

ANEP
E.T.S. MALDONADO
PROYECTO
Santa Software

ASIGNATURAS

Análisis y Diseño de Aplicaciones

Formación Empresarial

Programación III

Proyecto

Sistemas de Bases de Datos II

Sistemas Operativos III

INTEGRANTES DEL EQUIPO

**Dylan Rodriguez, Diego Pozzi, Brian Matos,
Facundo Rodriguez**

AÑO

2022

Agradecimientos

El agradecimiento va a todas las personas que nos dieron una mano en cuanto al proyecto, el trato especial por parte de algunos profesores y la ayuda por parte de algunos de nuestros amigos de Arrayanes.

Queremos dar un agradecimiento especial a Carlos y su esposa, por brindarnos su restaurante para poder desarrollar.

Un agradecimiento especial a Evelyn, por guiarnos en el camino del proyecto.

Grandes agradecimientos a Carlos y Juan, por enseñarnos los conocimientos necesarios para la implementación del desarrollo.

También queremos agradecer a Sandra, por su nutritivo conocimientos sobre empresas y estructuración.

Y por último, queremos agradecerlos entre los compañeros, los unos a los otros, por aprender y adaptarse a trabajar en equipo, con todo lo que eso conlleva.

Índice

1 Empresa	8
1.1 Datos	9
1.1.1 Nombre de la empresa y logo	9
1.1.2 Emplazamiento y ubicación geográfica	9
1.1.3 Fundamentación	11
1.1.4 Actividad principal	11
1.1.5 Definición misión, visión, objetivos y valores de la organización	11
1.1.6 Forma jurídica	12
1.1.7 Trámites necesarios en los diferentes organismos estatales para su formación	15
1.1.8 Trámites específicos según actividad	19
1.1.9 Identificación y descripción del entorno de la empresa	22
1.1.10 Estudio del mercado	23
1.1.10.1 Tamaño y tendencia	23
1.1.10.2 Variables económicas	24
1.1.10.3 Aspectos regulatorios	25
1.1.10.4 Análisis FODA	26
1.1.11 Mercado objetivo	27
1.1.11.1 Determinación de la demanda	27
1.1.11.2 Poder de compra de los consumidores	27
1.1.12 Análisis de la competencia	27
1.1.12.1 Situación competitiva actual	27
1.1.13 Plan de Marketing.	27
1.1.14 Mezcla Comercial	29
1.1.15 Plan de inversiones	29
1.1.16 Plan de Recursos	29
1.1.16.1 Humanos	29
1.1.16.2 Financieros	30
2 Anteproyecto	33
2.1 Introducción.	33
2.2 Presentación del cliente.	33
2.3 Presentación del problema.	36
2.3.1 Recepción del cliente en el local	37
2.3.1.1 Descripción	37
2.3.1.2 Diagrama de actividad	37

2.3.2 Atención al cliente	38
2.3.2.1 Descripción	38
2.3.2.2 Diagrama de actividad	38
2.3.3 Registro de orden	38
2.3.3.1 Descripción	38
2.3.3.2 Diagrama de actividad	39
2.3.4 Envío de orden a la cocina	39
2.3.4.1 Descripción	39
2.3.4.2 Diagrama de actividad	40
2.3.5 Elaboración de orden	40
2.3.5.1 Descripción	40
2.3.5.2 Diagrama de actividad	41
2.3.6 Entrega de orden	42
2.3.6.1 Descripción	42
2.3.6.2 Diagrama de actividad	42
2.3.7 Ingreso de stock	42
2.3.7.1 Descripción	42
2.3.7.2 Diagrama de actividad	43
2.3.8 Pago de orden	43
2.3.8.1 Descripción	43
2.3.8.2 Diagrama de actividad	44
2.3.9 Generar ticket	44
2.3.9.1 Descripción	44
2.3.9.2 Diagrama de actividad	45
2.3.10 Registro de stock	45
2.3.10.1 Descripción	45
2.3.10.2 Diagrama de actividad	46
2.3.11 Modificación del menú	46
2.3.11.1 Descripción	46
2.3.11.2 Diagrama de actividad	47
2.4 Actores involucrados	47
2.4.1 Diagrama de actores	48
2.4.2 Descripción	48
2.5 Descripción del entorno	49
2.6 Lista de necesidades.	51
2.7 Objetivos	52
2.7.1 Matriz de trazabilidad de objetivos y necesidades	52

2.8 Lista de requerimientos	53
2.8.1 Actores del Sistema	53
2.8.2 Requerimientos funcionales	53
2.8.3 Matriz de trazabilidad de objetivos y requerimientos funcionales	62
2.8.4 Requerimientos No Funcionales	66
2.9 Alcances y limitaciones	66
2.10 Estudio de alternativa	66
2.10.1 Análisis de los sistemas operativos a implementar	68
2.10.1.1 Server Ubuntu	68
2.10.1.2 Licenciamiento	68
2.10.1.3 Soporte	69
2.10.1.4 Requisitos	70
2.10.1.5 Seguridad	71
2.10.1.6 Gestión de los usuarios	72
2.10.1.7 Gestión de red	74
2.10.1.8 Particularidades	74
2.10.1.9 Server DEBIAN	74
2.10.1.10 Licenciamiento	75
2.10.1.11 Soporte	76
2.10.1.12 Requisitos de instalación	76
2.10.1.13 Seguridad	76
2.10.1.14 Particularidades	77
2.10.1.15 Server FEDORA	78
2.10.1.16 Licenciamiento	78
2.10.1.17 Soporte	78
2.10.1.18 Requisitos de instalación	79
2.10.1.19 Seguridad	79
2.10.1.20 Particularidades	80
Integrado con GNOME 3	81
2.10.2 Alternativas de diseño	82
2.10.2.1 Escritorio	82
2.10.2.1.1 Descripción	82
2.10.2.1.2 Ventajas	82
2.10.2.1.3 Desventajas	82
2.10.2.2 Web	82
2.10.2.2.1 Descripción	82
2.10.2.2.2 Ventajas	83

