiza una verificación del código, es decir, que cumpla los parámetros de tiempo y si todo es correcto, notifica mediante la interfaz al cliente que el pedido ha sido aceptado.
- El servidor enviará una notificación a la interfaz de administración para notificar que se encuentra un nuevo pedido en cola.
- Una vez que se haya entregado el pedido al cliente, el mesero confirmará la entrega presionando otro botón en el nodo para notificarle al servidor que el pedido se encuentra entregado para finalizar la acción de ese pedido.

Para el desarrollo del servidor, como se ha mencionado antes, se procedió a utilizar el framework Codeigniter versión 4.0. para ello se debe ingresar en la página de esta aplicación:

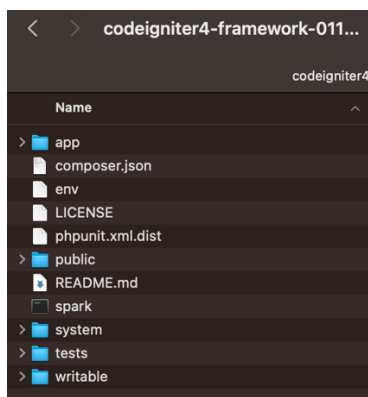
codeigniter.com y en la pestaña de instalación se debe seguir los pasos de la instalación escogida por preferencia. Se recomienda utilizar el método “composer” dado que instalará una herramienta que a futuro permitirá instalar paquetes adicionales compatibles con el framework.

Figura 22
Opciones de instalación de Codeigniter 4



Una vez realizado el proceso de instalación se obtendrá como resultado una carpeta con todos los archivos correspondientes para que el sistema funcione.

Figura 23
Carpeta de framework obtenida tras instalación



Dentro de esta carpeta general se tienen las siguientes subcarpetas:

- App: contiene todos los archivos y configuraciones del sistema con las configuraciones ingresadas por desarrollador para la ejecución de la aplicación. El redireccionamiento

interno entre archivos lo realiza el framework con las configuraciones que provea el desarrollador junto con las establecidas en la carpeta “system”.

- Public: esta carpeta se convierte en la raíz del proyecto, es decir, todos los archivos html o fuentes externas deben ser ingresados en esta carpeta.
- System: tiene todos los archivos principales de configuración del framework, es la base del sistema.
- Tests: contiene todos los archivos que comprueban si el sistema funciona de forma correcta o si existe algún error.
- Writable: contiene información para desarrollo como los logs y así observar los errores que vayan sucediendo a lo largo del desarrollo de la aplicación. (Codeigniter, 2023)

Para la programación del servidor se requiere conocimiento del lenguaje de programación PHP y en específico de la versión 7.4 en este caso. Este lenguaje de programación significa Hipertext Preprocessor y tiene la característica de ser utilizado para desarrollo de aplicaciones web dado que tiene una fácil complementación con HTML. Además, PHP trabaja del lado del servidor, generando datos que pueden ser enviados al lado del cliente de forma más compacta gracias a los códigos predefinidos que ofrecen el envío de información de manera más sencilla. (PHP Group, 2022)

Después de una breve introducción, a continuación se detallan las configuraciones básicas para el desarrollo de este proyecto. Como primer punto, dentro de la carpeta /app/config, existe el archivo app.php, este contiene una línea de comando para establecer la dirección base para el proyecto en donde se procede a colocar la dirección que se le ha asignado al servidor para su funcionamiento como se observa en el recuadro rosado de la siguiente figura.

Figura 24
Asignación de URL base para proyecto

```

23  * environments.
24  *
25  * @var string
26  */
27  public $baseUrl = 'http://192.168.1.123/';
28
29  /**
30  *

```

Otro archivo importante a configurar es el de la base de datos, el cual se encuentra en la misma carpeta de /app/config y es llamado database.php. Dentro de este se colocan las configuraciones respectivas para el acceso a la base de datos creada.

Figura 25
Configuración de archivo de base de datos

```

27  /**
28  * The default database connection.
29  *
30  * @var array
31  */
32  public $default = [
33      'dsn' => '',
34      'hostname' => 'localhost',
35      'username' => 'root',
36      'password' => 'Paz2259+',
37      'database' => 'restaurante',
38      'dbDriver' => 'MySQLi',
39      'dbPrefix' => '',
40      'pconnect' => false,
41      'dbDebug' => (ENVIRONMENT !== 'production'),
42      'charset' => 'utf8',
43      'dbCollat' => 'utf8_general_ci',
44      'swapPre' => '',
45      'encrypt' => false,
46      'compress' => false,
47      'strictOn' => false,
48      'failover' => [],
49      'port' => 3306,
50  ];

```

Finalmente, en esta misma carpeta se encuentran las rutas. Estas serán las guías de direccionamiento de las peticiones que ingresen al servidor, por ejemplo: ingresa la ruta 192.168.1.123/ en el navegador, el servidor recibe la petición y el sistema lo guía hacia el controlador Home junto con el método index dentro de este controlador. Este ejemplo se encuentra señalado en la figura siguiente en el recuadro azul.

El resto de las rutas hacen referencia a los métodos a utilizar para el procesamiento de los datos entre el servidor y las interfaces de usuario de cliente y de administración. La estructura que maneja la ruta: ipServidor/controlador/método:

- La dirección IP del servidor también puede ser reemplazada por la dirección URL en caso de haber adquirido el dominio.
- El controlador cumple con la función de procesar las peticiones HTTP enviadas al servidor, en otras palabras, es un traductor de operaciones que tiene diferentes funciones.
- El método es la función que se va a ejecutar dentro del controlador y dentro de cada función están las acciones que se van a realizar dentro del servidor como realizar consultas, ingresar datos nuevos a la base de datos, entre otras.

Figura 26
Rutas para sistema

```

$routees->get('/', 'Home::index');

$routees->cli('/websocket/start', 'Websocket::start');
$routees->cli('/websocket/send', 'Websocket::send');
$routees->get('/websocket/user/(:segment)', 'Websocket::user/$1');

$routees->get('/productos', 'Home::productos');

$routees->get('/admin-login', 'Admin::login');
$routees->post('/admin-login', 'Admin::login');
$routees->get('/logout', 'Admin::logout');

$routees->get('/admin/productos', 'Admin::productos');
$routees->get('/admin/pedidos', 'Admin::pedidos');
$routees->get('/admin/procesar/(:num)', 'Admin::procesar/$1');
$routees->get('/admin/despachar/(:num)', 'Admin::despachar/$1');
$routees->get('/admin/cancelar/(:num)', 'Admin::cancelar/$1');
$routees->get('/admin/resumen', 'Admin::resumen');
// $routees->resource('login');
$routees->get('/login/(:any)/(:any)', 'Login::index/$1/$2');
$routees->get('/mesa/alerta', 'Mesa::alerta');
$routees->get('/mesa/atendido', 'Mesa::atendido');
$routees->get('/mesa/recibir/(:num)', 'Mesa::recibir/$1');
$routees->get('/mesa/list', 'Mesa::list');

```

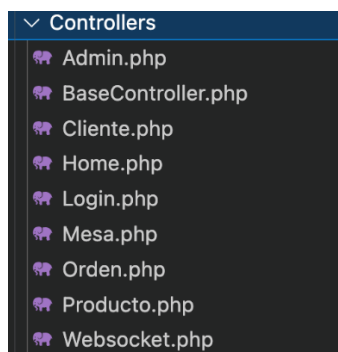
Seguido se explicará cada archivo del sistema importante solo la información relevante. El código completo se encontrará en la sección de anexos en el enlace del repositorio.

3.8.2.1. Controladores

Como se mencionó anteriormente, los controladores son los encargados de procesar las peticiones que llegan al servidor para establecer las acciones a tomar según la petición y la ruta seleccionada.

Para el desarrollo de este proyecto se crearon 8 controladores, pero en la imagen a continuación se observan 9 en total. Esto sucede porque cada uno de los archivos creados necesitan establecer una extensión del archivo BaseController para saber cuál va a ser su función en el servidor. El listado de los controladores se aprecia en la figura a continuación.

Figura 27
Listado de controladores del proyecto



El controlador “Admin” es el encargado de todas las funciones que se relacionen con la sección de administración y una de las funciones es la que se observa en la figura a continuación para poder establecer la verificación de acceso de los usuarios al panel de administración donde adquiere el post del usuario, compara si los dos valores son diferentes de nulo, realiza una consulta en la base de datos para obtener los datos del usuario ingresado si es que este existe, compara estos datos los cuales son cifrados para mayor seguridad y si son correctos se establece un inicio de sesión para permitir al usuario mantenerse dentro del perfil de administrador.

Figura 28
Controlador Admin: Función de ingreso

```

app > Controllers > Admin.php
12
13 class Admin extends BaseController
14 {
15     public function login(){
16         $model = new UsuarioModel();
17         if($this->request->getMethod()=='post'){
18             $username = $this->request->getPost('username');
19             $password = $this->request->getPost('password');
20
21             $mensajes = [];
22             if($username == null && $password == null){
23                 $mensajes[] = 'El usuario y contraseña son requeridos';
24                 return $this->twig->render('usuarios/login.html.twig', ['mensajes'=>$mensajes]);
25             }
26             $usuario = $model->where('username', $username)->first();
27             $mensajes[] = 'Usuario o contraseña incorrectos';
28             if(!$usuario){
29                 return $this->twig->render('usuarios/login.html.twig', ['mensajes'=>$mensajes]);
30             }
31
32             $hash = $usuario->password;
33
34             if(!password_verify((string)$password, $hash)){
35                 return $this->twig->render('usuarios/login.html.twig', ['mensajes'=>$mensajes]);
36             }
37
38             $session = session();
39             $session->set(['user'=>$username]);
40             return redirect('Admin:productos');
41         }
42         return $this->twig->render('usuarios/login.html.twig', ['mensajes'=>[]]);
43     }

```

Otras funciones que se observarán dentro de este controlador son: consultar productos, consultar pedidos con sus estados, cambiar el estado de un pedido y otra función muy relevante es la responsable de ejecutar el código de aprendizaje de máquina para generar las predicciones de venta.

Esta función se observa en la figura siguiente donde se toma como referencia la fecha del sistema para poder obtener como dato el día que se va a predecir basado en las ventas del día anterior. A continuación, el sistema realiza una consulta a diferentes tablas como la de “orden_item” y la de productos para obtener información sobre los productos, luego compara si el día seleccionado es feriado, luego se establece el rango del feriado con la información ingresada en la base de datos del sistema. Finalmente realiza la consulta sobre los valores salientes de cada plato del establecimiento, para este caso se han utilizado 20 platos de ejemplo para entrenar y para el desarrollo del sistema.

Figura 29
Controlador Admin

```

Controllers > Admin.php

public function resumem(){
    $session = session();
    $user = $session->get('user');
    if(true){
        $currentDate = new DateTime();
        $tempDate = new DateTime();
        $yesterdayTime = $tempDate->sub(new DateInterval('P1D'));
        $strCurrent = $currentDate->format('Y-m-d');
        $strYesterday = $yesterdayTime->format('Y-m-d');
        $sql = "SELECT d.cantidad, p.*, SUM(d.cantidad) as total
                FROM orden o
                INNER JOIN orden_item d ON o.id = d.id_orden
                INNER JOIN producto p ON d.id_producto = p.id AND p.codigo = :codigo:
                WHERE fecha >= '$strYesterday' AND fecha < '$strCurrent'
                GROUP BY p.id";

        $sqlFeriadoHoy = "SELECT * FROM `feriado` WHERE CURDATE() between inicio and fin";
        $sqlFeriadoAyer = "SELECT DATE_SUB(CURDATE(), INTERVAL 1 DAY) AS ayer, f.* FROM `feriado` f HAVING ayer between inicio and fin";

        $current_day = date("w", strtotime($currentDate->format('l')));
        $current_day = $current_day == 0 ? 7 : $current_day;
        $yesterday_day = $current_day - 1;
        $yesterday_day = $yesterday_day == 0 ? 7 : $yesterday_day;

        $db = \Config\Database::connect();
        $entradas = [];
        $shoyEsFeriado = 0;

        $feriadoHoy = $db->query($sqlFeriadoHoy)->getResult();
        if($feriadoHoy) $shoyEsFeriado = 1;

        $entradas[0] = $shoyEsFeriado;
        $entradas[1] = (int)$current_day;

        for($i = 0; $i < 20 ; $i++){
            $codigo = $i+1;
            $resumen = $db->query($sql, ['codigo'=>$codigo])->getResult();
            $entradas[2 + $i] = 0;
            if($resumen)$entradas[2 + $i] = (int)$resumen[0]->total;
        }

        $feriadoAyer = $db->query($sqlFeriadoAyer)->getResult();
        $ayerFueFeriado = 0;
        if($feriadoAyer) $ayerFueFeriado = 1;
        $entradas[22] = $ayerFueFeriado;
        $entradas[23] = $yesterday_day;
        return json_encode($entradas);
    }
}

```

El siguiente controlador es el de cliente. Este tiene dos funciones las cuales se encargan del registro y búsqueda de información en la base de datos para clientes existentes.

Figura 30
Controlador Clientes

```

app > Controllers > Cliente.php

1  <?php
2
3  namespace App\Controllers;
4
5  use App\Models\ClienteModel;
6  use CodeIgniter\RESTful\ResourceController;
7  use CodeIgniter\API\ResponseTrait;
8
9  class Cliente extends ResourceController
10 {
11     use ResponseTrait;
12     public function buscar()
13     {
14         $json = $this->request->getJSON();
15         $data=$json->ci;
16         $model = new ClienteModel();
17         $cliente=$model->where('ci',$data)->first();
18         return $this->respond($cliente);
19     }
20 }
21
22     public function guardar()
23     {
24         $response['success'] = false;
25         $model = new ClienteModel();
26         $data = $this->request->getJSON(true);
27         $cliente = new \App\Entities\Cliente();
28         $cliente->fill($data);
29         $model->save($cliente);
30         $response['success'] = true;
31         $response['cliente']=$data;
32         return $this->respond($response);

```

El controlador en la figura a continuación es el de Home, este tiene las funciones de llamar a la pantalla de bienvenida para los clientes y la otra función se encarga de realizar la consulta en la base de datos sobre los productos disponibles para mostrar en la pantalla de clientes y en la sección de administración de ítems.

Figura 31
Controlador Home

```
Controllers > Home.php
<?php

namespace App\Controllers;

use App\Models\ProductoModel;

class Home extends BaseController
{
    public function index()
    {
        $model = new ProductoModel();
        $productos = $model->findAll();

        echo view('index.html');
    }

    public function productos()
    {
        $model = new ProductoModel();
        $entradas = $model->where('tipo', 'ent')->findAll();
        $fuertes = $model->where('tipo', 'fr')->findAll();
        $bebidas = $model->where('tipo', 'dr')->findAll();
        $licores = $model->where('tipo', 'lc')->orWhere('tipo', 'ck')->findAll();
        $promociones = $model->where('tipo', 'pr')->findAll();
        return $this->twig->render('productos/show.html',
        [
            'entradas' => $entradas,
            'fuertes' => $fuertes,
            'bebidas' => $bebidas,
            'licores' => $licores,
            'promociones' => $promociones
        ]);
    }
}
```

El controlador siguiente es el de “Login”, este es el encargado de controlar el registro de los nodos de las mesas para comparar su información. Si todos los datos recibidos por el servidor son correctos, este controlador devolverá un token el cual será utilizado para filtrar la información recibida por los nodos para saber que nodo es el que envía las peticiones.

Figura 32
Controlador Login

```
Controllers > Login.php

public function index($codigo = '', $secreto = '')
{
    try {
        helper('encryption');
        $mesaModel = new MesaModel();

        $mesa = $mesaModel->where('codigo', $codigo)->first();

        if(is_null($mesa)) {
            return $this->respond(['error' => 'Datos de la mesa incorrectos.'], 401);
        }

        $hash = $mesa->secreto;
        $pwd_verify = password_verify($secreto, $hash);

        if(!$pwd_verify) {
            return $this->respond(['error' => 'Datos de la mesa incorrectos.'], 401);
        }

        $key = getSecret();
        $iat = time();
        $exp = $iat + 2592000;

        $payload = array(
            "iss" => "RESTAURANTE",
            "aud" => "RESTAURANTE",
            "sub" => "Authentication",
            "iat" => $iat,
            "exp" => $exp,
            "mesa_id" => $mesa->id,
        );

        $token = JWT::encode($payload, $key, 'HS256');
```

El controlador “Mesa” es el encargado de procesar las peticiones realizadas por los nodos, por ejemplo, una de las funciones de este controlador es el de generar el token o código de verificación para validar el pedido del cliente y así guardar el pedido en el sistema. Otras funciones son recibir las peticiones de ayuda y de entrega de pedido.

Figura 33
Controlador Mesa

```
> Controllers > Mesa.php

public function index()
{
    helper('jwt');
    $model = new TokenModel();
    $userData = getUserFromRequest($this->request);
    $token = new Token();
    $token->token = rand(100000, 999999);
    $token->inicio = time();
    $token->fin = $token->inicio + 180;
    $token->id_mesa = $userData->mesa_id;
    $model->save($token);
    return $this->respond($token);
}

public function alerta()
{
    helper('jwt');
    helper('websocket');
    $model = new MesaModel();
    $userData = getUserFromRequest($this->request);

    $id = $userData->mesa_id;
    $mesa = $model->find($id);

    $mesa->alerta = $mesa->alerta ? "0":"1";
    $model->update($id, $mesa);
    $msg = [];
    $msg['mesa_id'] = $id;
    $msg['alerta'] = $mesa->alerta;
    $jsonMsg = json_encode($msg);
    send($jsonMsg);
}
```

El siguiente controlador es el de “Orden”, este cumple la función principal de procesar las peticiones para poder realizar un pedido. Compara las variables ingresadas para verificar que el usuario haya ingresado ítems dentro del carrito de compras, confirma el código de verificación para enlazarlo al pedido y así saber que mesa realizó el pedido y finalmente envía el pedido hacia el servidor para que este aparezca en la pantalla principal del usuario de administración.

Figura 34
Controlador Orden

```

Controllers > Orden.php

public function create()
{
    $response['success'] = false;

    $data = $this->request->getJSON(true);

    if(!isset($data['productos']))
        return $this->respond(['error'=>'La variable productos es requerida.'], 400);
    if(count($data['productos']) == 0)
        return $this->respond(['error'=>'Sin productos.'], 400);
    if(!isset($data['token']))
        return $this->respond(['error'=>'Token no valido.'], 400);

    $tokenData = $data['token'];
    $tokenModel = new TokenModel();
    $token = $tokenModel->where('token', $tokenData)->orderBy('inicio', 'DESC')->first();

    if(!$token)
        return $this->respond(['error'=>'Token no existe.'], 400);

    $currentTime = time();
    $fin = $token->fin;
    if($currentTime>$fin)
        return $this->respond(['error'=>'Token caducado.'], 400);

    $model = new OrdenModel();
    $orden = new \App\Entities\Orden();
    $orden->id_token = $token->id;
    $orden->fecha = (new DateTime())->format('Y-m-d H:i:s');
    $orden->observaciones = isset($data['observaciones']) ? $data['observaciones'] : null;

    $productos = $data['productos'];

```

El encargado de todas las acciones que tienen que ver con los productos para el panel de administración, es el controlador “Producto”. Este tiene las funciones pertinentes para procesar los datos y poder crear, borrar, editar y mostrar los productos que existan en la base de datos.

Figura 35
Controlador Producto

```

Controllers > Producto.php

public function create()
{
    $response['success'] = false;
    $model = new ProductoModel();
    $data = $this->request->getJSON(true);
    $producto = new \App\Entities\Producto();
    $producto->fill($data);
    $model->save($producto);
    $response['success'] = true;
    $response['producto']=$model->find();
    return $this->respond($response);
}

public function index()
{
    $model = new ProductoModel();
    $data = $model->findAll();
    return $this->respond($data);
}

public function update($id = null)
{
    $response['success'] = false;
    $model = new ProductoModel();
    $data = $this->request->getJSON(true);
    $entity = new \App\Entities\Producto();
    $entity->fill($data);
    $model->update($id, $entity);
    $response['success'] = true;
    return $this->respond($response);
}

```

Finalmente se tiene al controlador responsable de que la comunicación en tiempo real exista, el controlador “WebSocket”. Este controlador nace de una librería que se encuentra en internet y se requiere adaptarla para que funcione con la dirección IP del servidor y se pueda establecer la función de notificaciones y entrega de pedidos en tiempo real.

Figura 36
Controlador WebSocket

```

Controllers > Websocket.php
* @author      Taki Elias <taki.elias@gmail.com>
* @license     http://opensource.org/licenses/MIT > MIT License
* @link        https://github.com/takielias
*
* CodeIgniter WebSocket library. It allows you to make powerful realtime application
*/
class WebSocket extends Controller
{
    /**
     * @var Config
     */
    protected $config;
    /**
     * @var CodeIgniterWebSocket
     */
    private $ws ;

    public function __construct()
    {
        helper('codeigniter_websocket');
        $this->config = config('CodeIgniterWebSocket');
    }

    public function start()
    {
        output('info', 'Corriendo web socket');
        $this->ws = service('CodeIgniterWebSocket');

        $this->ws->set_callback('auth', array($this, '_auth'));
        $this->ws->set_callback('event', array($this, '_event'));
        $this->ws->run();
    }
}

```

3.8.2.2. Entidades

Las entidades dentro del framework cumplen con la función de ordenar los datos en cierto formato para que puedan ser procesados por el servidor e interactuar con la base de datos. El proceso correcto para poder guardar información dentro de la base de datos es:

- Establecer la entidad según el formato de la base de datos
- Llenar los datos con el formato de la entidad
- Utilizar el modelo para ingresar los datos basados en la estructura de la entidad

La siguiente figura contiene un ejemplo de la estructura de una entidad utilizada para el desarrollo del proyecto. En esta entidad de cliente se observa los atributos y el orden de estos, dado

que la creación de este es como un arreglo de datos y no como un objeto, el orden de ingreso de los datos debe ser el mismo para que la base de datos los pueda tomar y almacenar.

Figura 37
Entidad Cliente

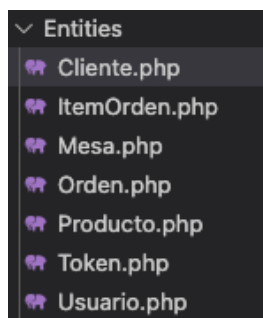
```

app > Entities > Cliente.php
1  <?php
2
3  namespace App\Entities;
4
5  use CodeIgniter\Entity\Entity;
6
7  class Cliente extends Entity
8  {
9      protected $dates = ['created_at', 'updated_at', 'deleted_at'];
10     protected $casts = [
11         'options' => 'array',
12         'options_object' => 'json',
13         'options_array' => 'json-array',
14         'precio' => 'float'
15     ];
16     protected $attributes = [
17         'id' => null,
18         'ci' => null,
19         'nombApp' => null,
20         'direccion' => null,
21         'telefono' => null,
22         'correo' => null,
23     ];

```

Las demás entidades manejan una estructura similar donde solo cambian los campos que tenga cada entidad o cada elemento del sistema. A continuación se muestra la figura con el listado de las entidades que se han creado para los elementos del sistema.

Figura 38
Lista de entidades del sistema



3.8.2.3. Modelos

Los modelos cumplen la función de llamar a la base de datos y a la fila de una tabla específica, en este caso, cada modelo está relacionado con el elemento que se desea tratar, como

se observa en la figura a continuación donde se encuentra el código del modelo de cliente, donde se toma de la tabla clientes utilizando el identificador (id) correspondiente a un código único para cada cliente y se coloca los datos a obtener según las columnas de la tabla. Este orden de datos debe ser el mismo de la entidad dado que como se observa en el recuadro azul de la figura a continuación, los datos obtenidos de la consulta serán entregados al usuario acorde a la estructura de la entidad, la cual cumple con obtener los campos de los datos en el mismo orden que la tabla.

Figura 39
Modelo Cliente

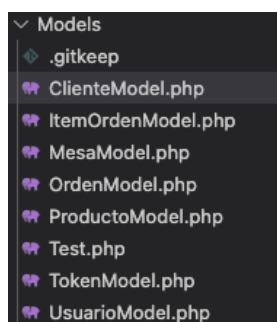
```

app > Models > ClienteModel.php
1  <?php
2  namespace App\Models;
3  use CodeIgniter\Model;
4  class ClienteModel extends Model
5
6
7      protected $table = 'clientes';
8      protected $primaryKey = 'id';
9      protected $allowedFields = [ 'ci', 'nombAp', 'direccion', 'telefono', 'correo'];
10     protected $returnType = \App\Entities\Cliente::class;
11

```

De igual forma que las entidades, por cada entidad se crea un modelo y a continuación se observa en la figura el listado de los modelos creados para los elementos del sistema.

Figura 40
Listado de modelos del sistema



3.8.2.4. Vistas

La siguiente y última carpeta importante es la de vistas. Este directorio contiene toda la codificación que permite crear la interfaz gráfica para clientes y administración. También tiene el

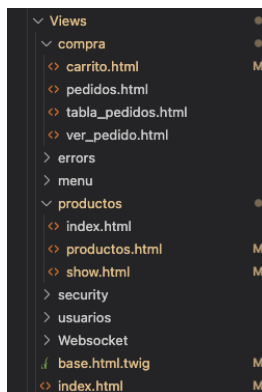
código necesario para poder tomar los datos ingresados por los usuarios para poder manipularlos y generar las interacciones correspondientes.

En este apartado se enlistarán las vistas que tiene el sistema y se detallarán las más importantes.

La figura a continuación contiene la lista de todas las vistas que se tiene en el sistema.

- Dentro de la carpeta de compra se tiene las vistas pertinentes para poder generar la tabla de pedidos en formato HTML, observar los pedidos en una pantalla y el carrito de compras del cliente.
- En la carpeta de “errors” se tiene todas las plantillas del sistema para presentar un error en caso de que no se encuentre un archivo.
- La carpeta de menú tiene los archivos fuente para poder mostrar las imágenes en la pantalla de inicio para el cliente.
- En la carpeta de productos se tiene los diseños de interfaz para mostrar los productos hacia el cliente de forma individual y en conjunto después de realizar la consulta sobre todos los dentro de la tabla de productos. cuenta con la interfaz para manipular los datos de los productos existentes o ingresar nuevos productos al sistema.
- La carpeta “usuarios” corresponde a la interfaz de ingreso para el usuario .
- El archivo “base.html.twig” es una plantilla que se incluye en diferentes pantallas para mantener un menú de opciones y poder redireccionar a los usuarios.

Figura 41
Listado de vistas del sistema

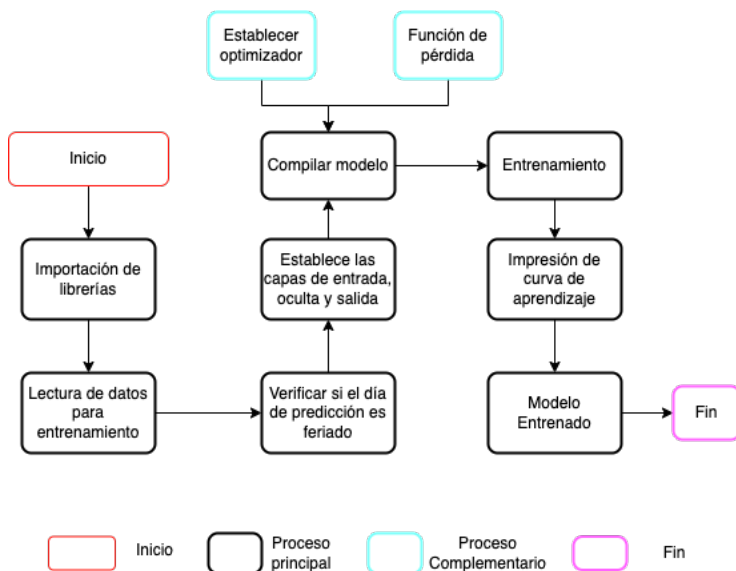


3.8.3. Machine Learning

Para poder realizar las predicciones de venta se realizó el desarrollo de una red neuronal. Una red neuronal no es más que una forma de inteligencia artificial y utiliza el aprendizaje profundo de los algoritmos de aprendizaje de máquina para poder analizar los datos de forma muy similar a como lo haría el cerebro humano y poder generar resultados o tomar decisiones basadas en la información entregada al sistema mucho más rápido que el ser humano.

La red neuronal mencionada anteriormente sigue la secuencia del diagrama de la figura a continuación para el entrenamiento del algoritmo de aprendizaje automático. En la parte inferior de la figura se tiene las características de cada recuadro y su función en el diagrama.

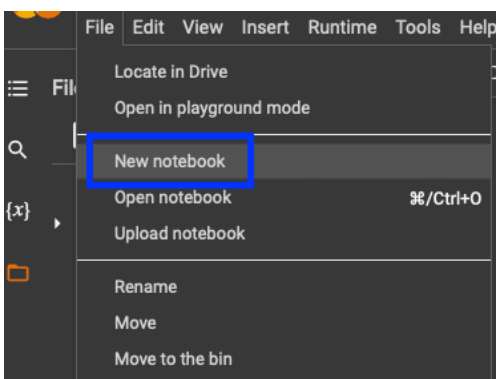
Figura 42
Diagrama de flujo para entrenamiento de algoritmo



Para realizar esta red neuronal se ha utilizado la herramienta gratuita de Google llamada Colaboratory o también conocido como “Google colab”. Esta herramienta permite ejecutar códigos de programación en lenguaje Python dentro del navegador y además cuenta con una extensa cantidad de librerías para aprendizaje de máquina, redes neuronales, entre otras.

Para iniciar se crea un nuevo cuaderno en la opción de Archivo/nuevo cuaderno como se observa en la figura siguiente.

Figura 43
Creación de nuevo cuaderno en google colab



A continuación, ya se tiene las líneas para empezar a programar. Como inicio se debe agregar las librerías que se va a utilizar junto con un alias para poder utilizarlas a futuro dentro del código y así evitar el escribir el nombre completo de la librería.

Figura 44
Importación de librerías en google colab

```
import pandas as pd
import numpy as np
import tensorflow as tf
import keras as k
```

Luego se requiere de tener la base de datos de entrenamiento, para esto se ha generado un archivo Excel con el reporte de ventas de los últimos 3 meses del establecimiento gastronómico en el que se va a realizar las pruebas en ambiente real y también se establece dentro de estos datos si existen días feriados dentro de los datos ingresados a analizar.

La imagen a continuación muestra los datos de entrada que servirán como alimentador para el sistema. La primera columna mantiene los datos del presente, es decir, si el día es feriado llevará un 1, caso contrario irá vacío o 0, el número de día (lunes=1, martes=2, etc.), las 20 siguientes columnas son los ítems y las ventas que han existido de estos ítems y finalmente tiene los datos del día anterior si fue feriado y el número de día anterior.

Figura 45
Datos de entrada para alimentación de sistema

feriado	dia	picanha_pre	burrito_pre	hamburguesa_pre	tacos_pre	matambre_pre	polloGiri_pre	mojito_pre	aguaSq_pre	pisenerpq_pre	pisenerq_pre	clubpq_pre	clubg_pre	chickenStrip_pre	bifeChorizo_pre	picadaLapa_pre	alfas12_pre	alfas6_pre	gaseosa_pre	tehelado_pre	nachos_pre	feriado_pre	dia_pre	
1	3	5	3	3	1	1	0	0	0	0	0	4	1	0	0	0	2	2	0	13	1		2	
1	4	4	0		1	0	7	1	4	1	0	2	0	0	0	0	1	1	3	2	6	2	1	3
	5	0	0		1	0	0	0	0	0	5	0	0	2	0	1	1	0	5	0	1	1	4	

Estos datos se acomodan de forma en que puedan ser procesados por la librería de pandas. Esta librería es similar a Excel pero de Python y trabaja de la mano con otra librería llamada numpy para el procesamiento de datos.

Figura 46
Lectura de archivos de datos para entrenamiento de red neuronal

```
entrada = pd.read_csv('entradas.csv', sep=';')
salida = pd.read_csv('salidas.csv', sep=';')
entrada.loc[entrada['feriado'] != 1, 'feriado'] = 0
entrada.loc[entrada['feriado_pre'] != 1, 'feriado_pre'] = 0
```

Luego se prepara el algoritmo con el número de neuronas. Este número de neuronas varía mucho según el número de datos a analizar. Para este caso se realizará la predicción de 20 ítems por lo que se establece 20 neuronas en las capas ocultas como se observa en la figura siguiente y se utiliza 2 capas ocultas para darle mayor precisión a la red neuronal y una capa de salida para finalmente entrenar el algoritmo de forma secuencial con la última línea de la figura siguiente.

Figura 47
Definición de neuronas y capas para entrenamiento de modelo

```
oculta1 = tf.keras.layers.Dense(units=20, input_shape=[24])
oculta2 = tf.keras.layers.Dense(units=20)
capaSalida = tf.keras.layers.Dense(units=20)
modelo = tf.keras.Sequential([oculta1, oculta2, capaSalida])
```

Lo siguiente es compilar el modelo y para ello se utiliza dos propiedades que corresponden al optimizador y la función de pérdida. Estas en conjunto van a programar el algoritmo para que este se encargue de aprender y no de desaprender y con el valor de 0.1 permitirá que se realice un ajuste de entrenamiento para la tasa de aprendizaje adecuado para la predicción de venta. Mientras menor sea este valor, será mejor la predicción, pero tomará más tiempo el entrenamiento del algoritmo y se requerirá más información.

Figura 48
Compilado de algoritmo

```
modelo.compile(
    optimizer=tf.keras.optimizers.Adam(0.1),
    loss='mean_squared_error'
)
```

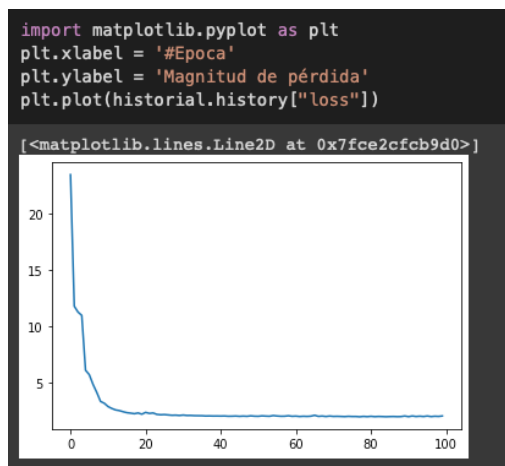
Lo siguiente es entrenar el algoritmo, pero primero se debe colocar los datos en arreglo para que Python los pueda leer y además se establece que los datos a manejar son de tipo float y no enteros para poder arrojar las predicciones.

Figura 49
Entrenamiento de modelo

```
print('Entrenando...')
npEntrada = entrada.to_numpy(dtype=float);
npSalida = salida.to_numpy(dtype=float);
historial = modelo.fit(npEntrada, npSalida, epochs=100, verbose=False)
print('Entrenamiento completado !')
```

Una vez finalizado este paso, se asume que el modelo ya está entrenado y se procede a imprimir la curva de error de aprendizaje para ver cuantas vueltas le tomo aprender al algoritmo hasta que el margen de error sea bajo. Como resultado se tiene la siguiente figura la cual resume que a las 60 vueltas el modelo ya se encontraba entrenado.

Figura 50
Curva de aprendizaje del modelo



Lo siguiente es realizar una predicción con datos de entrada como se observa en la figura a continuación y como resultado se obtiene la cantidad de cada ítem que se estima que se vendería según los datos ingresados al algoritmo.

Figura 51
Prueba de predicción

```

] print('TEST PREDICCIÓN')
inputTest = np.array([[0,4,0,0,2,0,1,0,0,1,1,0,2,0,1,0,0,0,0,0,0,0,3]], dtype=float)
len(inputTest)
resultado = modelo.predict(inputTest)
print('Resultado: ' + str(resultado))

```

```

TEST PREDICCIÓN
1/1 [=====] - 0s 142ms/step
Resultado: [[ 2.0656576  0.41095454  1.4842575  0.78345734  1.3488255  1.2907778
 -0.05815858  1.1802838  0.14940788  0.09415695  1.0284318  0.8225372
  0.8616986  1.1730502  0.08988646  0.097459  0.84055054  1.0329899
 1.6049125  0.3333195 ]]

```

Uno de los pasos finales es guardar el modelo para que este pueda ser utilizado luego en otros directorios o proyectos.

Figura 52
Guardado de modelo

```

modelo.save('modelo_restaurante.h5')

```

```

!ls

```

```

entradas.csv  modelo_restaurante.h5  salidas.csv

```

Como paso semifinal, se exporta este modelo pero en formato de JavaScript para poder implementarlo en la aplicación desarrollada en este documento.