2.10.2.2.3 Desventajas	83
2.10.2.3 Mixto	84
2.10.2.3.1 Descripción	84
2.10.2.3.2 Ventajas	84
2.10.2.3.3 Desventajas	84
2.10.3 Opciones de arquitectura	84
2.10.8 Selección de la alternativa	87
2.10.9 Análisis de factibilidad para la alternativa seleccionada	87
2.10.9.1 Operativa	87
2.10.9.2 Técnica.	88
2.10.9.3 Legal	88
2.10.9.4 Económica	89
2.11 Gestión de riesgos	91
2.11.1 Identificación de riesgos	91
2.11.2 Gestión de Riesgos	93
2.12 Plan de proyecto	95
2.12.1 Definición del proceso	95
2.12.1.1 Metodología	95
2.12.1.2 Ciclo de vida elegido	96
2.12.1.3 Integrantes y roles	96
2.12.1.4 Descripción de las herramientas	96
2.12.1.5 Plan de SQA	97
2.12.1.5.1 Estándares definidos y convenciones	99
2.12.1.5.2 Plan de testing	100
2.12.1.6 Plan de SCM	103
Informe GIT	103
2.12.1.7 Plan de capacitación	104
2.12.1.8 Plan de implementación	105
2.12.1.8.1 Iteración 1 (AGOSTO)	105
2.12.1.8.2 Iteración 2 (SEPTIEMBRE)	105
2.12.1.8.3 Iteración 3 (OCTUBRE)	106
2.12.1.9 Plan de implementación de SO	106
2.12.1.9.1 Hardware base	106
2.12.1.9.2 Configuración del BIOS setup	106
2.12.1.9.3 Instalación del SO servidor LAMP	106
2.12.1.9.4 Plan de seguridad	107
2.12.1.9.7 Shell script administrador	110

2.13 Cronograma de trabajo	113
2.14 Informe sobre SGBD elegido	129
Características:	131
• Mejoras en la velocidad	131
• Extensiones	131
• Errores y alertas	131
• Documentación y ayuda	131
• Licencia GPL	132
3 Proyecto	134
3.1 Análisis	134
3.2 Diseño	134
3.2.1 Diagrama de clases	135
3.2.2 Casos de Uso	138
3.2.3 Base de datos	215
3.2.3.1 MER	215
3.2.3.2 Tablas	217
3.2.3.3 Diccionario de Datos	218
3.3 Implementación	220
3.4 Gestión del proyecto	220
3.4.1 Control de avances de iteración 1(agosto)	220
3.4.1.1 Estado de situación	220
3.4.1.2 Conclusiones	220
3.4.1.3 Riesgos ocurridos	220
3.4.1.4 Modificaciones	221
3.4.2 Control de avances de iteración 2(septiembre)	221
3.4.2.1 Estado de situación	221
3.4.2.2 Conclusiones	221
3.4.2.3 Riesgos ocurridos	221
3.4.2.4 Modificaciones	222
3.4.3 Control de avances de iteración 3	222
3.4.3.1 Estado de situación	222
3.4.3.2 Conclusiones	222
3.4.3.3 Riesgos ocurridos	223
3.4.3.4 Modificaciones	223
3.5 Logros	223
3.5.1 Requerimientos implementados	223
3.5.2 Requerimientos no implementados	225

3.6 Conclusiones finales	225
3.6.1 Facundo Rodriguez	225
3.6.2 Diego Pozzi	225
3.6.3 Dylan Rodriguez	226
3.6.4 Brian Matos	226
3.6.5 Del equipo	226
3.7 Glosario	226
3.8 Bibliografía	227
3.9 Equipo	229
3.9.1 Inscripciones al equipo	229
3.9.2 Reglamento del equipo	229
3.9.3 Desarrollo de las actividades	230
3.10 Documentación complementaria	238
3.10.1 Entrevistas	238
3.11 Manual de Usuarios	241
3.11.1 Mozo	241
3.11.2 Gerente	243
3.11.3 Cocina	246
3.11.4 Caja	247
3.11.5 Administrador	248

1 Empresa

1.1 Datos

1.1.1 Nombre de la empresa y logo

- **Nombre:** Santa Software
- **Logo:** La elección de los colores del logo concuerda con el enfoque de la empresa, serio, sólido, utilizamos un espectro de colores binario, en representación de cómo es el software por dentro. El logo es una etiqueta de HTML, que a su vez posee caracteres utilizados en todos los lenguajes de programación.



FIG1. Logo Empresa

1.1.2 Emplazamiento y ubicación geográfica

Nuestra empresa tendrá lugar en un local ubicado en el centro, en frente del Campus Municipal de Maldonado en la calle 3 de Febrero, entre Rincón y Román Guerra. Esto le brindará a nuestros clientes y al equipo una mayor comodidad, al estar en el corazón de la ciudad de Maldonado.