Figura 53
Exportación de modelo para JavaScript

```

!tensorflowjs_converter --input_format keras modelo_restaurante.h5 resultados/

```

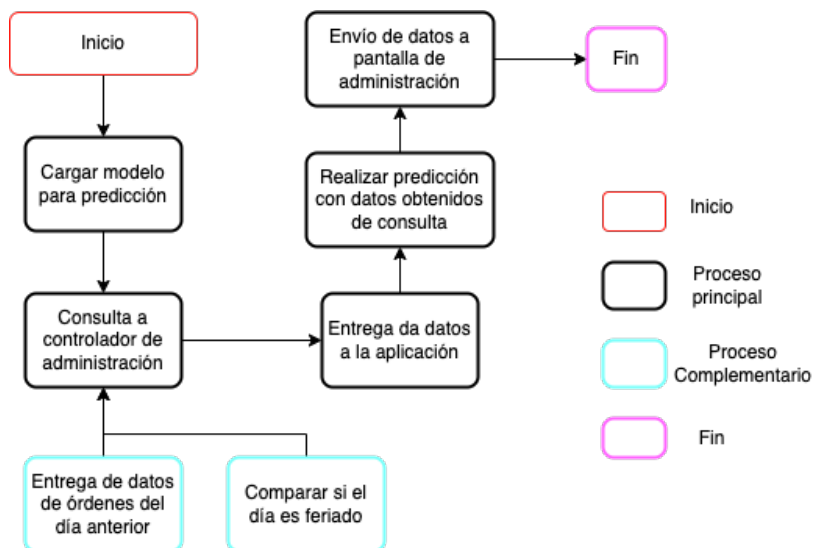
```

2023-02-14 07:25:14.645185: W tensorflow/compiler/xla/stream_executor/platform/default/
2023-02-14 07:25:14.645363: W tensorflow/compiler/xla/stream_executor/platform/default/
2023-02-14 07:25:14.645386: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-

```

Para poder realizar la importación del modelo entrenado en los pasos previos, se presenta la siguiente figura con el flujograma para ofrecer como resultado final la predicción de ventas como herramienta para el administrador del local.

Figura 54
Diagrama de flujo para implementación de red neuronal



El último paso es importar este modelo dentro del servidor como se muestra en el código a continuación donde se importa el modelo en la primera línea y luego se ejecuta la carga del modelo como similar a como si se estuviera utilizando la plataforma de Google Colab, pero en este caso se crean las funciones respectivas para utilizar el modelo, manipular los datos de la base de datos del sistema y aplicar el algoritmo ya entrenado para generar la predicción de venta. Esta predicción se puede observar en la sección de resultados del documento.

Figura 55
Importación de modelo de red neuronal en servidor

```

<script src="/js/tf.min.js"></script>
<script type="text/javascript">

  var modelo = null;
  var ventas = [];
  //Cargar modelo
  (async () => {
    console.log("Cargando modelo...");
    modelo = await tf.loadLayersModel("/model.json");
    console.log("Modelo cargado...");
  })();

```

Una vez importado el modelo, se procede con la consulta a la base de datos sobre la información de los productos. Estos datos serán solicitados por el servidor al cargar la página de administración mediante la función “obtenerDatos” que se muestra en la figura siguiente y se complementa con el controlado de administración (figura 29) con la función de resumen en conjunto para poder establecer el formato de los datos dentro del arreglo para poder ejecutar la predicción y obtener un resultado como se observa en la figura 51.

Figura 56
Función para obtener datos del sistema

```
function obtenerDatos(){
$.ajax({
  url: '/admin/resumen',
  method: 'GET',
  success: function(data){
    const jsonData = JSON.parse(data);
    console.log(jsonData)
    setTimeout(()=>{
      predecirVentas(jsonData);
    }, 1000)
  },
  error: function(err){
    console.log(err)
  }
})
}
```

Finalmente, con la función “predecirVentas” se procesan los datos obtenidos de la consulta a la base de datos con el controlador de Administración: Función resumen. Los resultados de la predicción utilizan la función “Math.ceil” para redondear los datos predictivos al mayor número cercano entero y luego llama a todos los resultados y los establece en un formato similar al input shape que se observa en la figura 47 y otro ejemplo es el arreglo de datos de la figura 51 que se usó para hacer el ejemplo de predicción. Luego de hacer la predicción, utiliza el identificador de cada ítem en la base de datos (ej: picanha=1) y los coloca dentro de una tabla para poder observar cual es la predicción que arroja el sistema en la pantalla como se puede observar en la sección de resultados, pruebas en ambiente real.

Figura 57
Función para predecir venta y mostrar en pantalla

```
function predecirVentas(data) {
  if (modelo !== null) {

    var tensor = tf.tensor2d([data]);
    var prediccion = modelo.predict(tensor).dataSync();
    prediccion = prediccion.map(p => {
      if(p < 0) p = 0;
      else p = Math.ceil(p, 0);
      return p;
    });

    ventas = Array.from(prediccion).map((p, index) => {
      const producto = productos.filter( producto => p
      producto.ventas = p;
      if(p > 0){
        let html = `<tr>
          <td>${producto.nombre}</td>
          <td>${producto.ventas}</td>
        </tr>`
        $("#predicciones").append(html);
      }
      return producto;
    });
    console.log(ventas);
  }
}
```

Capítulo IV: Pruebas de Funcionamiento y Análisis de Resultados

Este apartado contiene la descripción de las pruebas de funcionamiento y resultados obtenidos del desarrollo del proyecto desarrollado en este documento.

4.1. Pruebas de Funcionamiento

Esta sección tiene los detalles de las pruebas de funcionamiento realizadas al sistema una vez que se encuentre cerca de su etapa final para comprobar y corregir errores que se puedan obtener a lo largo del desarrollo del proyecto. Las pruebas a realizar en cada sección del sistema dependen de los diagramas de secuencia establecidos en las figuras 12 y 21 respectivamente donde se toman en cuenta las líneas de vida de los diagramas como escenarios de prueba para las etapas del sistema.

4.1.1. Nodo cliente

Para el nodo cliente se realizarán las siguientes pruebas que verificarán el funcionamiento del nodo con el sistema:

- Prueba 1: Conexión de nodo a red Wi-Fi (figura 58)
- Prueba 2: Proceso de autenticación del nodo con el servidor (figuras 59 a 61)
- Prueba 3: Solicitud de código de verificación para confirmación de pedido
- Prueba 4: Envío de solicitud de asistencia en mesa a servidor
- Prueba 5: Notificación de entrega de pedido completa

4.1.1.1. Prueba 1

Las pruebas que se van a realizar a este dispositivo van relacionadas a la conexión a la red y la comunicación entre el servidor y el nodo.

La figura que se muestra a continuación demuestra el mensaje exitoso de conexión a la red según la programación realizada.

Figura 58
Prueba de conexión de nodo a red



4.1.1.2. Prueba 2

Una vez conectado el nodo a la red, el nodo buscará conectarse al servidor con la dirección configurada dentro de este y como se observa en la figura siguiente, se logró obtener las capturas utilizando un sniffer llamado Wireshark para poder capturar las tramas de sincronización.

La primera solicita la comunicación con el servidor, la segunda es la respuesta del servidor al nodo para comunicarle que la comunicación y el puerto están habilitados y la tercera confirma el nodo el inicio de comunicación para enviar su petición.

Figura 59
Tramas de inicio de comunicación de nodo a servidor

Source	Destination	Protocol	Info
192.168.1.20	192.168.1.123	TCP	61425 → 80 [SYN] Seq=0 Win=5744 Len=0 MSS=1436
192.168.1.123	192.168.1.20	TCP	80 → 61425 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
192.168.1.20	192.168.1.123	TCP	61425 → 80 [ACK] Seq=1 Ack=1 Win=5744 Len=0

Luego, el nodo continúa su proceso y realiza la solicitud de su código de autenticación para la red. Las tramas capturadas para el proceso de autenticación se observan en la imagen a continuación donde la primera trama es el envío de la petición por parte del nodo para registrarse en el servidor como se ve en el recuadro azul de la imagen a continuación. El servidor responde con la solicitud aceptada le envía esta respuesta el nodo. Finalmente, el servidor realiza su proceso

interno de generar un token utilizando el controlador “LOGIN” y así enviar una respuesta al nodo con el token que se observa en el recuadro verde de la figura siguiente.

Figura 60
Tramas de proceso de autenticación de nodo con servidor

The screenshot shows a network traffic capture with the following details:

- Frame 32: HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
- Frame 32 details: 647 bytes on wire (5176 bits), 647 bytes captured (5176 bits) on interface eth0, id 0
- Ethernet II, Src: Raspberr_Se:91:9b (e4:5f:01:5e:91:9b), Dst: Espressi_25:cf:c4 (34:94:54:25:cf:c4)
- Internet Protocol Version 4, Src: 192.168.1.123, Dst: 192.168.1.20
- Transmission Control Protocol, Src Port: 80, Dst Port: 61425, Seq: 1, Ack: 193, Len: 593
- Hypertext Transfer Protocol
- JavaScript Object Notation: application/json
- Object:
 - Member: token
 - [Path with value [truncated]: /token:eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJSRVNUQVVSQSU5URSIImF1ZC16IjF1R1RlR1Iiwic3Vl...]
 - [Member with value [truncated]: token:eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJSRVNUQVVSQSU5URSIImF1ZC16IjF1R1RlR1Iiwic3Vl...]
 - [String value [truncated]: eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJSRVNUQVVSQSU5URSIImF1ZC16IjF1R1RlR1Iiwic3Vl...]
 - Key: token
 - [Path: /token]
 - Member: expires_at

Una vez finalizado este proceso, esta muestra en la pantalla el logotipo de la empresa como se observa en la figura 20 de este documento. El código de autenticación obtenido será utilizado para incluirlo en la cabecera de datos enviados al servidor en cada comunicación establecida. Si el nodo no es autenticado de manera correcta, las comunicaciones serán anuladas.

Para finalizar el proceso de comunicación, el nodo inicia el proceso de finalizar la comunicación con el nodo y como se observa en las tramas de la figura siguiente.

Figura 61
Tramas de finalización de comunicación entre nodo y servidor

192.168.1.20	192.168.1.123	TCP	61426 → 80 [FIN, ACK] Seq=435 Ack=424 Win=5321 Len=0
192.168.1.123	192.168.1.20	TCP	80 → 61426 [FIN, ACK] Seq=424 Ack=436 Win=63806 Len=0
192.168.1.20	192.168.1.123	TCP	61426 → 80 [ACK] Seq=436 Ack=425 Win=5320 Len=0
192.168.1.20	192.168.1.123	TCP	61425 → 80 [FIN, ACK] Seq=193 Ack=594 Win=5151 Len=0
192.168.1.123	192.168.1.20	TCP	80 → 61425 [FIN, ACK] Seq=594 Ack=194 Win=64048 Len=0

4.1.1.3. Prueba 3

La siguiente prueba de funcionamiento es realizar la solicitud de un código de verificación para confirmar la realización de un pedido.

A continuación el nodo vuelve a iniciar una comunicación con el servidor utilizando el mismo proceso de tramas que se observa en la figura 54 para establecer la sincronización. Esta vez, para pedir un código, se realiza la petición a otro controlador del servidor el cual se encarga

Figura 63
Tramas de respuesta con código de verificación para pedido de cliente

```

176 192.168.1.123 192.168.1.20 TCP 80 → 61427 [ACK] Seq=1 Ack=435 Win=63806 Len=0 Default
178 192.168.1.123 192.168.1.20 HTTP/JSON HTTP/1.1 200 OK , JavaScript Object Notation (application/json) Default
> Frame 178: 477 bytes on wire (3816 bits), 477 bytes captured (3816 bits) on interface eth0, id 0
> Ethernet II, Src: Raspberr_5e:91:9b (e4:5f:01:5e:91:9b), Dst: Espressi_25:cf:c4 (34:94:54:25:cf:c4)
> Internet Protocol Version 4, Src: 192.168.1.123, Dst: 192.168.1.20
> Transmission Control Protocol, Src Port: 80, Dst Port: 61427, Seq: 1, Ack: 435, Len: 423
> Hypertext Transfer Protocol
< JavaScript Object Notation: application/json
  Object
  > Member: id
  > Member: inicio
  > Member: fin
  < Member: token
    [Path with value: /token:477470]
    [Member with value: token:477470]
    Number value: 477470
    Key: token
    [Path: /token]
  < Member: id_mesa
    [Path with value: /id_mesa:1]
    [Member with value: id_mesa:1]
    String value: 1
    Key: id_mesa
    [Path: /id_mesa]

```

Finalmente el nodo almacena toda la información recibida y vuelve a cerrar comunicación con el servidor como se observa en las tramas de la figura 56. Con la información guardada, el nodo procede a separar los datos para seleccionar el código recibido y lo muestra en pantalla por un minuto como se observa en la figura a continuación para que el cliente pueda ingresar este código en su pantalla y así finalizar su pedido.

Figura 64
Solicitud de código de verificación para completar orden



4.1.1.4. Prueba 4

La siguiente prueba es notificar al servidor que una mesa requiere asistencia, para ello el nodo solicita nuevamente iniciar una comunicación como se ha observado en los otros pasos para

poder enviar su petición. Una vez iniciada esta comunicación, el nodo realiza el envío de la petición como se observa en la figura siguiente donde se logró capturar la trama. Nuevamente se observa la trama de solicitud pero esta incluye en su cabecera, al igual que las otras tramas de solicitud, el campo de autorización con su código de verificación obtenido en el proceso de autenticación para que el servidor pueda verificar el nodo y la mesa a la que corresponde.

Figura 65
Envío de petición de asistencia de nodo a servidor

```

240 192.168.1.20 192.168.1.123 HTTP GET /mesa/alerta HTTP/1.1
Frame 240: 495 bytes on wire (3960 bits), 495 bytes captured (3960 bits) on interface eth0, id 0
Ethernet II, Src: Espressi_25:cf:c4 (34:94:54:25:cf:c4), Dst: Raspberr_5e:91:9b (e4:5f:01:5e:91:9b)
Internet Protocol Version 4, Src: 192.168.1.20, Dst: 192.168.1.123
Transmission Control Protocol, Src Port: 61428, Dst Port: 80, Seq: 1, Ack: 1, Len: 441
Hypertext Transfer Protocol
> GET /mesa/alerta HTTP/1.1\r\n
Host: 192.168.1.123\r\n
User-Agent: ESP32HTTPClient\r\n
Connection: keep-alive\r\n
Accept-Encoding: identity;q=1,chunked;q=0.1,*;q=0\r\n
Content-Type: application/json\r\n
[truncated]Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJJSRVNUQVVSQU5URSIsIn
\r\n
[Full request URI: http://192.168.1.123/mesa/alerta]
[HTTP request 1/1]
[Response in frame: 242]

```

El servidor recibe la petición y notifica al nodo con su trama de ACK que el mensaje ha sido recibido.

Figura 66
Trama de ACK de servidor a nodo

```

260 192.168.1.123 192.168.1.20 TCP 80 -> 61429 [ACK] Seq=1 Ack=442 Win=63799 Len=0

```

El nodo al haber enviado la solicitud muestra en pantalla que la notificación ha sido recibido al usuario con el mensaje “En camino” como se observa en la figura siguiente.

Figura 67
Envío de solicitud de asistencia en mesa



Para completar este proceso, el personal del establecimiento se acercará a la mesa para brindar asistencia y una vez completado el requerimiento volverá a presionar el botón de asistencia para notificarle al servidor que ya ha sido atendido el requerimiento del cliente. El mensaje se borrará y mostrará nuevamente el logotipo del restaurante como se observa en la figura 20.

4.1.1.5. Prueba 5

La última prueba correspondiente al nodo es establecer una comunicación con el servidor para notificarle que el pedido ha sido entregado y esta acción será ejecutada presionando el botón con la letra “S” por parte del personal del establecimiento. De esta forma el nodo mostrará en su pantalla el mensaje de que se ha entregado su notificación al servidor y el servidor actualizará el estado del pedido de esa mesa a atendidos. Si este botón vuelve a ser presionado, la solicitud será rechazada ya que el filtro configurado en el servidor establece que solo pueda realizarse una vez dado que existe un código único generado para cada pedido.

Figura 68
Envío de notificación de entrega de pedido completo



En la figura a continuación se puede observar la trama capturada del envío de notificación de pedido completo donde se observa la trama con su campo de autorización con el código de autenticación del nodo y más adelante se observa un campo: “full request URI: http/192.168.1.123/mesa/recibir/477470”. Este enlace mantiene la estructura de: dirección de servidor / controlador/ función/ token. El token es el código de verificación que el cliente ingresó dentro de su pedido para poder notificarle al servidor que el pedido con ese token está completo y en la pantalla de administración podrá notificar al usuario administrador que el pedido de la mesa remitente ha sido entregado.

Figura 69
Trama de notificación de pedido entregado entre nodo y servidor

```

+-----+-----+-----+-----+-----+-----+
| 277 | 192.168.1.20 | 192.168.1.123 | HTTP | GET /mesa/recibir/477470 | HTTP/1.1 |
+-----+-----+-----+-----+-----+-----+
> Frame 277: 503 bytes on wire (4024 bits), 503 bytes captured (4024 bits) on interface eth0, id 6
> Ethernet II, Src: Espressi_25:cf:c4 (34:94:54:25:cf:c4), Dst: Raspberr_5e:91:9b (e4:5f:01:5e:91:9b)
> Internet Protocol Version 4, Src: 192.168.1.20, Dst: 192.168.1.123
> Transmission Control Protocol, Src Port: 61430, Dst Port: 80, Seq: 1, Ack: 1, Len: 449
< Hypertext Transfer Protocol
  > GET /mesa/recibir/477470 HTTP/1.1\r\n
    Host: 192.168.1.123\r\n
    User-Agent: ESP32HTTPClient\r\n
    Connection: keep-alive\r\n
    Accept-Encoding: identity;q=1,chunked;q=0.1,*;q=0\r\n
    Content-Type: application/json\r\n
    [truncated]Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJJSRVVUQVVSQU
    \r\n
    [Full request URI: http://192.168.1.123/mesa/recibir/477470]
    [HTTP request 1/1]
    [Response in frame: 279]
  
```

Es importante recalcar que si en algún momento el nodo pierde conexión o la autenticación de este no fue realizada de manera correcta, ninguna de las operaciones será ejecutada y todas las

comunicaciones serán ignoradas por parte del servidor hacia el dispositivo que no se ha autenticado previamente.

4.1.2. Servidor

Las pruebas de servidor serán orientadas a la vista y experiencia por parte del cliente y desde la ventana de administración para controlar los pedidos y productos del establecimiento. Las pruebas a realizar se enlistan a continuación.

- Prueba 1: Acceso al servidor mediante navegador web
- Pruebas 2: Realización de pedido por parte de cliente
- Prueba 3: Agregar datos de facturación
- Prueba 4: Confirmación de orden de cliente con código correcto a incorrecto de validación
- Prueba 5: Acceso a panel de administración
- Prueba 6: Manipulación de ítems del establecimiento gastronómico
- Prueba 7: Recepción de pedidos en panel
- Prueba 8: Recepción de solicitud de asistencia de una mesa en panel
- Prueba 9: Visualizar pedido generado por cliente en panel e impresión
- Prueba 10: Procesamiento y confirmación de entrega de pedido a cliente

4.1.2.1. Prueba 1

Tanto para administración como para clientes, la conexión con el servidor se debe realizar por la pantalla de ingreso bienvenida y el servidor mostrará la página diseñada para la bienvenida de los usuarios en la cual se debe seguir navegando para poder continuar con las opciones configuradas en el servidor.

Figura 70
Pantalla de bienvenida para clientes y administrador

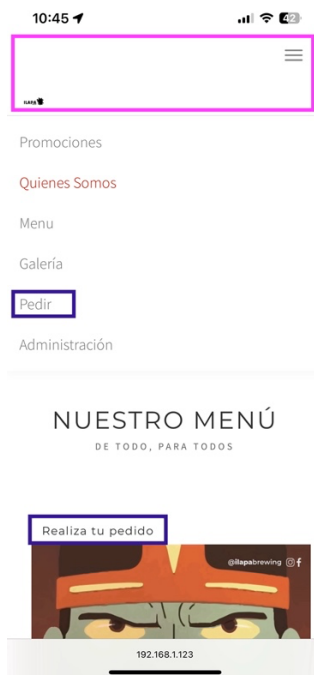


4.1.2.2. Prueba 2

Este apartado brinda los detalles de las pruebas realizadas en una simulación como cliente hacia el sistema para realizar un pedido.

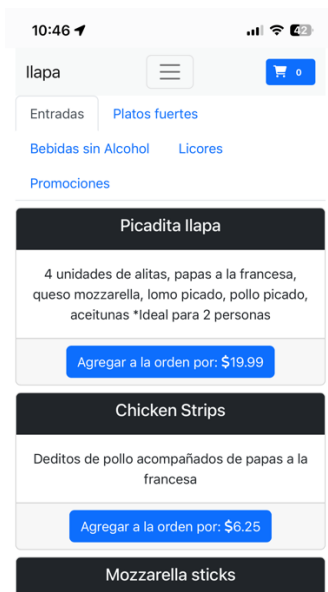
Una vez que se avance en la pantalla de bienvenida, el cliente se encontrará con diferentes opciones para conocer un poco más sobre las promociones e información del local sobre su historia, etc. Dentro de las opciones de navegación se encontrará presente una barra de navegación conocida como “Navbar” en HTML (recuadro rosado de la siguiente figura) y en ella podrá seleccionar los enlaces pertinentes para realizar el pedido si es un cliente concurrente y sabe que ofrece el establecimiento, caso contrario puede encontrar el botón de “Realizar tu pedido” al inicio y al final del menú presentado en la ventana de bienvenida del sistema (recuadros morados son los enlaces a la ventana de pedidos).

Figura 71
Ventana de inicio en navegación de vista cliente



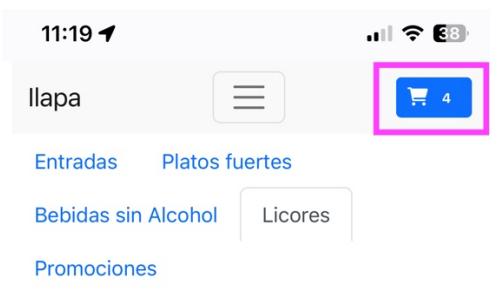
Dentro del enlace para realizar el pedido, el cliente se encontrará con los ítems que ofrece el establecimiento (dos figuras siguientes) pero esta vez podrá agregarlos a un carrito de compras cada vez que se presione el botón de “Agregar a la orden por...”. Además, los ítems del establecimiento se encuentran clasificados por su tipo para una mejor navegación para el cliente.

Figura 72
Ventana para realizar pedidos para clientes



En la figura siguiente se puede apreciar en el recuadro rosado el carrito de compras indicando que ya existen ítems agregados.

Figura 73
Carrito de compras con ítems agregados



Una vez que el cliente haya finalizado su pedido, este puede acceder al carrito de compras y observar el resumen de su pedido para aumentar ítems, borrarlos, regresar al menú de configuración o agregar observaciones para su pedido. Además, se muestra un mensaje en rojo notificando de que por defecto los datos de facturación serán en consumidor final y que para editarlos se debe presionar el botón de “Datos de Facturación”.

Figura 74
Carrito de compras de cliente

Carrito de compras

Has agregado al carrito:

1	Picadita llapa	-	+
1	Chicken Strips	-	+
1	Animal fries	-	+
1	Té helado	-	+

Total: 37.49 \$

Observaciones de tu pedido

Por defecto los comprobantes saldrán a consumidor final.
Puedes cambiar los datos en los siguientes campos a continuación

Datos de Facturación

Continuar pedido Confirmar orden

4.1.2.3. Prueba 3

Los datos de facturación se encuentran por defecto en consumidor final, pero al dar un click en el campo de “Número de cédula o RUC” los datos del campo se borrarán y el cliente ingresará sus datos.

Figura 75
Sección de facturación para pedido

Por defecto los comprobantes saldrán a consumidor final.
Puedes cambiar los datos en los siguientes campos a continuación

Datos de Facturación

Número de Cedula o RUC
999999999

Nombre y apellido
Consumidor final

Dirección
Ibarra

Teléfono
999999999

Correo electrónico
correo@ejemplo.com

Continuar pedido Confirmar orden

Una vez que el cliente ingrese sus datos, el sistema realizará una verificación al detectar 10 dígitos para cédula y 13 dígitos para RUC. Si el cliente existe, el sistema lo encontrará y este

publicará el resto de sus datos en el resto de los campos para verificar que estos sean correctos y continuar con el pedido.

Figura 76
Cliente encontrado en base de datos de servidor

Por defecto los comprobantes saldrán a consumidor final.
Puedes cambiar los datos en los siguientes campos a continuación

Datos de Facturación

Número de Cedula o RUC

Nombre y apellido

Dirección

Teléfono

Correo electrónico

En caso de que el cliente no exista, aparecerá un mensaje en rojo notificando que no está registrado y a su vez un botón para registrarse. Luego de esto el cliente estará listo para poder continuar con su confirmación de orden.

Figura 77
Cliente no encontrado en base de datos de servidor

Datos de Facturación

Número de Cedula o RUC

Nombre y apellido

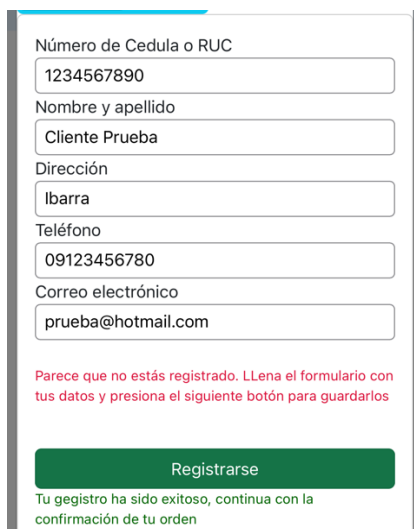
Dirección

Teléfono

Correo electrónico

Parece que no estás registrado. Presiona el siguiente botón para guardar tus datos

Figura 78
Cliente registrado con éxito



Número de Cedula o RUC
1234567890

Nombre y apellido
Cliente Prueba

Dirección
Ibarra

Teléfono
09123456780

Correo electrónico
prueba@hotmail.com

Parece que no estás registrado. Llena el formulario con tus datos y presiona el siguiente botón para guardarlos

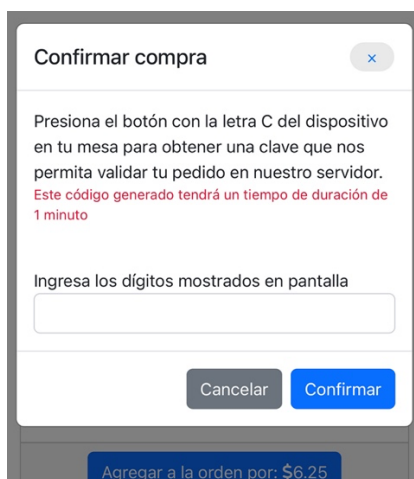
Registrarse

Tu registro ha sido exitoso, continua con la confirmación de tu orden

4.1.2.4. Prueba 4

Para finalizar la toma del pedido, el cliente debe seguir las instrucciones mostradas en su pantalla.

Figura 79
Solicitud de código de verificación y finalización de pedido



Confirmar compra

Presiona el botón con la letra C del dispositivo en tu mesa para obtener una clave que nos permita validar tu pedido en nuestro servidor. Este código generado tendrá un tiempo de duración de 1 minuto

Ingresa los dígitos mostrados en pantalla

Cancelar Confirmar

Agregar a la orden por: \$6.25

Una vez que ha ingresado el código de verificación obtenido por el nodo cliente como se observa en la sección 4.1.1, si este código es correcto, el pedido será enviado al servidor para poder mostrarlo en la pantalla de administración, caso contrario la solicitud será denegada y el servidor ignorará las peticiones hasta que el código sea correcto.

Figura 80
Código de verificación incorrecto

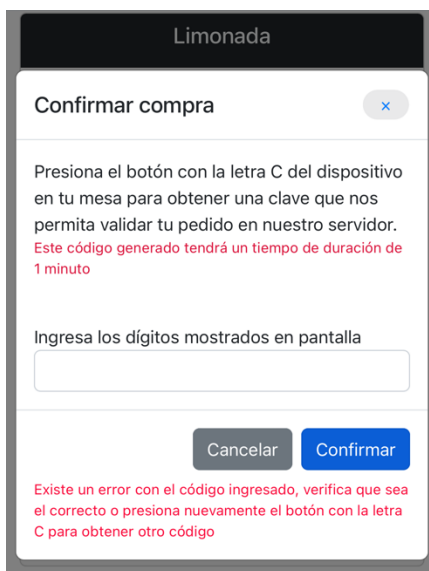


Figura 81
Pedido recibido con éxito



4.1.2.5. Prueba 5

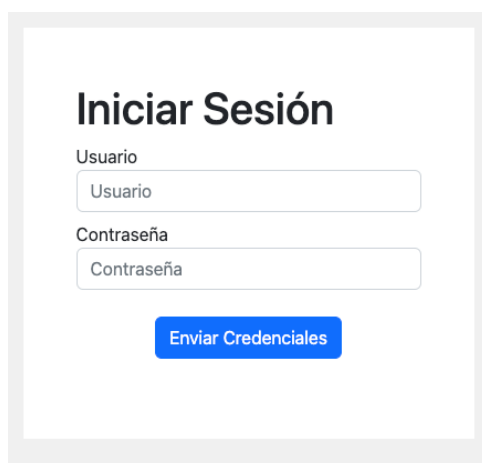
La vista para el panel de administración será diferente al tener un acceso al servidor a través de otro dispositivo el cual puede ser una tableta o un computador. En el navbar mencionado al inicio de la sección anterior y mostrado a continuación pero con vista desde computador, se observa la opción de “Administración”.

Figura 82
Enlace a ventana de acceso de administración

Promociones Quienes Somos Menu Galería Pedir **Administración**

La opción de administración realiza un direccionamiento hacia una página para ingresar credenciales de acceso como medida de seguridad para evitar el acceso de usuarios no autorizados al panel del control.

Figura 83
Solicitud de credenciales para ingreso a panel de administración



Formulario de inicio de sesión con el título "Iniciar Sesión". Incluye campos de entrada para "Usuario" y "Contraseña", y un botón azul que dice "Enviar Credenciales".

El sistema cuenta con un mecanismo de verificación para controlar si las credenciales ingresadas son las correctas, caso contrario mantiene al usuario en la ventana de ingreso.

Figura 84
Ingreso denegado a panel de control por credenciales erróneas



Formulario de inicio de sesión con el título "Iniciar Sesión". Incluye un mensaje de error en un recuadro rojo que dice "Usuario o contraseña incorrectos". Incluye campos de entrada para "Usuario" y "Contraseña", y un botón azul que dice "Enviar Credenciales".

Si las credenciales son correctas, se procede al direccionamiento del panel de control en donde se tendrá como pantalla inicial la correspondiente a manipulación de ítems del establecimiento.

4.1.2.6. Prueba 6

En la esquina superior derecha de la siguiente figura se observa en el recuadro rosado dos botones que permiten editar (botón amarillo) o eliminar el ítem (botón rojo). Si se desea ingresar un nuevo ítem, se procede a llenar los campos en la parte izquierda y al finalizar una edición o un ingreso de variables se procede a enviar estos datos a la base de datos del sistema.

Figura 85
Ventana de inicio de panel de control

The screenshot shows a web interface for product management. At the top, there is a navigation bar with 'llapa Inicio Pedidos' on the left and 'Hola admin Cerrar Sesión' on the right. Below this is a blue header for 'Gestion de producto'. On the left side, there is a form with fields for 'Nombre', 'Precio', 'Descr', and 'Tipo', and a 'Guardar' button. On the right side, there is a table with the following data:

Id	Nombre	Precio	Descripción	Tipo	Acciones
1	Picadita llapa	19.99	4 unidades de alitas, papas a la francesa, queso mozzarella, lomo picado, pollo picado, aceitunas *Ideal para 2 personas	ent	[Editar] [Eliminar]
2	Animal fries	8.75	Papas a la francesa acompañadas de cerdo mechado en salsa BBQ y salsa llapa	fr	[Editar] [Eliminar]
6	Chicken Strips	6.25	Deditos de pollo acompañados de papas a la francesa	ent	[Editar] [Eliminar]
9	Mozzarella sticks	5.5	Dedos de mozzarella apanados	ent	[Editar] [Eliminar]
8	Chicken llapa	8.5	Sanduche de pollo crispy, queso mozzarella, tomate, lechuga, pickles	fr	[Editar] [Eliminar]
3	Pulled Pork	9.75	Cerdo mechado bbq, mozzarella y coleslaw	fr	[Editar] [Eliminar]

La figura a continuación realiza la prueba de ingreso de un nuevo artículo a la base de datos.

Figura 86
Creación de nuevo ítem en panel de control



The screenshot shows the 'Gestion de producto' control panel with the form for creating a new item. The fields are filled with the following data:

Nombre: prueba
Precio: 100
Descr: es una prueba para doc
Tipo: ent

The 'Guardar' button is visible at the bottom right of the form.

La imagen siguiente demuestra que el ítem se ingresó a la base de datos correctamente

Figura 87
Ítem creado en panel de control

79	prueba	100	es una prueba para documentar	ent		
----	--------	-----	-------------------------------	-----	---	---

Tras realizar una edición a la descripción del ítem de prueba y guardar los cambios se observa que la operación se realizó con éxito.

Figura 88
Ítem editado en panel de control

79	prueba	100	es una prueba para editar	ent		
----	--------	-----	---------------------------	-----	---	---

Finalmente, de forma automática el sistema muestra el nuevo ítem ingresado (recuadro rosado) en la ventana de clientes para que puedan realizar el pedido como se observa a continuación.

Figura 89
Ítem editado y mostrado en panel de clientes

Ilapa Inicio

Entradas **Platos fuertes** Bebidas sin Alcohol Licores Promociones

<p>Picadita Ilapa</p> <p>4 unidades de alitas, papas a la francesa, queso mozzarella, lomo picado, pollo picado, aceitunas *Ideal para 2 personas</p> <p>Agregar a la orden por: \$ 19.99</p>	<p>Chicken Strips</p> <p>Deditos de pollo acompañados de papas a la francesa</p> <p>Agregar a la orden por: \$ 6.25</p>	<p>Mozzarella sticks</p> <p>Dedos de mozzarella apanados</p> <p>Agregar a la orden por: \$ 5.50</p>	<p>prueba</p> <p>es una prueba para editar</p> <p>Agregar a la orden por: \$ 100.00</p>
--	--	--	--

En la sección de pedidos, se obtiene la siguiente visualización de datos:

- Izquierda: información sobre el estado de las mesas y una descripción.
- Derecha: información sobre los pedidos realizados por los clientes con pestañas de pendientes, atendidos y cancelados

4.1.2.7. Prueba 7

En la figura siguiente se observa además el pedido que se realizó previamente para comprobar el funcionamiento de la parte de cliente junto con la de comunicación del nodo con el servidor.

Figura 90
Panel de control de pedidos para administración

The screenshot shows a web interface for order management. On the left, there are two green cards representing tables: M-01 (Mesa ubicada a la derecha de caja) and M-02 (Mesa ubicada a la izquierda de caja). On the right, there is a 'Gestion de pedidos Pendientes' section with a table showing one pending order.

Id	Tiempo	Hora	Mesa	Total	Observaciones	Acciones
26	64	2023-02-25 02:18:14	M-01	37.49		Procesar Cancelar Ver

4.1.2.8. Prueba 8

Suponiendo que la mesa 1 requiera ayuda y esta aplastó el botón de solicitud de asistencia, la pantalla de administración indicará en rojo aquella mesa que requiera atención como muestra la figura siguiente. Una vez que este requerimiento sea atendido, el personal del establecimiento presionará el botón nuevamente de asistencia y la mesa notificará al sistema que se ha cumplido con la asistencia para volver a su estado anterior como se observa en la figura anterior.

Figura 91
Notificación de solicitud de asistencia mostrada en panel de control

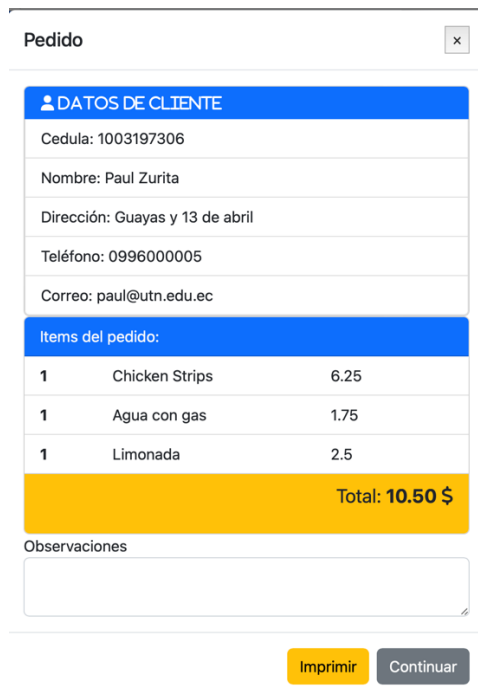
The screenshot shows the same web interface as Figure 90, but with the M-01 table card highlighted in red to indicate it needs attention. The 'Gestion de pedidos Pendientes' table now shows a different order.

Id	Tiempo	Hora
26	119	2023-02-25 02:18:14

4.1.2.9. Prueba 9

El administrador puede revisar el pedido que hizo el cliente para poder imprimirlo y enviarlo a cocina para cambiar el estado del pedido en la pantalla a “Procesando”

Figura 92
Observación de pedido realizado por cliente en panel de control



The screenshot shows a window titled "Pedido" with a close button (X) in the top right corner. The content is organized into several sections:

- DATOS DE CLIENTE** (Customer Data): A blue header section containing the following information:
 - Cedula: 1003197306
 - Nombre: Paul Zurita
 - Dirección: Guayas y 13 de abril
 - Teléfono: 0996000005
 - Correo: paul@utn.edu.ec
- Items del pedido:** (Order Items): A blue header section containing a table of items:

Cantidad	Item	Precio
1	Chicken Strips	6.25
1	Agua con gas	1.75
1	Limonada	2.5
- Total:** A yellow bar at the bottom of the items section showing "Total: 10.50 \$".
- Observaciones:** (Observations): A text input field with a placeholder and a small icon in the bottom right corner.
- Botones:** At the bottom right, there are two buttons: "Imprimir" (Print) in a yellow box and "Continuar" (Continue) in a grey box.