El lugar cuenta con todo lo necesario, acceso a fibra óptica, un baño, cocina, espacio para el equipamiento, 2 ambientes, uno principal que da a la calle para atención al cliente y uno atrás donde estaría la oficina.

El lugar cuenta también con estacionamiento para los clientes.

- \$U 22.000 mensuales
- Espacio construido: 35m²
- Internet: Acceso a fibra óptica de hasta 750/60mbps
- Oficina en común
- Ambiente de atención al cliente
- Acceso a servicio de electricidad y agua
- Línea telefónica
- Zona céntrica con acceso a avenidas principales
- Cercanía a terminal y ruta de muchas líneas de omnibus
- Cercanía al sanatorio Mautone y a La Asistencial
- Estacionamiento



FIG2. Local Comercial

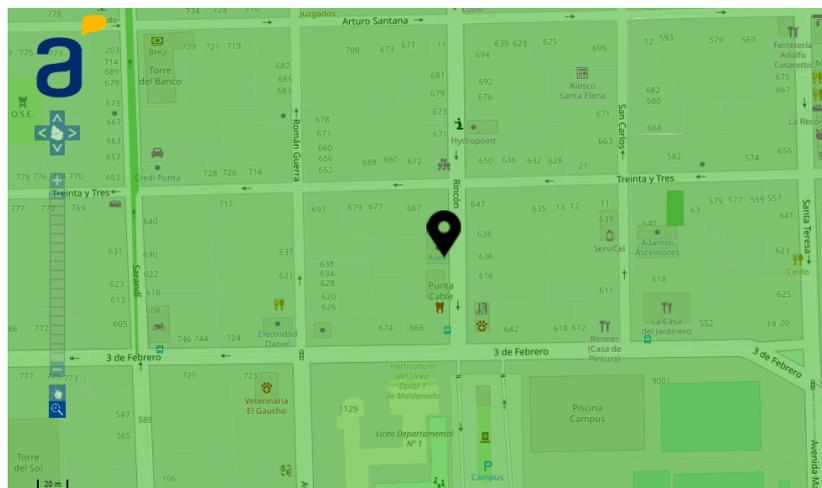


FIG3. Mapa de cobertura de Antel

Publicación del local en ML

1.1.3 Fundamentación

Se eligió el local por los siguientes puntos:

- **Costo:** \$U 22.000 mensuales
- **Espacio construido:** 35m²
- **Internet:** Acceso a fibra óptica de hasta 750/60mbps
- **Oficina en común**
- **Ambiente de atención al cliente**
- **Acceso a servicio de electricidad y agua**
- **Línea telefónica**
- **Zona céntrica con acceso a avenidas principales**
- **Cercanía a terminal y ruta de muchas líneas de omnibus**
- **Cercanía al sanatorio Mautone y a La Asistencial**
- **Estacionamiento**

1.1.4 Actividad principal

La actividad principal consistirá en el desarrollo, personalizado acuerdo a un cliente, de software, proveyendo a su vez soporte y servicios tanto de hosting como técnico.

1.1.5 Definición misión, visión, objetivos y valores de la organización

Razón: S.R.L porque requiere una baja inversión inicial y nos da la libertad de ser 2 o más socios. Además, es una constitución rápida y económica.

Misión: Desarrollo de software orientado a brindar soluciones a empresas.

Visión: Ser pioneros en brindar soluciones tecnológicas a empresas.

Valores: responsabilidad, transparencia y puntualidad.

Objetivo General: Cubrir un mercado altamente solicitado, lograr consagrarse como un modelo a seguir por otras empresas al ser la preferencia de la mayoría de clientes, cubrir espectro más amplio de soluciones informáticas reclutando talento regional, conseguir clientela internacional.

Objetivos específicos: Hacer 10 desarrollos para empresas en 2 años.

Mudarnos en 3 años a una oficina más grande y con más capacidad de trabajo, publicitar nuestro servicio en todos los medios posibles, generar ingresos pasivos.

1.1.6 Forma jurídica

SOCIEDAD DE RESPONSABILIDAD LIMITADA. En la ciudad de Maldonado, el día 28 de mayo de 2022, comparecen:

I) Facundo Rodríguez, Uruguayo, mayor de edad, soltero, CI 5520.5321, calle Albatros entre Ñandú y Cisnes 918.

II) Dylan Rodriguez, Uruguayo, mayor de edad, soltero, 54634557, Calle 11 Altos de la Laguna.

III) Diego Pozzi, Uruguayo, mayoría de edad, soltero, 55945099, calle Mariano Soler, San Carlos.

IV) Brian Matos, Cubano, mayor de edad, soltero, 64202634, calle Manolo Lima, Lausana

Conviene en celebrar el siguiente contrato:

PRIMERO (Denominación).- Los comparecientes convienen en constituir una SOCIEDAD DE RESPONSABILIDAD LIMITADA que se regirá por la Ley 16.060, el decreto 155/010 de 24 de mayo de 2010, y demás normas concordantes y complementarias, para la que adoptan la denominación de (Santa Software) S.R.L.

SEGUNDO (Domicilio) – La sociedad tendrá su domicilio en el departamento de Maldonado, pudiendo instalar sucursales dentro y fuera del país.

TERCERO (Objeto) – El objeto de la sociedad que se constituye será el de: desarrollo de software.