Dentro de la figura anterior se observa al final los botones de continuar e imprimir. Si el administrador desea imprimir se desplegará una ventana nueva con toda la información del pedido del cliente lista para imprimir en el formato de las impresoras termales.

Figura 93
Previsualización de impresión de pedido de cliente

Cod	Descripcion	Cantidad	Precio Unitario	Descuento	Total
107	Chicken Strips	1	6.25	0.00	6.25
108	Agua con gas	1	1.75	0.00	1.75
109	Limonada	1	2.50	0.00	2.5
				Descuento	0.00
				Propina:	0.00
				VALOR TOTAL	10.50

4.1.2.10. Prueba 10

El administrador al ingresar el pedido en cocina debe presionar el botón de procesar en la pantalla y el pedido cambiará a un estado de procesamiento como se observa a continuación.

Figura 94
Pedido recibido sin procesar

Gestion de pedidos Pendientes						
Id	Tiempo	Hora	Mesa	Total	Observaciones	Acciones
17	112	2023-02-13 22:47:53	M-01	37.24		Procesar Cancelar Ver

Figura 95
Pedido en estado de procesamiento en panel de control

Gestion de pedidos Pendientes						
Id	Tiempo	Hora	Mesa	Total	Observaciones	Acciones
26	183	2023-02-25 02:18:14	M-01	37.49		Despachar Cancelar Ver

Una vez que el personal notifique que último ítem solicitado por el cliente está siendo despachado, el administrador debe presionar el botón de despachar para colocar el pedido en estado de despacho.

Figura 96
Pedido en estado de despacho en panel de control

Id	Tiempo	Hora	Mesa	Total	Observaciones	Acciones
26	200	2023-02-25 02:18:14	M-01	37.49		Cancelar Ver

Una vez que el personal del establecimiento haya confirmado la entrega del pedido, este presionará el botón de la mesa con la letra “S” para proceder con el envío de notificación de pedido completado y así el servidor recibirá esta notificación para ese pedido específico colocarlo en la ventana de pedidos atendidos.

Figura 97
Pedido redirigido a la ventana de atendidos en panel de control

Id	Tiempo	Hora	Mesa	Total	Observaciones	Acciones
26	5	2023-02-25 02:18:14	M-01	37.49		Cancelar Ver

De esta forma se complementa la sección de manipulación de pedidos en la sección de servidor y de pruebas de funcionamiento en ambiente de desarrollo y la siguiente tabla contiene información sobre las pruebas realizadas con su descripción y si cumple o no.

Tabla 18
Resultados de pruebas realizadas en ambiente de desarrollo

Prueba	Descripción	Cumple	No Cumple
Nodo			
1	Conexión de nodo a red Wi-Fi	X	
2	Proceso de autenticación del nodo con el servidor	X	

3	Solicitud de código de verificación para confirmación de pedido	X
4	Envío de solicitud de asistencia en mesa a servidor	X
5	Notificación de entrega de pedido completa	X
Servidor		
1	Acceso al servidor mediante navegador web	X
2	Realización de pedido por parte de cliente	X
3	Agregar datos de facturación	X
4	Confirmación de orden de cliente con código correcto a incorrecto de validación	X
5	Acceso a panel de administración	X
6	Manipulación de ítems del establecimiento gastronómico	X
7	Recepción de pedidos en panel	X
8	Recepción de solicitud de asistencia de una mesa en panel	X
9	Visualizar pedido generado por cliente en panel	X
10	Procesamiento y confirmación de entrega de pedido a cliente	X

4.2. Pruebas de Funcionamiento en Ambiente Real

Las pruebas de ambiente real serán similares a las pruebas de funcionamiento previas en ambiente de desarrollo y se listan a continuación:

- Prueba 1: Análisis y ubicación de enrutador en posición adecuada
- Prueba 2: Verificar la predicción de venta
- Prueba 3: Realización de pedido con cliente en la mesa más lejana del local
- Prueba 4: Comprobación del funcionamiento del botón de asistencia
- Prueba 5: Comprobación de recepción de pedido
- Prueba 6: Repetir todo el proceso con otro cliente
- Prueba 7: Comprobación de predicción para el siguiente día

4.2.1. Prueba 1

Para las pruebas en ambiente real se realizó primero un análisis de la ubicación de las mesas y se buscó el punto más lejano de la mesa, la cual está aproximadamente a 20 metros de distancia del punto de venta del local, casi junto a la puerta de entrada al establecimiento.

Figura 98
Vista del establecimiento gastronómico del área de clientes



Para la ubicación del enrutador se tomó colocó en una sección del counter con las antenas apuntando en diferentes posiciones para cubrir la mayor área posible con la red inalámbrica. Además, se observa que solo existe un cable para poder conectar el servidor al enrutador de manera cableada y esto garantiza que el servidor funcionará sin internet.

Figura 99
Ubicación de enrutador en counter de establecimiento



4.2.2. Prueba 2

Una vez iniciado el sistema, se ingresó al servidor de forma remota utilizando la red inalámbrica y una laptop. Dentro del panel de configuración se observó la pestaña de predicción, la que arrojó como resultado la figura siguiente sobre las ventas que se planeaba tener para el día jueves 23 de febrero de 2023.

Figura 100
Predicción de venta para el día de prueba

Producto	Cantidad
Burrito Pollo	1
Matambre	1
Mojito	1
Agua sin gas	1
Pilsener 330ml	1
Pilsener 750ml	2
Club 330ml	2
Club 750ml	2
Chicken Strips	2
Bife de chorizo	1
Té helado	1
Nachos Simples	2

4.2.3. Prueba 3

Lo siguiente fue esperar a que llegue un par de clientes que estén dispuestos a probar el sistema. Para el primer caso llegó una pareja que accedió a ser parte de la prueba del sistema. En la figura siguiente se observa al cliente observando la pantalla de bienvenida del sistema

Figura 101
Cliente observando pantalla de bienvenida para usuarios



Luego de haber tomado la decisión sobre el pedido, el cliente pasó a la siguiente etapa para realizar el pedido en la sección respectiva como se observa en la figura siguiente.

Figura 102
Cliente realizando pedido



Una vez realizado el pedido, el cliente procedió a confirmar su orden revisando todos sus ítems una vez más y en este caso no deseaba factura por lo que omitió los datos de facturación.

Figura 103
Cliente revisando pedido antes de confirmar orden



El siguiente paso fue solicitar el código de verificación al servidor a través del nodo para completar la orden. La figura siguiente describe esta acción.

Figura 104
Cliente solicitando código de verificación



4.2.4. Prueba 4

Luego de hacer el pedido, el cliente solicitó ayuda al personal por lo que utilizó la opción de solicitar asistencia sin tener que levantarse de la mesa o buscar al personal del establecimiento, quien en se encontraba ocupado en la cocina y fue notificado por el administrador para atender el requerimiento del cliente. Una vez atendido el requerimiento, el mesero presionó el botón de asistencia nuevamente para notificar que se ha cumplido con el requerimiento.

Figura 105
Cliente atendido tras haber solicitado asistencia con el nodo cliente



Figura 106
Solicitud de asistencia recibida por servidor

Id	Tiempo	Hora	Mesa	Total
20	47	2023-02-23 20:59:11	M-02	16.74

La siguiente figura es una referencia de la distancia a la que se encontraba la mesa del cliente, era la última mesa por lo que se logra demostrar el funcionamiento al rango máximo de cobertura planteado en los requerimientos del sistema.

Figura 107
Imagen de referencia de distancia de la mesa de cliente

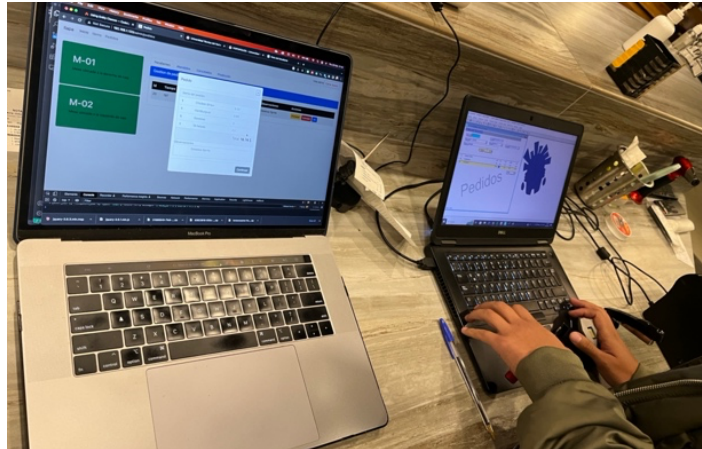


4.2.5. Prueba 5

Una vez enviado el pedido por el cliente, el servidor detectó el pedido nuevo y lo presentó en la pantalla de administración. Este fue revisado y se procedió a ejecutar la orden de cocina y facturación para el registro oficial en el sistema actual del establecimiento. Es importante

mencionar que el acceso al servidor se está realizando de manera remota a través del computador de la izquierda de la figura a continuación.

Figura 108
Pedido recibido en servidor



4.2.6. Prueba 6

Luego, el propietario del establecimiento decidió probar el sistema como se observa en las figuras a continuación. El procedimiento es el mismo y la descripción de cada acción se observa en la descripción de cada figura.

Figura 109
Cliente 2 navegando en pantalla de bienvenida del sistema



En esta figura siguiente se puede apreciar el funcionamiento del nodo utilizando una batería de 9000 mAh.

Figura 110

Cliente 2 siendo atendido por personal tras solicitar ayuda con el nodo



Figura 111

Cliente 2 realizando confirmación de pedido con código de verificación



En la pestaña de pedidos atendidos se observa todas las ordenes realizadas del día como se observa en la figura siguiente.

Figura 112

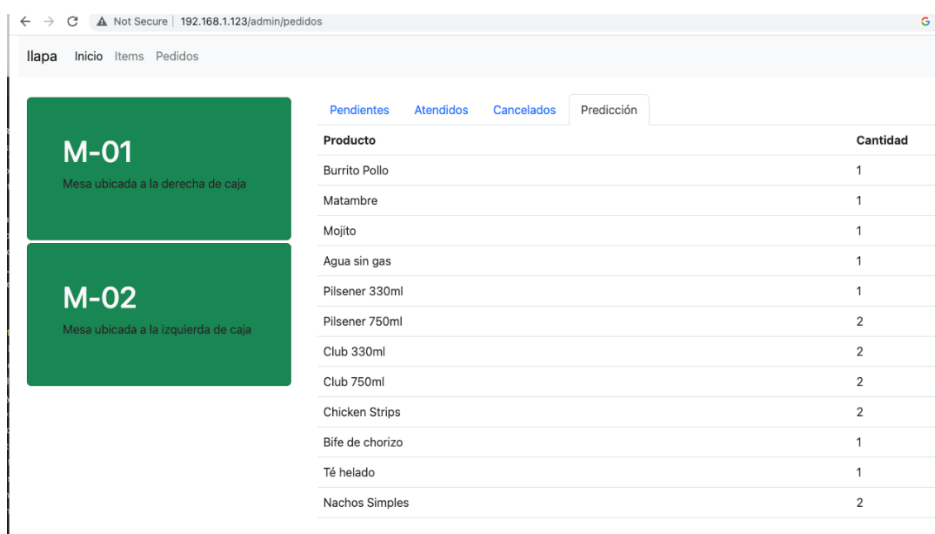
Pedidos atendidos del día de prueba

Id	Tiempo	Hora	Mesa	Total	Observaciones	Acciones
21	44	2023-02-23 21:07:43	M-02	28.49		Cancelar Ver
22	44	2023-02-23 21:20:26	M-02	57.19		Cancelar Ver
23	44	2023-02-23 21:23:54	M-02	46.71		Cancelar Ver
24	44	2023-02-23 21:25:58	M-02	27.97		Cancelar Ver
25	44	2023-02-23 21:28:57	M-02	27.73		Cancelar Ver

4.2.7. Prueba 7

Al día siguiente, después de haber ingresado los pedidos del día, y al revisar la predicción de ventas nuevamente en el panel de administración, se tuvo una información totalmente diferente a la de la figura al inicio de esta sección correspondiente a la predicción de ventas inicial. La figura a continuación muestra la predicción de venta para el viernes 24 de febrero de 2023.

Figura 113
Predicción de venta del día siguiente al día de prueba inicial



Producto	Cantidad
Burrito Pollo	1
Matambre	1
Mojito	1
Agua sin gas	1
Pilsener 330ml	1
Pilsener 750ml	2
Club 330ml	2
Club 750ml	2
Chicken Strips	2
Bife de chorizo	1
Té helado	1
Nachos Simples	2

De esta forma se logra demostrar el funcionamiento total del sistema demostrando la toma de pedidos inteligente, la comunicación inalámbrica utilizando un enrutador que permite manejar tecnología basada en el estándar 802.11ax, predicciones de venta utilizando algoritmos de machine learning y la implementación de nodos que permitan solicitar asistencia a los clientes sin tener que levantarse de la mesa o buscar al personal del establecimiento gastronómico.

Finalmente se tiene el análisis del tiempo de respuesta del sistema en donde se logra analizar las tramas que se verán en las figuras siguientes para observar los tiempos que transcurren entre una trama y otra involucradas en la comunicación de nodos con el servidor.

La figura siguiente tiene 3 tramas, la primera es la solicitud al servidor de envío de asistencia a la mesa, la segunda es la confirmación de recepción de petición y la tercera es la trama

correspondiente a la respuesta con la información para notificarle al nodo que su petición fue recibida y se enviará la asistencia a mesa.

Figura 114
Tramas de análisis para tiempos de respuesta

38	192.168.1.20	192.168.1.123	HTTP	GET /mesa HTTP/1.1
39	192.168.1.123	192.168.1.20	TCP	80 → 61426 [ACK] Seq=1 Ack=435 Win=63806 Len=0
40	192.168.1.123	192.168.1.20	HTTP/JSON	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)

El tiempo en que llegó la solicitud oficial, después de haber establecido el proceso de sincronización entre el nodo y el servidor, es a las 11:23:54.21224774 segundos al servidor como se observa en la figura a continuación señalado en rosado.

Figura 115
Tiempo de llegada de petición de nodo a servidor

38	192.168.1.20	192.168.1.123	HTTP	GET /mesa HTTP/1.1
Frame 38: 488 bytes on wire (3904 bits), 488 bytes captured (3904 bits) on interface eth0 Section number: 1 > Interface id: 0 (eth0) Encapsulation type: Ethernet (1) Arrival Time: Feb 27, 2023 11:23:54.212264774 -05				

La trama de respuesta oficial fue enviada desde el servidor hacia el nodo a las 11:23:54:313625972 segundos como se ve en la figura siguiente en el recuadro verde. Si se resta estos dos tiempos, se obtiene que ha pasado en total un tiempo de 0.101280052 segundos como se ve señalado en rosado en la figura siguiente y este tiempo es bastante rápido para garantizar una buena comunicación entre los nodos y clientes con el servidor.

Figura 116
Tiempo de trama de respuesta a cliente

40	192.168.1.123	192.168.1.20	HTTP/JSON	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
Frame 40: 477 bytes on wire (3816 bits), 477 bytes captured (3816 bits) on interface eth0, Section number: 1 > Interface id: 0 (eth0) Encapsulation type: Ethernet (1) Arrival Time: Feb 27, 2023 11:23:54.313625972 -05 [Time shift for this packet: 0.000000000 seconds] Epoch Time: 1677515034.313625972 seconds [Time delta from previous captured frame: 0.101280052 seconds] [Time delta from previous displayed frame: 0.101280052 seconds]				

A continuación se muestra una tabla de las pruebas realizadas en ambiente real con su descripción y si se cumplió o no la prueba al igual que la tabla 18.

Tabla 19
Pruebas en ambiente real

Prueba	Descripción	Cumple	No Cumple
1	Análisis y ubicación de enrutador en posición adecuada	X	
2	Verificar la predicción de venta	X	
3	Realización de pedido con cliente en la mesa más lejana del local	X	
4	Comprobación del funcionamiento del botón de asistencia	X	
5	Comprobación de recepción de pedido	X	
6	Repetir todo el proceso con otro cliente	X	
7	Comprobación de predicción para el siguiente día	X	

Discusión

El presente proyecto logra cumplir con los objetivos de la sección de antecedentes y además deja abiertas las posibilidades para mejorarlo e implementarlo acorde a las necesidades de todos los lectores de este documento.

Dos de los trabajos relacionados a este es el de Ehsan Rahman y Shweta Shashikant Tanpure (en la sección de anexos están los documentos relacionados) se encargaron de realizar sistemas centralizados para poder tomar pedidos utilizando tabletas o PDAS las cuales eran manipuladas por el personal del establecimiento para enviar estas órdenes hacia el administrador

y también hacia cocina. El trabajo desarrollado en este documento permite que los clientes realicen su propio pedido y el personal del establecimiento estará disponible para asistir a los clientes que requieran ayuda para realizar su pedido o solventar alguna necesidad adicional de los clientes que ya hayan realizado su pedido y estén a la espera de la entrega de este. Por otra parte, el proyecto de Rahman no cuenta con una metodología aparte de la tradicional para solicitar asistencia, el sistema desarrollado en este documento cuenta con un nodo ubicado en las mesas para solicitar la presencia de un miembro del personal de trabajo del establecimiento. Al igual que el sistema desarrollado por Rahman, este proyecto utiliza una red inalámbrica basada en el estándar 802.11 para la comunicación entre dispositivos clientes y el servidor. El trabajo de Shweta tenía una característica adicional la cual era la comunicación en tiempo real la cual se logró implementar en este documento utilizando websockets.

Las otras aplicaciones mencionadas en el párrafo anterior utilizaban un sistema como aplicación para poder realizar la toma de pedidos. Varsha Chavan, en 2015, desarrollo una aplicación web y así brindar el acceso al servidor utilizando el navegador web. La limitación de este sistema era que se encontraba desarrollado para ser compatible con el sistema operativo Android. El proyecto de titulación desarrollado por el autor de este documento utiliza una aplicación web a la cual se puede acceder desde cualquier sistema operativo de dispositivos móviles así como ordenadores portátiles.

El proyecto planteado en este documento deja abiertas muchas puertas de implementación y mejora. Estas se mencionan de manera resumida en la siguiente lista.

- La primera opción es realizar el complemento del sistema con el sistema de facturación para poder realizar una facturación directa para los clientes de forma

electrónica y cumplir uno de los requisitos de los sistemas de venta con los requisitos del Servicio de Rentas Internas (SRI).

- Otro complemento que se puede implementar en este proyecto es el método de pago por tarjeta de crédito al momento de finalizar el pedido y que este envíe notificaciones a los usuarios administradores al momento de recibir un pago realizado.
- En cuanto a herramientas de administración queda abierta la posibilidad de realizar un diagrama que permita observar los niveles de venta según un periodo de tiempo específico y así poder analizar las ventas del local.
- Una opción extra para un macroproyecto sería la implementación de usuarios para un patio de comidas en donde cada restaurante tenga un usuario de administrador, pero dependiendo del usuario se selecciona el algoritmo de aprendizaje de máquina para generar una predicción de venta por locales varios. Además, los nodos al tener un código de verificación, se puede agregar un capo adicional que permita vincular el código del pedido con el restaurante al cual se esté realizando el pedido.