En cumplimiento de dicho objeto la sociedad podrá realizar todo tipo de actos.

CUARTO (Capital y cuotas) – El capital social asciende a la suma de \$300.000 (trescientos mil pesos uruguayos), dividido en 4 cuotas de \$75000 (setenta y cinco mil pesos uruguayos) cada una.

QUINTO (Aportes) – Los socios, a efectos de la conformación del capital social, aportan en este acto, al contado y en efectivo las siguientes sumas:

I) Facundo Rodriguez, la suma de \$75.000 pesos uruguayos, cien mil, correspondiéndole 1 cuota;

II) Dylan Rodriguez, la suma de \$75.000 pesos uruguayos, cien mil, correspondiéndole 1 cuota;

III) Diego Pozzi la suma de \$75.000 pesos uruguayos, (cien mil), correspondiéndole 1 cuota;

IV) Brian Matos, la suma de \$75.000 pesos uruguayos, cien mil, correspondiéndole 1 cuota;

La sociedad otorga a cada uno de los socios la correspondiente carta de pago.

SEXTO (Distribución de ganancias y pérdidas) –Las ganancias se distribuirán y las pérdidas se soportarán en proporción a lo aportado.

SÉPTIMO (Administración) – La administración y representación de la sociedad corresponderá a cualquiera de los socios en forma indistinta. El representante tendrá las más amplias facultades para el gobierno, administración, gravamen y disposición de los bienes sociales. Podrá otorgar poderes generales y especiales.

OCTAVO (Plazo) – El plazo de la sociedad que se constituye es de 30 años a partir de hoy. Dicho plazo se prorrogará automáticamente por períodos de 5 años cada uno, a contar del vencimiento del plazo original. Los socios podrán ejercer el derecho de receso cada 3 años a contar del presente retirándose de la sociedad y continuando ésta con los restantes, debiendo comunicar su decisión mediante telegrama dirigido a la Sede social y a la sociedad con una antelación no menor a sesenta días a contar del respectivo vencimiento.

NOVENO (Cesión de cuotas) – La cesión de cuotas se registrá por el régimen legal.

DÉCIMO (Toma de decisiones) – Las decisiones sociales se adoptarán conforme a lo dispuesto por la normativa vigente.

DÉCIMO PRIMERO (Pacto de continuación) - En caso de fallecimiento o incapacidad de cualquiera de los socios la sociedad continuará con los sucesores o cónyuge supérstite del socio fallecido o con el representante del socio incapaz.

DÉCIMO SEGUNDO (Telegrama colacionado y domicilios especiales) – Las partes pactan la validez del telegrama colacionado para toda comunicación, notificación e intimación entre ellas y como domicilios especiales los denunciados como suyos en la comparecencia.

DÉCIMO TERCERO (Solicitud de certificación de firmas) – Las partes solicitan al Escribano Dilan Santos la certificación notarial de sus firmas.

CERTIFICO QUE: I) Las firmas que anteceden en el contrato de Sociedad de Responsabilidad Limitada que antecede, identificado con el pié de página son auténticas y fueron puestas en mi presencia por las personas hábiles Diego Pozzi, Brian

Matos, Facundo Rodriguez, Dylan Rodriguez acreditando su identidad con los correspondientes documentos de identidad que me exhiben, y cuyos demás datos individualizantes surgen del referido documento, que previa lectura que del mismo les hice lo otorgaron y firmaron. II) El contrato que antecede fue otorgado conforme lo dispuesto en la Ley 16.060 y el decreto 155/010 de 24 de mayo de 2010. III) El domicilio fiscal de (denominación social sin aditamento) S.R.L. es e hi n la calle Rincon número 123, entre las calles 3 de Febrero y Treinta y Tres de la ciudad de Maldonado del departamento de Maldonado, y

su domicilio constituido en la calle Rincon número 123, entre las calles 3 de Febrero y Treinta y Tres de la ciudad de Maldonado del departamento de Maldonado. EN FE DE ELLO, a solicitud de (denominación social sin aditamento) S.R.L., y para su presentación ante Registro Nacional de Comercio, Dirección General Impositiva y Banco de Previsión Social, expido el presente que sello, signo y firmo en Maldonado, departamento de Maldonado el 6 de Junio del dos mil veintidós.

1.1.7 Trámites necesarios en los diferentes organismos estatales para su formación

		DECLARACIÓN DE REGISTRO INSCRIPCIÓN Y ACTUALIZACIÓN EMPRESAS Y OTRAS ENTIDADES UNIPERSONALES O PLURIPERSONALES				0351  VERSIÓN 05													
RUBRO 1 - IDENTIFICACIÓN																			
D. O. B.	Nº de REGISTRO DE CONTRIBUYENTE		Nº de REGISTRO EMPRESA		Nº DE RUT														
	85256373		89494994		10787758571														
Acto que se realiza			BPS	DGI	Vigencia del acto														
Inicio de actividades			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Día	Mes													
Reinicio de actividades			<input type="checkbox"/>	<input type="checkbox"/>	Año														
Modificación (completar sólo los campos que se actualizan)			<input type="checkbox"/>	<input type="checkbox"/>	26	6													
2022																			
RUBRO 2 - DATOS DE LA ENTIDAD																			
Nombre o denominación saintainc																			
Tipo de entidad																			
Persona física	<input type="checkbox"/>	Núcleo familiar	<input type="checkbox"/>	Condominio	<input type="checkbox"/>	Sucesión indivisa													
Sociedad de hecho	<input type="checkbox"/>	Sociedad colectiva	<input type="checkbox"/>	Sociedad de respons. limitada	<input checked="" type="checkbox"/>	Soc. en comandita simple													
Soc. en comandita por acc.	<input type="checkbox"/>	SA con acciones nominativas	<input type="checkbox"/>	SA con acciones al portador	<input type="checkbox"/>	Cooperativas													
Pers. de derecho público	<input type="checkbox"/>	Asociaciones/Fundaciones	<input type="checkbox"/>	Entidad pluripersonal no resid.	<input type="checkbox"/>	SAS acciones nominativas													
Fideicomiso	<input type="checkbox"/>	Grupo de interés económico	<input type="checkbox"/>	Asoci./soc. agrarias	<input type="checkbox"/>	SAS acciones escriturales													
Comisión Administradora	<input type="checkbox"/>	Represen. diplomát. extranjeras	<input type="checkbox"/>	Organismos internacionales	<input type="checkbox"/>	Otros													
Fecha de constitución			Inscripción en el		Número														
Día Mes Año			Registro Nal Com.		Día Mes Año														
26 7 2022																			
Nombre de fantasía santa software																			
Residente			No residente con		País de residencia														
<input checked="" type="checkbox"/>			Establecimiento permanente		<input type="checkbox"/>														
RUBRO 3 - DOMICILIO FISCAL DEL LOCAL PRINCIPAL																			
Calidad en que ocupa el domicilio fiscal																			
<input type="checkbox"/> Propietario <input type="checkbox"/> Arrendatario <input type="checkbox"/> Sub-arrendat. <input type="checkbox"/> Comodatario <input type="checkbox"/> Usufructuario <input type="checkbox"/> Otros																			
Departamento		Localidad		Calle		Código postal													
Maldonado		Maldonado		Calle 11		20000													
Complemento del domicilio																			
Nº torre, Edif., Block		Calle interna		Manzana		Solar													
Otros detalles de ubicación: (E. Entre calle y calle)																			
Contactos																			
Persona o empresa			Teléfono fijo		Teléfono móvil														
Andrea Camblong					099578722														
					andrea777camblong@hotmail.com														
RUBRO 4 - DOMICILIO CONSTITUIDO																			
Indicar si el Domicilio Constituido es igual al Fiscal																			
<input type="checkbox"/> Si <input type="checkbox"/> No																			
Departamento		Localidad		Calle		Código postal													
Complemento del domicilio																			
Nº torre, Edif., Block		Calle interna		Manzana		Solar													
Otros detalles de ubicación: (E. Entre calle y calle)																			
Contactos																			
Persona o empresa			Teléfono fijo		Teléfono móvil														
RUBRO 5 - ACTIVIDAD ECONÓMICA																			
Alta Baja Grupo Sub G Cap Band																			
Actividad Principal																			
Actividad Secundaria																			
Actividad Secundaria																			
RUBRO 6 - OBLIGACIONES																			
Alta	Baja	Obligación	Características					Alta	Baja	Obligación	Características								
		IRAE	01	02	03	04	05			IMESI	NUM								
		IVA - Construc.								IMESI	NUM								
		IVA - Serv. Pers.								IMESI	NUM								
		IVA - AGROP.								IMESI	NUM								
		IVA - GRAL	<input checked="" type="checkbox"/>							IMESI	NUM								
		IVA - Peg. Emp.								MONOTRIBUTO									
		PAT - Entidades								RPF I		<input checked="" type="checkbox"/>							
		PAT - P.F.								RPF II									
		IIP								RNR									
		ICOSA								PRIMARIA									
		IMEBA								FIS									
		IMEBA ADIC.								ENT.ASEGURA.									
Combina Capital y Trabajo		SI		NO															