4.3. Análisis de costos

Esta sección tiene los costos referenciales para la implementación el proyecto. De todos los ítems que se listan en la siguiente tabla, el hardware para el servidor ser opcional en caso de que se tenga otro equipo con sistema operativo Windows server o con Linux y que tenga características similares o superiores al servidor que se utilizó para desarrollar este proyecto.

Tabla 20
Desglose de costos de implementación del proyecto

Componente	Cantidad	Precio Unitario (\$)	Total (\$)
Modulo ESP32	2	12	24
Raspberry Pi4	1	319	319

Impresión 3D	2	10	20
Botones para nodo	6	0.5	3
Display LCD	2	5	10
Placa de cobre	1	3	3
Resistencia 330 ohms	6	0.10	0.60
Cables conectores	10	0.1	1
Router 802.11ax	1	100	100
Total de costo de implementación(\$)			480.6

El origen de los costos de referencia para el desarrollo del proyecto se encuentra en el anexo 4 al final de este documento. Los precios pueden variar según la fecha de apertura de este documento.

Conclusiones y Recomendaciones

Este apartado contiene las conclusiones obtenidas a través de la experiencia del desarrollo de este proyecto y también las recomendaciones para aquellos lectores que deseen utilizar o implementar este proyecto para solventar un problema o una necesidad similar.

Conclusiones

El sistema desarrollado en este documento brinda herramientas para la toma de pedidos de forma inteligente a través de una aplicación web y predicción de venta utilizando una red neuronal. Su funcionamiento es de forma independiente de una conexión a internet, además se complementa con el despliegue de una red inalámbrica utilizando la última tecnología disponible en el mercado para esta, es decir, el estándar 802.11ax también conocido como Wi-Fi 6. A esta red se encuentran conectados también los nodos ubicados en cada mesa los cuales establecen una comunicación con el servidor en tiempo real para el envío de notificaciones de requerimientos de asistencia, entrega de pedido y generación de códigos de verificación.

La recopilación de información sobre los elementos y tecnologías que se van a utilizar para el desarrollo de un proyecto permite conocer más sobre diferentes caminos y herramientas que pueden facilitar el desarrollo del proyecto e incluso se llega a evitar cometer errores por falta de compatibilidad de tecnologías a causa del desconocimiento de la parte teórica. Para este caso, la implementación de un framework facilitó el desarrollo del sistema permitiendo utilizar la plantilla de un sistema base que ya tenía predeterminadas ciertas funciones que deben ser modificadas acorde a cada necesidad de cada aplicación

La metodología en cascada permitió verificar cada acción que se iba ejecutando en el desarrollo del proyecto para garantizar el cumplimiento de objetivos, requerimientos y finalmente

el funcionamiento del sistema acorde a las necesidades identificadas en las encuestas y en trabajos relacionados.

El diseño por etapas permitió observar el comportamiento del sistema en cada paso y facilitó la identificación de errores o también conocidos como “bugs” para corregirlos pronto y así continuar con el desarrollo del proyecto permitiendo evitar los errores cometidos en otros escenarios.

Para la red inalámbrica se utilizó como dispositivo principal un enrutador que trabaje con tecnología 802.11ax, este estándar permite manejar múltiples conexiones y transmisiones de datos simultáneas. Además, la red al utilizar frecuencias de 2.4 GHz y 5GHz para el despliegue de las redes inalámbricas, permite la compatibilidad de dispositivos actuales como los antiguos y gracias a esto los nodos ubicados en las mesas pueden enlazarse a la red de forma exitosa.

La fiabilidad del sistema para la predicción se encuentra en un 87%, esto es a causa de que el registro de la base de datos no es tan amplio dado que se requiere al menos el registro de ventas de un año para poder obtener una predicción más amplia.

El tiempo de respuesta entre un usuario, sea nodo o cliente, y el servidor es menor a 0,2 segundos de respuesta según el análisis de tramas realizado en las capturas lo cual se traduce a un tiempo de respuesta rápido y muy bueno para mantener una comunicación con los clientes que deseen acceder al servidor.

La aplicación del algoritmo de aprendizaje de máquina queda habilitado como complemento al sistema y no requiere de paga como lo hacen otras aplicaciones que tienen esta característica. La aplicación del algoritmo de aprendizaje de máquina se debe realizar acorde a las características de cada establecimiento y esto se puede realizar guiándose en las figuras dentro de este documento para luego implementarlas en otro escenario de venta.

Recomendaciones

Es importante el buscar información actualizada para la solución de un problema. Algunas soluciones a los problemas que aparecieron en el transcurso del desarrollo de este proyecto se encontraron en los blogs de las comunidades de desarrolladores dado que la información que se obtuvo en algunas situaciones era desactualizada y para solucionar un problema se debía cambiar ciertas funciones e importar otras librerías nuevas o cambiar ciertas versiones de algunos paquetes requeridos para el funcionamiento del proyecto.

Recomiendo analizar bien la metodología que se va a utilizar para la ejecución de un proyecto dado que esto trazará el camino a seguir para el cumplimiento de los requerimientos y objetivos. En algunos casos ciertas metodologías son muy flexibles y provocan que la identificación y corrección de errores se compliquen. El uso de diagramas de flujo o bloques son herramientas esenciales cuando se habla de desarrollo de proyectos que permiten a los usuarios entender de mejor manera el sistema y su funcionamiento.

Una vez que se tenga la idea principal de un proyecto, es importante investigar sobre todas las opciones que tengan las características para solucionar el problema y realizar una comparativa entre todas ellas para definir la mejor opción. Una vez seleccionada una opción, para el siguiente problema es importante verificar que sea posible acoplarla con las demás seleccionadas previamente.

Para mejorar el entrenamiento de la red neuronal se recomienda utilizar una cantidad de registros de venta de un año para poder alimentar la base de datos y de esta forma permitir que la red genere una predicción más exacta.

Para la implementación del sistema en un ambiente real, recomiendo empezar por la cobertura de la red para garantizar la conexión de los nodos y el acceso de los clientes al servidor donde se encuentra alojada la aplicación web. En caso de que la cobertura con un enrutador no sea suficiente, se debe realizar el análisis respectivo del ambiente de trabajo para el sistema y así distribuir los puntos de acceso dentro del local.

Bibliografía

- A, M. (2021, diciembre 13). *Machine Learning: definición, funcionamiento, usos*.
<https://datascientest.com/es/machine-learning-definicion-funcionamiento-usos>
- Ajitgupta Prerna. (2021, febrero 16). *I2C Communication Protocol - GeeksforGeeks*.
<https://www.geeksforgeeks.org/i2c-communication-protocol/>
- Arduino. (2018, febrero 5). *What is Arduino? | Arduino*. <https://www.arduino.cc/en/Guide/Introduction>
- ArubaNetworks. (2019). *WHAT IS 802.11AX (WI-FI 6)?*
https://www.arubanetworks.com/assets/so/SO_80211ax.pdf
- Ashraf Said. (2022, noviembre). *Arduino Communication with I2C Protocol | Udemy*.
https://www.udemy.com/course/arduino-i2c-itwoc-communication-with-i2c-protocol/?utm_source=adwords&utm_medium=udemyads&utm_campaign=DSA_Catchall_la.EN_cc.ROW&utm_content=deal4584&utm_term=._ag_88010211481_.ad_535397282064_.kw_.de_c_.dm_.pl_.ti_dsa-46281664096_.li_9076633_.pd_.&matchtype=&gclid=CjwKCAiA2fmdBhBpEiwA4CcHzf73udBV80uJ6nX2XCajLwHEGm9iV3087N5B0hwsTOWwogWx_uZoEhoCxqEQAvD_BwE
- BBVA. (2019, noviembre 8). *Te contamos qué es el “machine learning” y cómo funciona*.
<https://www.bbva.com/es/machine-learning-que-es-y-como-funciona/>
- BirthLH. (2018). 3.3.- *Red en estrella. | ICTV09.- Instalaciones de redes digitales de datos en viviendas y edificios*.
https://ikastaroak.ulhi.net/edu/es/IEA/ICTV/ICTV09/es_IEA_ICTV09_Contenidos/website_33_red_en_estrella.html
- Cablefree. (2017, mayo 18). *The History of WiFi: 1971 to Today - CableFree*.
<https://www.cablefree.net/wireless-technology/history-of-wifi-technology/>

- CodeIgniter. (2023). *Build Your First Application — CodeIgniter 4.3.1 documentation*.
https://codeigniter.com/user_guide/tutorial/index.html
- Computer Hope. (2018, enero 24). *What is Bus Topology?*
<https://www.computerhope.com/jargon/b/bustopol.htm>
- Corvo, H. (2019, octubre 9). *Topología de malla: características, ventajas, desventajas*.
<https://www.lifeder.com/topologia-de-malla/>
- Electrostore. (2021). *MÓDULO ESP32 ESP-32 WIFI BLUETOOTH – Grupo Electrostore*.
<https://grupoelectrostore.com/shop/placas-para-programacion/esp/modulo-esp32-esp-32-wifi-bluetooth/>
- Extreme. (2021, marzo 1). *A History of Wireless Standards: Wi-Fi Back to Basics*.
<https://www.extremenetworks.com/extreme-networks-blog/a-history-of-wireless-standards-wi-fi-back-to-basics/>
- ExtremeNetworks. (2020). *What is 802.11ax (Wi-Fi 6)? New Wi-Fi Standard | Extreme Networks*.
<https://www.extremenetworks.com/wifi6/what-is-80211ax/>
- GONCALVES, R. (2016, octubre 11). *Breve historia del ensamblaje automotriz - Motor y Racing*.
<https://www.motoryracing.com/pruebas/noticias/breve-historia-del-ensamblaje-automotriz/>
- Grapheverywhere. (2021). *Algoritmos de Machine Learning - Conoce cuales son sus potencialidades*.
<https://www.grapheverywhere.com/algoritmos-de-machine-learning/>
- Hans-Dieter, W. (2021). *The first wireless LAN /WLAN - Teldat Blog - Connecting the World*.
<https://www.teldat.com/blog/the-first-wireless-lan-wlan/>
- IBM. (2021a). *¿Qué es Machine Learning? - México | IBM*. <https://www.ibm.com/mx-es/analytics/machine-learning>

- IBM. (2021b, junio 10). *Regresión Logística - Documentación de IBM*.
<https://www.ibm.com/docs/es/spss-statistics/SaaS?topic=regression-logistic>
- Itoh, C. (2013). *A History of Wireless Communication and Yokogawa's Approach* | Yokogawa Electric Corporation. <https://www.yokogawa.com/library/resources/yokogawa-technical-reports/a-history-of-wireless-communication-and-yokogawas-approach/>
- Luque, A., & Pavlov Raul. (2020, junio). *Vista de Impacto de la tecnología en la sociedad: el caso de Ecuador* | Revista Ciencias Pedagógicas e Innovación.
<https://incyt.upse.edu.ec/pedagogia/revistas/index.php/rcpi/article/view/299/422>
- MDN Contributors. (2021, febrero 11). *Que es un servidor WEB? - Aprende sobre desarrollo web* | MDN. https://developer.mozilla.org/es/docs/Learn/Common_questions/What_is_a_web_server
- Mejía, J. (2015, mayo 3). *TIPOS DE TOPOLOGIAS PACKET TRACER*.
<http://josedejesusmejiagonzalez.blogspot.com/2015/05/red-en-anillo-red-con-topologia-de.html>
- Netgear. (2021). *NIGHTHAWK R6700AX DATASHEET*.
<https://www.netgear.com/images/datasheet/networking/wifirouter/R6700AX.pdf>
- NIBUSINESS. (2020). *Business benefits of artificial intelligence* | nibusinessinfo.co.uk.
<https://www.nibusinessinfo.co.uk/content/business-benefits-artificial-intelligence>
- PHP Group. (2022). *PHP: What is PHP? - Manual*. <https://www.php.net/manual/en/intro-what-is.php>
- Pratt, M. (2021, junio 18). *7 Key Benefits of AI for Business*.
<https://www.techtarget.com/searchenterpriseai/feature/6-key-benefits-of-AI-for-business>
- Raspberry PI foundation. (2022). *What is a Raspberry Pi?* <https://www.raspberrypi.org/help/what-is-a-raspberry-pi/>
- Raspbian. (2022). *RaspbianAbout - Raspbian*. <https://www.raspbian.org/RaspbianAbout>

- RaspberryPi. (2022). *Raspberry Pi 4 Model B specifications – Raspberry Pi*.
<https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/>
- Rehman, J. (2018, agosto 1). *Examples and types of mini computers - IT Release*.
<https://www.itrelease.com/2018/08/examples-and-types-of-mini-computers/>
- SMELPRO. (2020, junio 25). *¿Qué es un microcontrolador? – Smelpro*. <https://smelpro.com/blog/ques-un-microcontrolador/>
- Steve. (2019). *Home Wireless Networking (Wi-Fi) for Beginners*.
<https://stevesmarthomeguide.com/home-wireless-networking/>
- Techopedia. (2017, junio 15). *What is a Minicomputer? - Definition from Techopedia*.
<https://www.techopedia.com/definition/4615/minicomputer>
- Techtarget. (2005, septiembre). *What is free software? - Definition from WhatIs.com*.
<https://www.techtarget.com/whatis/definition/free-software>
- Toro, J. (2015, noviembre 4). *Topología de árbol*. <https://es.slideshare.net/Jessica10-4/topologa-de-rbol-54729968>
- Wi-fi Alliance. (2019, junio 4). *Wi-Fi History*. <https://www.wi-fi.org/news-events/newsroom/wi-fi-alliance-celebrates-20-years-of-wi-fi>

Anexos

1. ANEXO 1: Formato de cuestionarios con resultados

CUESTIONARIO DIRIJIDO A PROPIETARIOS DE RESTAURANTES

Objetivo:

El propósito de esta encuesta es obtener datos que corroboren la relevancia del desarrollo del proyecto de grado a realizar e identificar otro tipo de variables dignas de análisis para la ejecución de una posible solución a dichos conflictos que se encuentren en los establecimientos gastronómicos encuestados y también para obtener requisitos para el desarrollo del sistema.

La participación en este estudio es completamente voluntaria y las respuestas brindadas del estudio serán utilizadas exclusivamente para el presente proyecto y en los repositorios de la Universidad Técnicas del Norte.

Por su tiempo y apoyo le estamos agradecidos.

Datos de entrevistado:

Ciudad: Ibarra

Nombre y Apellido:

Local Gastronómico al que pertenece:

Cargo que Ocupa en el establecimiento:

Fecha de encuesta:

Datos de encuestador:

Nombres y Apellidos: Paul Alejandro Zurita Zambrano

Cédula de identidad: 1003197306

Cargo: Estudiante de la Universidad Técnica del Norte de la carrera de Ingeniería en Electrónica y Redes de Comunicación

Preguntas:

1. Seleccione el tiempo que permanece su local abierto al público.
 - 4-5 horas
 - 6-7 horas
 - 8 horas o más
2. En su establecimiento, ¿Cuenta con un servicio que le permita tener acceso a internet a usted y a sus clientes? Si su respuesta es negativa pase a la pregunta 5
 - Si
 - No
3. Si su respuesta anterior fue afirmativa, ¿Ha sufrido problemas de conexión mediante el uso de este servicio en horas de trabajo?
 - Si
 - No
4. Si su respuesta fue afirmativa anteriormente, seleccione el periodo de interrupciones del servicio de internet que más se aproxime a su situación:
 - Una vez a la semana
 - Más de una vez a la semana
 - Una vez cada quince días
 - Una vez al mes
5. Seleccione la distancia de la mesa más cercana al punto de acceso otorgado por la empresa que le brinda el acceso a internet o counter de venta del establecimiento.

- 2-4 metros
 - 5-7 metros
 - 7-9 metros
6. Seleccione la distancia de la mesa más lejana al punto de acceso otorgado por la empresa que le brinda el acceso a internet o counter de venta del establecimiento.
- 5-9 metros
 - 10-20 metros
 - En un segundo piso del establecimiento
7. ¿Han experimentado sus clientes alguno de los siguientes problemas o dificultades al requerir de asistencia por parte de su personal?
- El personal de servicio se encuentra ocupado.
 - El cliente se encuentra en una mesa con poca visibilidad del establecimiento.
 - El cliente hace una petición y el personal la olvida sin mala intención.
8. ¿Cuenta con un sistema de toma de pedidos digital? Si su respuesta es negativa pase a la pregunta 10.
- Si
 - No
9. ¿Ha experimentado problemas de conectividad al sistema de gestión de pedidos de sus clientes?
- Si
 - No

10. En cuanto al tiempo de servicio, ¿Ha experimentado problemas con sus clientes por falta de información sobre los pedidos y el estado de preparación de los platillos solicitados por sus comensales?
- Si
 - No
11. ¿Su establecimiento cuenta con un mecanismo de predicción en cuanto a pronósticos de venta o número de clientes aproximados para un día específico?
- Si
 - No
12. ¿Cree usted que un método de información sobre el estado de pedidos a través de una red de dispositivos ubicados en cada mesa del establecimiento podría ayudar a disminuir la de duda sobre el tiempo de espera restante para la entrega del pedido?
- Si
 - No
13. ¿Estaría usted interesado en un sistema de toma de pedidos inteligente que funcione de manera independiente, es decir, que no dependa de una conexión a internet, de costo accesible y brinde información como predicciones de venta? Si su respuesta es negativa pase a la pregunta 16.
- Si
 - No
14. Este sistema planea integrar una interfaz para los clientes. Seleccione cuál sería la característica principal para poder desarrollar esta interfaz de usuario:

- Básica: Sin tanto detalle, con información simple y sin muchas opciones para los clientes.
- Media: Con información más detallada y que permita a los clientes ingresar observaciones en sus pedidos.
- Compleja: Debe ofrecer información detallada de los productos y al cliente debe permitirle armar su plato y seleccionar los ingredientes que desee en su comida.

15. Dentro del área administrativa se tiene otra interfaz. Seleccione cual cree que sea la mejor opción en cuanto a características principales del sistema.

- Básica: Toda la información debe estar resumida sin tanto detalle y sin opciones para el administrador.
- Mediana: Debe contar con opciones extras para diferentes operaciones y mostrar información sobre las mesas.
- Compleja: Ventanas varias con botones para obtener información y manipular platos o bebidas de los pedidos.

16. ¿Considera usted útil una implementación de dispositivos en las mesas para evitar que los clientes busquen al personal de servicio para solicitar ayuda? Si su respuesta es negativa la encuesta habría culminado.

- Si
- No

17. ¿Preferiría usted que estos dispositivos usen baterías o una alimentación constante con una toma de corriente?

- Baterías
- Toma de corriente

18. Seleccione las características que cree usted que debería tener estos nodos:

- Compactos
- Estéticos y con tipografía legible.
- Grandes y robustos.
- Fácil de manipular
- Resistente a líquidos

Realizado por: Paul Alejandro Zurita Z.