FIG4. Registro BPS

		N° de RUT		10787758571	
RUBRO 7 - OTROS DATOS					
Condición					
Importador	Exportador	Asimilado a Exportador	Imprenta Autorizada		
Usuario directo Z. Franca	Usuario indirecto Z. Franca	Administradora de Crédito	Entidad Colaboradora		
Establecim. De Temporada	No Contribuyente	Ent.no resid. sin activ. empr.	Ent. resid. Atribuidora de rent.		
Imprenta Autorizada WEB	Comercio autorizado Tax free	Registro INAC - CVA 90 días	Registro INAC - CVA 30 días		
Exonerado	Otros (detallar)				
Balance		Día	Mes	Día	Mes
		27	7		
Antecedentes					
En caso de inicio o reinicio por		Compra Total	Compra Parcial	Sucesión	Escisión
Transferencia, indicar la forma		Conversión Total	Conversión Parcial	Otros (detallar)	
Nombre o denominación del		N° de Registro de Empresa		N° de R U T	
antecesor				10787758571	
sucesor					
RUBRO 8 - RÉGIMEN DE APORTACIÓN AL B.P.S.					
Industria y comercio	<input checked="" type="checkbox"/>	Servicios Personales	<input type="checkbox"/>	Rural	<input type="checkbox"/>
Construcción	<input type="checkbox"/>	Civil	<input type="checkbox"/>		
RUBRO 9 - DATOS DEL TITULAR					
Persona Física (indicar)					
			Alta <input checked="" type="checkbox"/>	Permanencia	
			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Tipo de Vinculos					
Dueño		<input type="checkbox"/>	Director	<input type="checkbox"/>	Socio
Ingresar Código (*)		<input type="checkbox"/>	<input type="checkbox"/>	02	01
Socio sin Administración		<input type="checkbox"/>	Socio Administrador conjunto	<input type="checkbox"/>	Socio Administrador indistinto
Representante		<input type="checkbox"/>	Representante Legal SAS	<input type="checkbox"/>	Otro
Tipo de documento		N° de documento		País de origen del documento	
C.I.	<input checked="" type="checkbox"/>	DNI	Pasaporte	54634556	uruquay
1° Apellido		2° Apellido		1° Nombre	2° Nombre
Pozzi		Gramajo		Diego	Adrian
Fecha de Nacimiento	Estado Civil	Sexo	Firma		
23/03/2005	soltero	h			
Tipo de residencia		<input type="checkbox"/>	NO Residente	<input checked="" type="checkbox"/>	Residente
		País de residencia uruquay			
Domicilio Particular					
Departamento		Localidad	Calle	Número	Adjunto
Maldonado		san carlos	mariano soler	990	
Complemento del domicilio		N° torre, Edif., Block	Calle interna	Manzana	Solar
Ruta	Km.	Nombre del establecimiento	N° de Padrón	N° de Secc. Judicial	N° de Secc. Policial
Otros detalles de ubicación: (E). Entre calle y calle)					
Teléfono fijo	Fax	Teléfono móvil	Correo electrónico	Otro correo electrónico o contacto	
42660528		098464425	Diegopozzi2@gmail.com		
Persona Física (indicar)					
			Alta	Permanencia	
			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Tipo de Vinculos					
Dueño		<input type="checkbox"/>	Director	<input type="checkbox"/>	Socio
Ingresar Código (*)		<input type="checkbox"/>	<input type="checkbox"/>	02	01
Socio sin Administración		<input type="checkbox"/>	Socio Administrador conjunto	<input type="checkbox"/>	Socio Administrador indistinto
Representante		<input type="checkbox"/>	Representante Legal SAS	<input type="checkbox"/>	Otro
Tipo de documento		N° de documento		País de origen del documento	
C.I.	<input checked="" type="checkbox"/>	DNI	Pasaporte	54634556	uruquay
1° Apellido		2° Apellido		1° Nombre	2° Nombre
Rodriguez		Camblong		Dylan	Alis
Fecha de Nacimiento	Estado Civil	Sexo	Firma		
15/10/2004	Soltero	Masculino			
Tipo de residencia		<input type="checkbox"/>	NO Residente	<input checked="" type="checkbox"/>	Residente
		País de residencia Uruquay			
Domicilio Particular					
Departamento		Localidad	Calle	Número	Adjunto
Maldonado		Maldonado	Calle 11		
Complemento del domicilio		N° torre, Edif., Block	Calle interna	Manzana	Solar
Ruta	Km.	Nombre del establecimiento	N° de Padrón	N° de Secc. Judicial	N° de Secc. Policial
Otros detalles de ubicación: (E). Entre calle y calle)					
Teléfono fijo	Fax	Teléfono móvil	Correo electrónico	Otro correo electrónico o contacto	
		099123321	dylanrodriguezccamblonga@gi		
Observaciones					
Firma del Representante o Autorizado					TIMBRE PROFESIONAL
Aclaración de firma					
Documento de Identidad					
El/los firmante/s declara/n que los datos establecidos en este formulario son correctos y completos y que conoce/n las sanciones aplicables en caso de falsa declaración, previsto en los Art. 96 del Código Tributario y 239 del Código Penal.					
BORRAR DATOS			IMPRIMIR		