Revisado por: MsC. Fabián G. Cuzme R.

RESULTADOS DE CUESTIONARIOS DIRIJIDOS A PROPIETARIOS

Estas encuestas fueron realizadas a los propietarios de dos establecimientos gastronómicos más sobresalientes de la ciudad de Ibarra para la recolección de información que permitan definir los requerimientos del sistema.

Anexo 1

Propietarios de establecimientos gastronómicos encuestados

Datos de entrevistado:

Nombre y Apellido, Local gastronómico, Cargo que Ocupa en el establecimiento gastronómico, Fecha de encuesta

2 respuestas

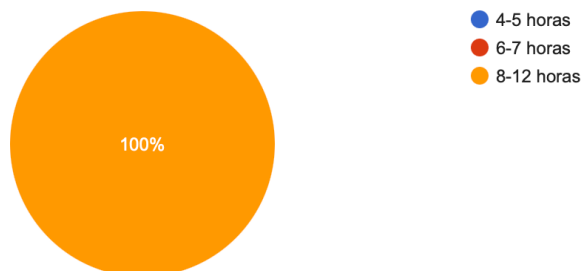
Israel Muñoz - Ilapa Brewing - Gerente - 13 Enero 2023

Francisco Amores, La Catrina tacos & tequilas

Anexo 2
Cuestionario Propietarios Pregunta 1

Seleccione el tiempo que permanece su local abierto al público.

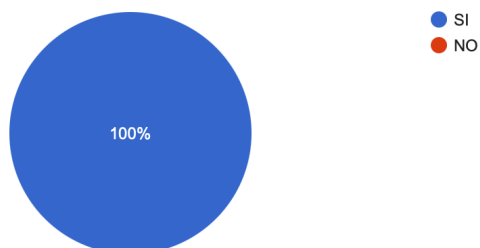
2 respuestas



Anexo 3
Cuestionario Propietarios Pregunta 2

En su establecimiento, ¿Cuenta con un servicio que le permita tener acceso a internet a usted y a sus clientes? Si su respuesta es negativa pase a la pregunta 5

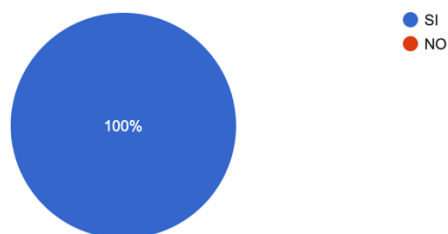
2 respuestas



Anexo 4
Cuestionario Propietarios Pregunta 3

Si su respuesta anterior fue afirmativa, ¿Ha sufrido problemas de conexión mediante el uso de este servicio en horas de trabajo?

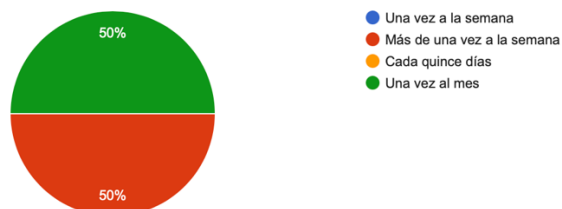
2 respuestas



Anexo 5
Cuestionario Propietarios Pregunta 4

Si su respuesta fue afirmativa anteriormente, seleccione el periodo de interrupciones del servicio de internet que más se aproxime a su situación:

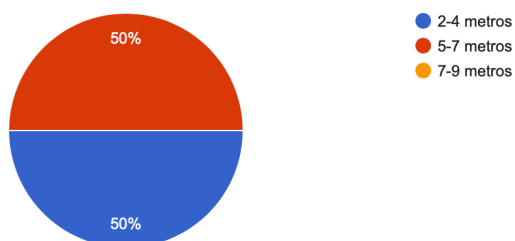
2 respuestas



Anexo 6
Cuestionario Propietarios Pregunta 5

Seleccione la distancia de la mesa más cercana al punto de acceso otorgado por la empresa que le brinda el acceso a internet o counter de venta del establecimiento.

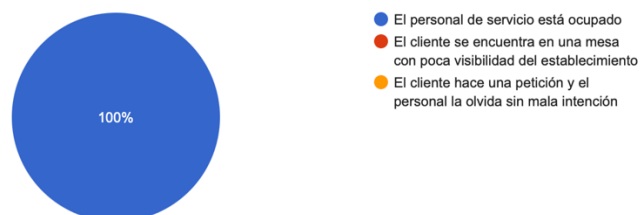
2 respuestas



Anexo 7
Cuestionario Propietarios Pregunta 6

¿Han experimentado sus clientes alguno de los siguientes problemas o dificultades al requerir de asistencia por parte de su personal?

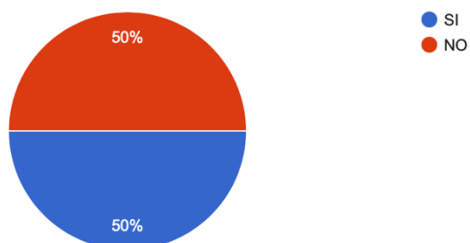
2 respuestas



Anexo 8
Cuestionario Propietarios Pregunta 7

¿Cuenta con un sistema de toma de pedidos digital? Si su respuesta es negativa pase a la pregunta 10.

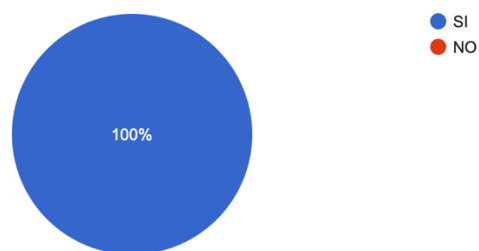
2 respuestas



Anexo 9
Cuestionario Propietarios Pregunta 8

¿Ha experimentado problemas de conectividad al sistema de gestión de pedidos de sus clientes?

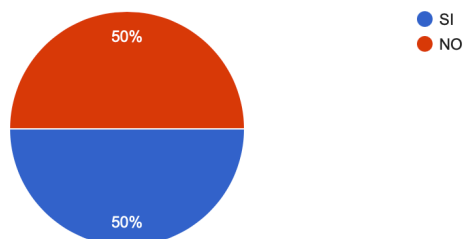
1 respuesta



Anexo 10
Cuestionario Propietarios Pregunta 9

En cuanto al tiempo de servicio, ¿Ha experimentado problemas con sus clientes por falta de información sobre los pedidos y el estado de preparación de los platillos solicitados por sus comensales?

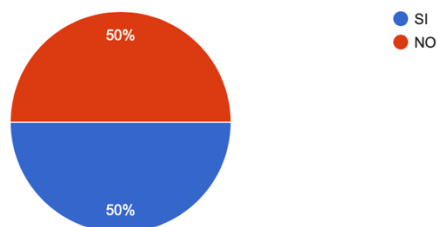
2 respuestas



Anexo 11
Cuestionario Propietarios Pregunta 10

¿Su establecimiento cuenta con un mecanismo de predicción en cuanto a pronósticos de venta o número de clientes aproximados para un día específico?

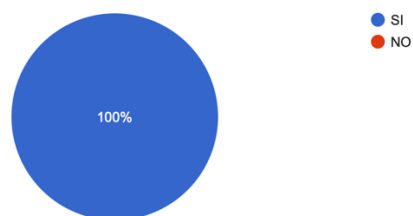
2 respuestas



Anexo 12
Cuestionario Propietarios Pregunta 11

¿Cree usted que un método de información sobre el estado de pedidos a través de una red de dispositivos ubicados en cada mesa del establecimiento podría ayudar a disminuir la de duda sobre el tiempo de espera restante para la entrega del pedido?

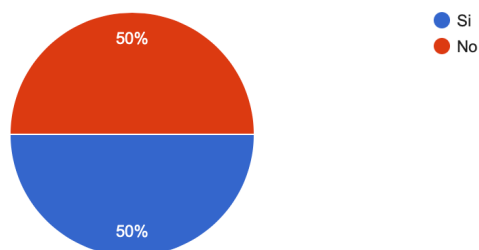
2 respuestas



Anexo 13
Cuestionario Propietarios Pregunta 12

¿Estaría usted interesado en un sistema de toma de pedidos inteligente que funcione de manera independiente, es decir, que no dependa de una conexión a internet, cuente con una red de nodos que permitan una comunicación con el cliente, que sea de un costo accesible y brinde información como predicciones de venta?

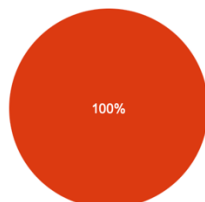
2 respuestas



Anexo 14
Cuestionario Propietarios Pregunta 13

Este sistema planea integrar una interfaz para los clientes. Seleccione cuál sería la característica principal para poder desarrollar esta interfaz de usuario

2 respuestas

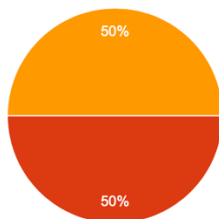


- Básica: Sin tanto detalle, con información simple y sin muchas opciones para los clientes.
- Media: Con información más detallada y que permita a los clientes ingresar observaciones en sus pedidos
- Compleja: Debe ofrecer información detallada de los productos y al cliente debe permitirle armar su plato y seleccionar los ingredientes que desee en su comida.

Anexo 15
Cuestionario Propietarios Pregunta 14

Dentro del área administrativa se tiene otra interfaz. Seleccione cual cree que sea la mejor opción en cuanto a características principales del sistema

2 respuestas

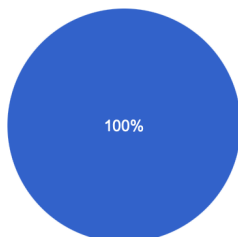


- Básica: Toda la información debe estar resumida sin tanto detalle y sin opciones para el administrador.
- Mediana: Debe contar con opciones extras para diferentes operaciones y mostrar información sobre las mesas.
- Compleja: Ventanas varias con botones para obtener información y manipular platos o bebidas de los pedidos.

Anexo 16
Cuestionario Propietarios Pregunta 15

¿Considera usted útil una implementación de dispositivos en las mesas para evitar que los clientes busquen al personal de servicio para solicitar ayuda? Si su respuesta es negativa la encuesta habría culminado.

2 respuestas

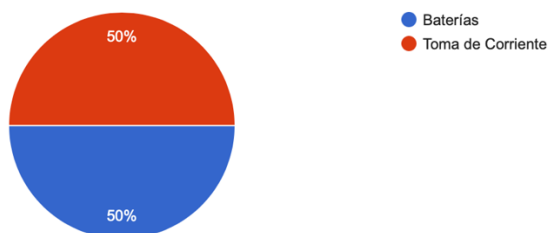


- Si
- No

Anexo 17
Cuestionario Propietarios Pregunta 16

¿Preferiría usted que estos dispositivos usen baterías o una alimentación constante con una toma de corriente?

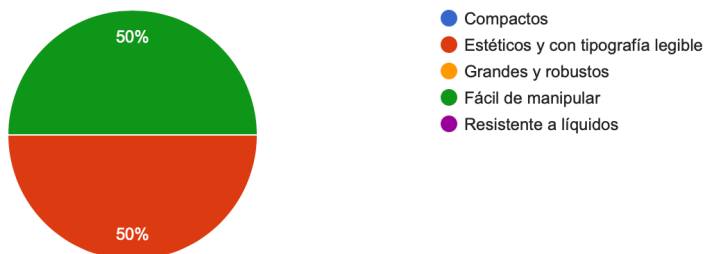
2 respuestas



Anexo 18
Cuestionario Propietarios Pregunta 17

Seleccione la característica que cree usted que debería tener estos nodos:

2 respuestas



CUESTIONARIO DIRIJIDO A ASESORES GASTRONÓMICOS

Datos de entrevistado:

Ciudad: Ibarra

Nombre y Apellido:

Cargo que Ocupa:

Fecha de encuesta:

Datos de Entrevistador:

Nombres y Apellidos: Paul Alejandro Zurita Zambrano

Cédula de identidad: 1003197306

Cargo: Estudiante de la Universidad Técnica del Norte de la carrera de Ingeniería en Electrónica y Redes de Comunicación

Objetivo:

El propósito de esta encuesta es obtener datos que corroboren la relevancia del desarrollo del proyecto de grado a realizar e identificar los requisitos para el desarrollo del sistema en base a su opinión y experiencia para que este sea lo óptimo y esté acorde a la realidad de la provincia y del país.

Preguntas:

1. Cuál es el tiempo máximo que suele abrir un local gastronómico al público diariamente. Seleccione la más común:
 - 4-5 horas
 - 6-7 horas
 - 8-12 horas

2. ¿Estaría usted interesado en un sistema de toma de pedidos inteligente que funcione de manera independiente, es decir, que no dependa de una conexión a internet, cuente con una red de nodos que permitan una comunicación con el cliente, que sea de un costo accesible y brinde información como predicciones de venta?
 - Si
 - No

3. ¿Considera usted útil una implementación de dispositivos en las mesas para evitar que los clientes busquen al personal de servicio para solicitar ayuda? Si su respuesta es negativa la encuesta habría culminado.
 - Si

- No
4. ¿Preferiría usted que estos dispositivos usen baterías o una alimentación constante con una toma de corriente?
- Baterías
 - Toma de corriente
5. Seleccione las características que cree usted que debería tener estos nodos:
- Compactos
 - Estéticos y con tipografía legible.
 - Grandes y robustos.
 - Fácil de manipular
 - Resistente a líquidos
6. Este sistema planea integrar una interfaz para los clientes. Seleccione cuál sería la característica principal para poder desarrollar esta interfaz de usuario:
- Básica: Sin tanto detalle, con información simple y sin muchas opciones para los clientes.
 - Media: Con información más detallada y que permita a los clientes ingresar observaciones en sus pedidos.
 - Compleja: Debe ofrecer información detallada de los productos y al cliente debe permitirle armar su plato y seleccionar los ingredientes que desee en su comida.
7. Dentro del área administrativa se tiene otra interfaz. Seleccione cual cree que sea la mejor opción en cuanto a características principales del sistema.
- Básica: Toda la información debe estar resumida sin tanto detalle y sin opciones para el administrador.

- Mediana: Debe contar con opciones extras para diferentes operaciones y mostrar información sobre las mesas.
- Compleja: Ventanas varias con botones para obtener información y manipular platos o bebidas de los pedidos.

Realizado por: Paul Alejandro Zurita Z.

Revisado por: MsC. Fabián G. Cuzme R.

RESULTADOS DE CUESTIONARIOS DIRIJIDOS A ASESORES GASTRONÓMICOS

Los cuestionarios fueron dirigidos hacia dos asesores gastronómicos para poder recolectar información sobre los requerimientos del sistema.

Anexo 19
Asesores gastronómicos encuestados

Datos de entrevistado:

Nombre y Apellido, Cargo que Ocupa, Fecha de encuesta

2 respuestas

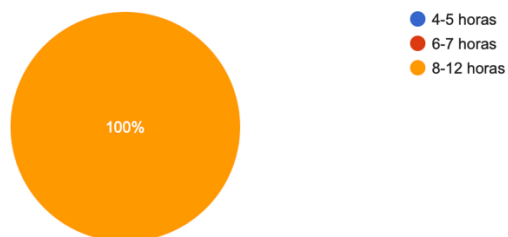
Nadia Garzon, administradora LCR Ibarra, 11 de enero de 2023

Pablo Proaño Chef 09/01/2023

Anexo 20
Encuesta Asesores Gastronómicos Pregunta 1

Cuál es el tiempo máximo que suele abrir un local gastronómico al público diariamente. Seleccione la más común.

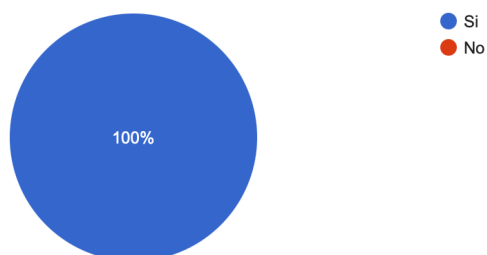
2 respuestas



Anexo 21
Encuesta Asesores Gastronómicos Pregunta 2

¿Estaría usted interesado en un sistema de toma de pedidos inteligente que funcione de manera independiente, es decir, que no dependa de una conexión a internet, cuente con una red de nodos que permitan una comunicación con el cliente, que sea de un costo accesible y brinde información como predicciones de venta?

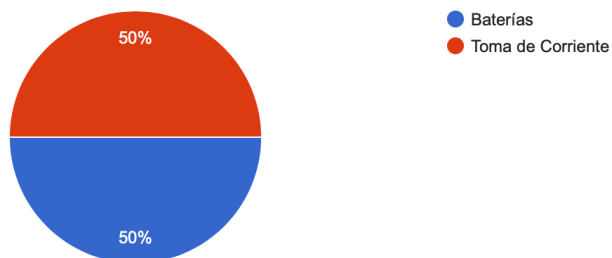
2 respuestas



Anexo 22
Encuesta Asesores Gastronómicos Pregunta 3

¿Preferiría usted que estos dispositivos usen baterías o una alimentación constante con una toma de corriente?

2 respuestas

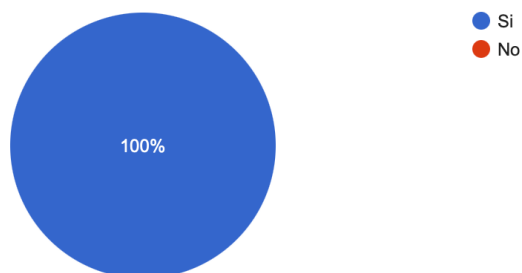


Anexo 23

Encuesta Asesores Gastronómicos Pregunta 4

¿Considera usted útil una implementación de dispositivos en las mesas para evitar que los clientes busquen al personal de servicio para solicitar ayuda? Si su respuesta es negativa la encuesta habría culminado.

2 respuestas

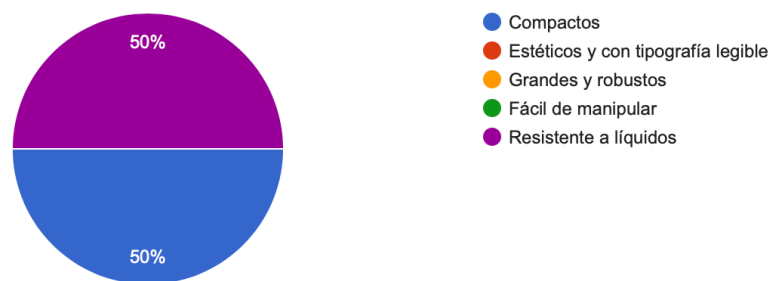


Anexo 24

Encuesta Asesores Gastronómicos Pregunta 5

Seleccione la característica que cree usted que debería tener estos nodos:

2 respuestas

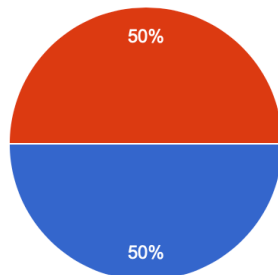


Anexo 25

Encuesta Asesores Gastronómicos Pregunta 6

Este sistema planea integrar una interfaz para los clientes. Seleccione cuál sería la característica principal para poder desarrollar esta interfaz de usuario

2 respuestas



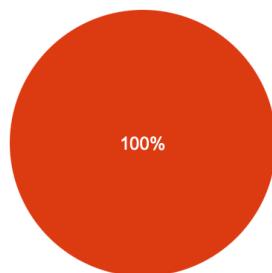
- Básica: Sin tanto detalle, con información simple y sin muchas opciones para los clientes.
- Media: Con información más detallada y que permita a los clientes ingresar observaciones en sus pedidos
- Compleja: Debe ofrecer información detallada de los productos y al cliente debe permitirle armar su plato y seleccionar los ingredientes que desee en su comida.

Anexo 26

Encuesta Asesores Gastronómicos Pregunta 7

Dentro del área administrativa se tiene otra interfaz. Seleccione cual cree que sea la mejor opción en cuanto a características principales del sistema

2 respuestas



- Básica: Toda la información debe estar resumida sin tanto detalle y sin opciones para el administrador.
- Mediana: Debe contar con opciones extras para diferentes operaciones y mostrar información sobre las mesas.
- Compleja: Ventanas varias con botones para obtener información y manipular platos o bebidas de los pedidos.

2. ANEXO 2: Revisión de trabajos relacionados

Objetivo:

- Revisar trabajos que tengan relación o semejanza con el proyecto de titulación y así obtener ciertos requisitos para el desarrollo del sistema que garanticen la funcionalidad de este.

2.1. Trabajo relacionado 1:

Ficha bibliográfica 1

Autor:	Ehsan Rahman	Editorial:	Hackster.io
Título:	“Restaurant uses a DIY Raspberry Pi System to automate order taking”		

Año:	2016	Ubicación:	Surrey, Inglaterra
------	------	------------	--------------------

Resumen del contenido:

- **Sistema de toma de pedido**

El sistema del cual se habla en este trabajo ocupa un minicomputador como servidor para la recepción de los platos solicitados por los clientes.

- **Envío de pedidos**

En lugar de usar comandas o hacer el pedido en un punto de venta, los meseros del establecimiento utilizan tabletas o teléfonos inteligentes que les permite tomar el pedido rápidamente y así mismo comprobar el estado de este utilizando una red inalámbrica.

- **Comunicación entre dispositivos**

El servidor se encuentra conectado al enrutador de manera directa utilizando un cable ethernet y la comunicación de todos los dispositivos de los meseros interactúan con el servidor utilizando una red inalámbrica basada en el estándar 802.11.

2.2. Trabajo relacionado 2:

Ficha bibliográfica 2

Autores:	Shweta Shashikant Tanpure, Priyanka R. Shidankar, Madhura M. Joshi	Editorial:	International Journal of Advanced Research in Computer Science and Software Engineering – Pune University
Título:	“Automated Food Ordering System with Real-Time Customer Feedback”		
Año:	2013	Ubicación:	India, Pune

Resumen del contenido:

- **Comparativa del sistema de toma de pedidos tradicional**

Este artículo menciona la comparativa de la toma de pedidos tradicionales basados en papel y lápiz junto con el requerimiento de personal específico para poder realizar esta dinámica y poder llevar la orden del cliente hacia la cocina del restaurante y cómo esto es un consumo de recursos económicos en una empresa concluyendo que es

mucho más efectivo utilizar los avances tecnológicos para la implementación de dispositivos que cumplan el rol de tomar el pedido a un cliente de manera automática.

- **Metodología de toma de pedido**

Dado el año de desarrollo de este proyecto se utilizaba la metodología de una persona que se acerque con un portátil hacia los clientes y tomar el pedido de estos. Una vez finalizada la toma del pedido se procedía a enviar la información a través de una red inalámbrica hacia un servidor el cual pasaba la información a la cocina del establecimiento y así el mesero quedaba libre para seguir tomando más órdenes en diferentes mesas o continuar con otras labores asignadas.

- **Dispositivos usados para la implementación de toma de pedidos**

En el año 2013 estaban aún de moda el uso de asistentes digitales personales (PDA) los cuales tenían ciertas limitantes como lo es el manejo de interfaces interactivas dado que la capacidad de procesamiento que tenían estas era básica y además el documento menciona que esto es un gasto adicional para el establecimiento al tener que adquirir estos dispositivos para la toma de pedidos. En la actualidad el uso de estos PDA viene a ser reemplazado por el teléfono móvil inteligente personal de cada cliente por lo que el gasto de implementación en dispositivos conocidos como tabletas se eliminaría.

2.3. Trabajo relacionado 3

Ficha bibliográfica 3

Autores: Varsha Chavan, Priya Jadhav, Snehal Korade y Priyanka Teli	Editorial: International Journal of Innovative Science, Engineering & Technology
---	--

Título:	“Implementing Customizable Online Food Ordering System Using Web Based Application”		
Año:	2015	Ubicación:	India, Pune

Resumen del contenido:

- **Uso de aplicación web para toma de pedidos**

Utilizando la tecnología inalámbrica basada en el estándar 802.11 De trabajo se configura un servidor web el cual funciona como un concentrador de todos los datos que se vayan a solicitar a la aplicación y estos datos serán enviados por los clientes al ingresar a dicho servidor para poder observar el menú y gestionar su pedido.

En este trabajo para la toma de pedidos se utiliza una aplicación desarrollada específicamente para el sistema operativo Android de tabletas las cuales estan conectadas al servidor según la configuración de este para poder guiar a los usuarios y poder realizar su pedido.

2.4. Trabajo relacionado 4:

Ficha bibliográfica 4

Autor:	Leong Wai Hong	Editorial:	University Tunku Abdul Rahman
Título:	: “Food Ordering System using Mobile Phone”		
Año:	2016	Ubicación:	India, Pune

Resumen del contenido:

- **Utiliza un computador para establecer el servidor al cual se puede acceder a través de una red inalámbrica**

La interfaz de usuario diseñada en este proyecto es lo más simple posible para que esta sea rápida y fácil de manejar para los usuarios y personal.

Mantiene algunos direccionamientos basados en usuario y contraseña para acceso al panel de administración o de cocina.

Firma autorización

Ing. Fabián Geovanny Cuzme Rodríguez

Director

Sr. Paúl Alejandro Zurita Zambrano

Estudiante

3. Anexo 3: Recopilación de requisitos de ANEXOS 1 Y 2

- Según revisión de trabajos relacionados

La revisión de los trabajos relacionados permitió al desarrollador de este proyecto observar ciertas características de otros sistemas que se pueden implementar en el desarrollo e implementación de este acorde a las necesidades de cada establecimiento gastronómico.

- Adquisición de datos

Otro método que permitió recolectar más información sobre las necesidades o preferencias reales de propietarios de restaurantes y también aprovechar la experiencia de asesores gastronómicos fueron las encuestas realizadas a diferentes personas mencionadas como stakeholders.

- Hardware y software

La selección de dispositivos y definición de requisitos para el uso de estos se la realizó utilizando una tabla comparativa acorde a la tecnología implementada dentro de los trabajos relacionados y también con los productos similares que se encuentra hoy en el mercado con características mejoradas y con un precio más accesible para los usuarios.

3.1. Definición de importancia de cada requisito

Las características que debe tener el sistema se califican en la tabla a continuación en donde se explica también la importancia que tiene cada requisito etiquetado según la importancia seleccionada para ver si es o no indispensable para el desarrollo del proyecto.

Tabla 21
Etiquetas de prioridad para requerimientos del proyecto

PRIORIDAD	DESCRIPCIÓN
Alta	Requerimiento crítico que debe ser considerado en el desarrollo del sistema.
Media	Requerimientos que pueden ser omitidos solo en condiciones de fuerza mayor.

Baja	Requerimientos que se pueden omitir, este requerimiento no genera impacto significativo en el sistema.
------	--

3.2. Requerimientos de stakeholders

La tabla número 4 detalla en resumen los requisitos de stakeholders y esta sección detalla su origen y la razón por la cual los datos obtenidos se convirtieron en un requerimiento para el desarrollo de este proyecto.

- El tiempo de funcionamiento que debe tener el sistema se obtuvo del resultado de las encuestas realizadas y acorde a estos datos, el periodo mínimo de uso que se le daría al sistema va entre 4 y más de 8 horas. Este rango de operación depende mucho de la dinámica de cada establecimiento gastronómico por lo que en resumen se obtiene que el tiempo de trabajo mínimo del sistema debe ser de 8 horas.
- En un restaurante es complicado el extender cableado específico para poder realizar una transmisión de datos y a su vez esto significa un gasto adicional que pocos usuarios están dispuestos a realizar. Lo más fácil en la actualidad es realizar dispositivos que envíen sus datos a un servidor o concentrador de forma inalámbrica, por lo que se define el segundo requerimiento de la tabla 4 correspondiente a una necesidad de comunicación inalámbrica para el envío de datos de los nodos clientes.
- Después de haber realizado la visita a los establecimientos gastronómicos que se han tomado como referencia para poder desarrollar las pruebas de implementación del sistema y también a otros establecimientos gastronómicos, se observó que las tomas de corriente cercanas a las mesas de los restaurantes son mínimas y no existe uno por mesa. De esta forma se obtiene que el funcionamiento de los nodos a

colocar en cada mesa debe ser alimentado por baterías que duren el mismo tiempo mínimo de trabajo del sistema.

- Basado en las encuestas la distancia mínima de la mesa más cercana al counter o al punto de acceso de los establecimientos gastronómicos entrevistados y visitados es de 5 metros y la distancia más lejana en algunos establecimientos es de 20 metros ya que estos no son tan grandes pero en el caso de El restaurante y lapa la distancia más grande a la última mesa es de 35 metros dado que es un restaurante al aire libre. Acorde al estándar 802.11 la distancia máxima de alcance de los puntos de acceso es de 100 metros por lo que teóricamente se podría decir que no existiría problema para poder establecer la comunicación entre los nodos ubicados en las mesas y el punto de acceso para poder establecer la conexión al servidor. En resumen, el rango de cobertura mínimo de la red que se ha decidido establecer es de 20 metros.
- La conexión del servidor dónde se va a alojar la aplicación del sistema debe ser lo más confiable posible y acorde al primer trabajo relacionado donde se utilizó un minicomputador Raspberry Pi, La conexión del servidor hacia el enrutador se realizó mediante cable para poder garantizar la fiabilidad de transferencia de datos en toda la red.
- Los usuarios que respondieron el formulario establecieron que es importante tener un método para solicitar ayuda o servicio personalizado desde la mesa para evitar que el cliente deba levantarse y buscar al personal del establecimiento gastronómico.

- Los asesores gastronómicos en sus encuestas respondieron que soportar el derramamiento de líquidos y tener un diseño estético son unas de las características importantes de los nodos que se van a ubicar en las mesas.
- Después de la revisión de los trabajos relacionados y el desarrollo de estos el manejo de una interfaz simple garantiza el buen funcionamiento del sistema y una buena experiencia del usuario. esto también lo corroboran las encuestas realizadas a los asesores gastronómicos y propietarios de restaurantes.

3.3. Requerimientos de sistema

Los requerimientos de sistema tienen la abreviatura de SRS. Estos se resumen en la tabla número 5 en donde se establecen otros requerimientos de interfaz, uso y físicos. A continuación se explica el origen de cada requerimiento.

- Basados en los trabajos relacionados, todos los sistemas similares cumplen con el tiempo de funcionamiento acorde a la duración de la jornada de trabajo por lo cual se obtiene otro requerimiento del tiempo de servicio acorde a una jornada laboral, o sea, 8 horas o más dependiendo del establecimiento gastronómico. Este requerimiento se va relacionado con el primero requerimiento de la tabla 4.
- Basado en las hojas de datos de cada dispositivo, se llega a la conclusión de que los voltajes a manejar para los nodos clientes deben ser entre 3.3 voltios y 5 voltios como máximo.
- Los trabajos relacionados tomaban información de los clientes y también generaba información y esta debía ser almacenada en algún lugar para tener acceso a dichos datos luego. Para dar solución a esto, se requiere que los datos se guarden en una base de datos dentro del servidor mismo y que esta funcione de manera local.

- El acceso al servidor por parte de los meseros en los trabajos relacionados utiliza una comunicación inalámbrica basada en el estándar 802.11. En este proyecto también se utiliza un enlace inalámbrico basado en el estándar 802.11xx para el acceso al servidor.
- El trabajo realizado por Leong Wai en 2016, la interfaz de usuario no debe ser lo más llamativa posible sino lo más simple y que cuente con directrices claras para el usuario que se encuentre navegando en el servidor.
- Tomando en cuenta las encuestas realizadas a los asesores gastronómicos se establece que los nodos deben tener un fácil acceso para cualquier mantenimiento preventivo de hardware y a su vez deben tener suficiente ventilación para evitar que el servidor colapse por temperatura o se sature.
- En cuanto al diseño se obtuvieron requisitos de las encuestas respondidas en donde se señalan las características básicas pero no esenciales para el diseño de los nodos ubicados en las mesas son: tener un diseño compacto, que soporten salpicaduras de líquidos y que en lo posible sean herméticos.

3.4. Requerimientos de arquitectura

Los requerimientos de arquitectura se obtienen basados más en la experiencia de los desarrolladores de la aplicación y de la revisión de otros trabajos relacionados. En la tabla número 6 se puede observar resumidos los requerimientos para poder continuar con el desarrollo del sistema planteado en este documento.

- Basado en el cuarto requerimiento de sistema, en el segundo requerimiento de stakeholders y trabajos relacionados se obtiene el primer requerimiento lógico de arquitectura de enviar los datos hacia el servidor utilizando un enlace inalámbrico dado que es la mejor opción para la implementación de este proyecto.
- En cuanto a los requerimientos de diseño, es importante mantener un bajo costo para la implementación de este proyecto dado que en la revisión de los trabajos relacionados se observa que la implementación de dichos proyectos no posee costos de elevados y es por eso que son llamativos para la implementación de un emprendimiento.
- Acorde a la experiencia del tutor de este trabajo y a la experiencia del desarrollador de este, Se concluye que los requerimientos mínimos de memoria RAM para el servidor en donde se alojará todos los datos del establecimiento gastronómico debe tener al menos 4 gigabytes de memoria RAM para poder realizar las distintas operaciones designadas al servidor sin saturar a este.
- Es importante contar con puertos adicionales para poder conectar periféricos como dispositivos de almacenamiento externos, teclado o un ratón para poder realizar ciertas operaciones dentro del servidor y asimismo se requiere de un puerto que brinde la posibilidad de conectar un monitor que permita observar las acciones realizadas por parte de un usuario en el sistema. Si la implementación del servidor se realizara en un computador que cuente con una pantalla integrada el puerto para conexión de una pantalla externa será igual importante en caso de que la pantalla principal llegué a presentar una falla.

- De igual forma que el punto anterior, el quinto requerimiento establece la necesidad de un puerto de conexión ethernet en el servidor para establecer una conexión cableada que brinde mayor fiabilidad a la transferencia de datos del servidor hacia la red y viceversa. dado el caso que no se tenga este puerto se necesita que al menos del servidor funcione con un módulo inalámbrico para poder brindar el acceso de los usuarios del servidor.
- En conjunto con el requerimiento dos de la tabla cuatro, en este punto se menciona nuevamente la importancia de establecer una comunicación inalámbrica para los nodos con el servidor, pero en este caso se establece la implementación de módulos compatibles con el estándar 802.11.
- Basada en la experiencia del desarrollador de este proyecto el cual ha trabajado con microcontroladores a lo largo de sus estudios universitarios, se recalca la importancia de tener un botón de reinicio en estos dispositivos o en caso de que no lo tengan buscar la manera de ejecutarlo mediante software con un botón externo para poder reiniciar el sistema programado en el microcontrolador en caso de que éste se sature por alguna razón y así corregir errores de ejecución en el sistema.
- Utilizando la experiencia del desarrollador de este proyecto como en el punto anterior, se establece como requerimiento que el voltaje de trabajo para los nodos debe ir entre los 3.3 voltios hasta los 5 voltios como máximo.
- Para el desarrollo de este sistema es importante contar con buen material de respaldo que permita buscar más de una opción para tomar una decisión y poder solucionar los problemas que puedan ir apareciendo en el proceso de desarrollo de este proyecto. Por esta razón, se establece como requerimiento el usar plataformas

de programación de código abierto y a su vez que estas plataformas cuenten con las librerías necesarias para poder trabajar con los dispositivos que se van a implementar en cuanto a los nodos clientes ubicados en cada mesa.

<p>Msc. Fabián G. Cuzme R.</p> <hr style="width: 20%; margin: auto;"/> <p>Director</p>	<p>Sr. Paúl Alejandro Zurita Z.</p> <hr style="width: 20%; margin: auto;"/> <p>Estudiante</p>
---	--

4. Anexo 4: Referencias de Costos

Los precios de referencia de los equipos ya importados al país se obtuvieron de la página web de compras Mercado Libre Ecuador. A continuación se adjuntan los enlaces de compra de los equipos que se utilizaron para el desarrollo de este proyecto.

- Módulo ESP32: https://articulo.mercadolibre.com.ec/MEC-520430594-modulo-esp32-32s-38-pines-wifi-bluetooth-iot-megatronica-JM#position=5&search_layout=stack&type=item&tracking_id=095ea5c4-73bd-403c-94e2-c46360b7f2fe
- Raspberry Pi4 de 8GB de RAM: https://articulo.mercadolibre.com.ec/MEC-524765095-raspberry-pi-4-2-gb-ram-kit-completo-cpu-ram-2-gb-computador-JM#is_advertising=true&position=1&search_layout=stack&type=pad&tracking_id=2251b18f-e554-4f81-b065-7682b852251a&is_advertising=true&ad_domain=VQCATCORE_LST&ad_position=1&ad_click_id=MmJmYzAyOGMtMTYxZC00MDc5LTkwNjktMWFkMGRhMTcyYTBl

- Pantalla LCD: https://articulo.mercadolibre.com.ec/MEC-515123284-display-lcd-oled-12864-096-i2c-ssd1306-ecuaplus-__JM#position=4&search_layout=stack&type=item&tracking_id=59871b63-72aa-4977-9d7d-c342890f0fb4
- Enrutador con soporte para el estándar 802.11ax: https://articulo.mercadolibre.com.ec/MEC-517550106-access-point-tp-link-ax20-wifi-6-ax1800-gigabit-dualband-usb-__JM#position=44&search_layout=stack&type=item&tracking_id=82b3f79e-0e3b-4630-86de-e491fe467c18
- Impresión en 3D para nodos (2,5 por hora): https://articulo.mercadolibre.com.ec/MEC-519084072-servicio-de-impresion-3d-excelente-calidad-__JM#position=1&search_layout=stack&type=item&tracking_id=81682e48-3068-4bd7-adc3-f0987cd9fb30