FIG5. Registro BPS 2

 Instituto de Seguridad Social	DECLARACIÓN DE REGISTRO ACTUALIZACIÓN / ANEXO DE INSCRIPCIÓN PERSONAS FÍSICAS VINCULADAS EMPRESAS Y OTRAS ENTIDADES UNIPERSONALES O PLURIPERSONALES				0352 VERSIÓN 03	 DIRECCIÓN GENERAL IMPOSITIVA	
	RUBRO 1 - IDENTIFICACIÓN						
N° de REGISTRO DE CONTRIBUYENTE 89494994		N° de REGISTRO EMPRESA 89494994		N° de RUT 10787758571		Vigencia del acto Día Mes Año 26 6 2022	
RUBRO 10- DATOS DE PERSONAS FÍSICA VINCULADAS							
Persona Física Vinculada (Indicar)							
				Alta		Permanencia	
Tipo de Vinculos Ingresar Código		Dueño <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/> Director <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/> Socio <input type="checkbox"/> 02 <input type="checkbox"/> 01 <input type="checkbox"/> Cónyuge colab. <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/> Administrador <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/> Sindico <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/>		Socio sin Administración <input type="checkbox"/> Representante <input type="checkbox"/>		Socio Administrador conjunto <input type="checkbox"/> Otro (detallar) <input type="checkbox"/>	
Tipo de documento C.I. <input checked="" type="checkbox"/> DNI <input type="checkbox"/> Pasaporte <input type="checkbox"/>		N° de documento 55205321		País de origen del documento Uruguay			
1° Apellido Facundo		2° Apellido Rodriguez		1° Nombre Facundo		2° Nombre Rodriguez	
Fecha de Nacimiento Estado Civil Sexo		Firma					
Tipo de residencia No Residente <input type="checkbox"/> Residente <input checked="" type="checkbox"/>		País de residencia Uruguay					
Domicilio Particular							
Departamento Maldonado		Localidad Maldonado		Calle maldonado park albatros 03		Número Adjunto Apart. Código postal 03 A 03 20000	
Complemento del domicilio N° torre, Edif., block Otros detalles de ubicación: (E.): Entre calle y calle; Ruta, Km. Nombre establecimiento, No. de Padrón, Sec. Jud., Sec. Pol., Paraje)		Calle interna Manzana Solar					
Teléfono fijo 42226789		Fax Teléfono móvil 099032109		Correo electrónico facuroduy@gmail.com		Otro correo electrónico o contacto	
En los casos de Persona Física RESIDENTE sin actividad empresarial, el domicilio que se declara corresponde al domicilio fiscal a todos los efectos tributarios.							
Persona Física Vinculada (Indicar)							
				Alta		Permanencia	
Tipo de Vinculos Ingresar Código		Dueño <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/> Director <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/> Socio <input type="checkbox"/> 02 <input type="checkbox"/> 01 <input type="checkbox"/> Cónyuge colab. <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/> Administrador <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/> Sindico <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/>		Socio sin Administración <input type="checkbox"/> Representante <input type="checkbox"/>		Socio Administrador conjunto <input type="checkbox"/> Otro (detallar) <input type="checkbox"/>	
Tipo de documento C.I. <input checked="" type="checkbox"/> DNI <input type="checkbox"/> Pasaporte <input type="checkbox"/>		N° de documento 64202634		País de origen del documento Uruguay			
1° Apellido Matos		2° Apellido Mederos		1° Nombre Brian		2° Nombre	
Fecha de Nacimiento 28/04/2004		Estado Civil Soltero		Sexo Masculino		Firma	
Tipo de residencia No Residente <input type="checkbox"/> Residente <input type="checkbox"/>		País de residencia					
Domicilio Particular							
Departamento Maldonado		Localidad Maldonado		Calle Manolo Lima, Lausana		Número Adjunto Apart. Código postal 029 20000	
Complemento del domicilio N° torre, Edif., block Otros complementos o detalles de ubicación: (E.): Entre calle y calle; Ruta, Km. Nombre establecimiento, No. de Padrón, Sec. Jud., Sec. Pol., Paraje)		Calle interna Manzana Solar					
Teléfono fijo		Fax Teléfono móvil 096128082		Correo electrónico brayanmatosk510@gmail.com		Otro correo electrónico o contacto	
En los casos de Persona Física RESIDENTE sin actividad empresarial, el domicilio que se declara corresponde al domicilio fiscal a todos los efectos tributarios.							
Persona Física Vinculada (Indicar)							
				Alta		Permanencia	
Tipo de Vinculos Ingresar Código		Dueño <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/> Director <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/> Socio <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/> Cónyuge colab. <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/> Administrador <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/> Sindico <input type="checkbox"/> VF <input type="checkbox"/> SS <input type="checkbox"/>		Socio sin Administración <input type="checkbox"/> Representante <input type="checkbox"/>		Socio Administrador conjunto <input type="checkbox"/> Otro (detallar) <input type="checkbox"/>	
Tipo de documento C.I. <input type="checkbox"/> DNI <input type="checkbox"/> Pasaporte <input type="checkbox"/>		N° de documento		País de origen del documento			
1° Apellido		2° Apellido		1° Nombre		2° Nombre	
Fecha de Nacimiento		Estado Civil		Sexo		Firma	
Firma del Representante o Autorizado		Aclaración de Firma					

FIG6. Registro BPS 3



DECLARACION JURADA
DE TRABAJADORES
NO DEPENDIENTES
APORTACION INDUSTRIA Y COMERCIO
REC. 205

SOLO PARA TRANSACCIONES DEL INTERIOR			
LOCAL DE B.P.S.			COR.
N° DE TRANSACCION			
FECHA DE TRANSACCION	DIA	MES	AÑO
	26	6	2022

IDENTIFICACION CONTRIBUYENTE	RUT C.I. ATYR	10787 7958571
N°	85256373	

DENOMINACION: _____
_____ santa software

DOCUMENTO IDENTIDAD		APELLIDOS Y NOMBRES	N° EMPRESA Ind. y Comercio por la que Aporta	INDIQUE FECHA DESDE	INDIQUE FECHA HA STA	OPCION DE SUELDO*		FIRMA
TIPO	NUMERO					CAT. FICTA	REM. REAL	
CI	54634556	Rodriguez, Dylan		26/6/2022	/		✓	
CI	55945099	Pozzi, Diego		26/6/2022	/		✓	
CI	55205321	Rodriguez, Facundo		26/6/2022	/		✓	
CI	64202634	Matos, Brian		26/6/2022	/		✓	

FIRMA del Representante o Autorizado Aclaración de FIRMA Documento Identidad:	Firma y N° Funcionario Receptor _____	El/los firmante/s declara/n que los datos establecidos en este formulario son correctos y completos y que <u>conocen</u> las sanciones aplicables en caso de falsa declaración, previsto en los art. 96 del Código Tributario y 239 del Código Penal.	Timbre Profesional
	Firma y N° Funcionario Revisor _____		
	Firma y N° Funcionario Digitador _____		

FIG7. Registro BPS 4

1.1.8 Trámites específicos según actividad



Intendencia de Maldonado

CONSULTA

No.: **27836**

VIABILIDAD PARA INSTALACION DE:

ACTIVIDAD COMERCIAL
 ACTIVIDAD INDUSTRIAL

DATOS DEL COMERCIO:

DIRECCION: Rincón y Ramán Guerra, 3 de Febrero TEL./FAX: 92 22 6912

PADRON: MANZANA: LOCALIDAD: Maldonado

GIRO: Comercial

DESCRIPCION DE LA ACTIVIDAD: Oficina de desarrollo de software

DATOS DEL INTERESADO:

NOMBRE: Brian Matos C.I.: 64202634

DOMICILIO: Mando Lima el N. Blanes y M. Joro TEL./FAX: 92 22 6912

DATOS COMPLEMENTARIOS: ANTECEDENTES DE:

AREA EDIFICADA: 35m² EXPTE. OBRA: FINAL:

28 6 2022	
Fecha	Funcionario

Brian
Firma

85.09.13 original - oficina



Intendencia de Maldonado

CONSTANCIA DE CONSULTA

No.: **27836**

(VIABILIDAD PARA INSTALACION DE ACTIVIDAD COMERCIAL O INDUSTRIAL)

FIG8. Consulta viabilidad



Intendencia de Maldonado
CONSULTA

No.: **27836**

VIABILIDAD PARA INSTALACION DE:

ACTIVIDAD COMERCIAL
 ACTIVIDAD INDUSTRIAL

DATOS DEL COMERCIO:

DIRECCION: Rincón y Roman Guerra, 3 de febrero TEL./FAX: 92 22 6912

PADRON: MANZANA: LOCALIDAD: Maldonado

GIRO: Comercial

DESCRIPCION DE LA ACTIVIDAD: Oficina de desarrollo de software

DATOS DEL INTERESADO:

NOMBRE: Brian Matos C.I.: 64202631

DOMICILIO: Manda Lima e/ N. Blanes y M. D. M. TEL./FAX: 92 22 6912

DATOS COMPLEMENTARIOS: ANTECEDENTES DE:

AREA EDIFICADA: 35 m² EXPTE. OBRA: FINAL:

28 6 2022	
Fecha	Funcionario


 Firma

DIRECCION DE HIGIENE Y BROMATOLOGIA

RESOLUCION: NO es viable SI es viable

Rubro autorizado y validez:

La presente constancia sólo da derecho a la iniciación de los trámites de presentación de planos o solicitud de Habilitación Higiénica. **NO** da permiso a iniciación de obras, instalación de equipamientos o maquinarias, mejoras de terrenos o edificaciones.

FECHA: FIRMA:

FIG9. Consulta viabilidad 2



Intendencia de Maldonado N° 74833

CONTRALOR DE HIGIENE AMBIENTAL

R.U.T.
N° 10787758571
B.P.S.
N° 89494994
CORRELATIVO
N°

SOLICITUD
 MODIFICACION
 CLAUSURA

I) IDENTIFICACION:

NOMBRE COMERCIAL: Santa Software
 DIRECCION DEL COMERCIO: 3 de Febrero e/ Rincón y Román Guerra
 GIRO: Comercial PADRON: _____ MANZANA: _____
 LOCALIDAD: Maldonado

TITULAR DEL COMERCIO: Ramón Suarez C.I.: 35517903
 DIRECTOR/ADMINISTRADOR: Dylan Rodriguez C.I.: 54636671
 DOMICILIO PARTICULAR: Mando Lima, Lonsana, Maldonado
 NOTIFICAR A: Teléfonos: 42.22.6912 Fax/ Correo Elect: _____

II) DATOS COMPLEMENTARIOS

A) AREA EDIFICADA (incluye pasiva) 35m² B) HABITACIONES (Hoteles y afines)
 C) Vínculo que se mantiene con la propiedad:
 Propietario Inquilino
 Préstamo Comodato

III) TASA ANUAL:

Recibo N° _____ de fecha: _____ por \$ _____

IV) DOCUMENTACION:

<input type="checkbox"/> Certificado Notarial de Propiedad	<input checked="" type="checkbox"/> Habilitación de Bomberos
<input checked="" type="checkbox"/> Contrato de Alquiler	<input type="checkbox"/> Anuencia Copropietarios
<input type="checkbox"/> Comodato o Préstamo	<input type="checkbox"/> Acondicionamiento Acústico
<input checked="" type="checkbox"/> Contrato Social	<input type="checkbox"/> Impacto Ambiental
<input type="checkbox"/> Poder a: _____	

Fecha	Dirección de Higiene y Brom.	Firma Titular	C.I.
6 28/2002			54634556

OBSERVACIONES:

85.09.03



Intendencia de Maldonado N° 74833

CONSTANCIA DE TRAMITE

FIG10. Consulta CHA

1.1.9 Identificación y descripción del entorno de la empresa

Nuestra empresa está establecida en una zona céntrica de gran concurrencia de peatones y vehículos. Rodeado de varias empresas financieras, la intendencia de Maldonado, Sanatorios, locales gastronómicos, servicios diversos y un Shopping en construcción. También, es cercana a la terminal de Maldonado, aumentando así su visibilidad.

Nuestro establecimiento cuenta con un estacionamiento propio, ideal para que nuestros clientes no pierdan tiempo buscando donde estacionar cuando se reúnen en nuestras oficinas. Contamos con un gran espacio para instalar cartelería y realizar Branding de la empresa.

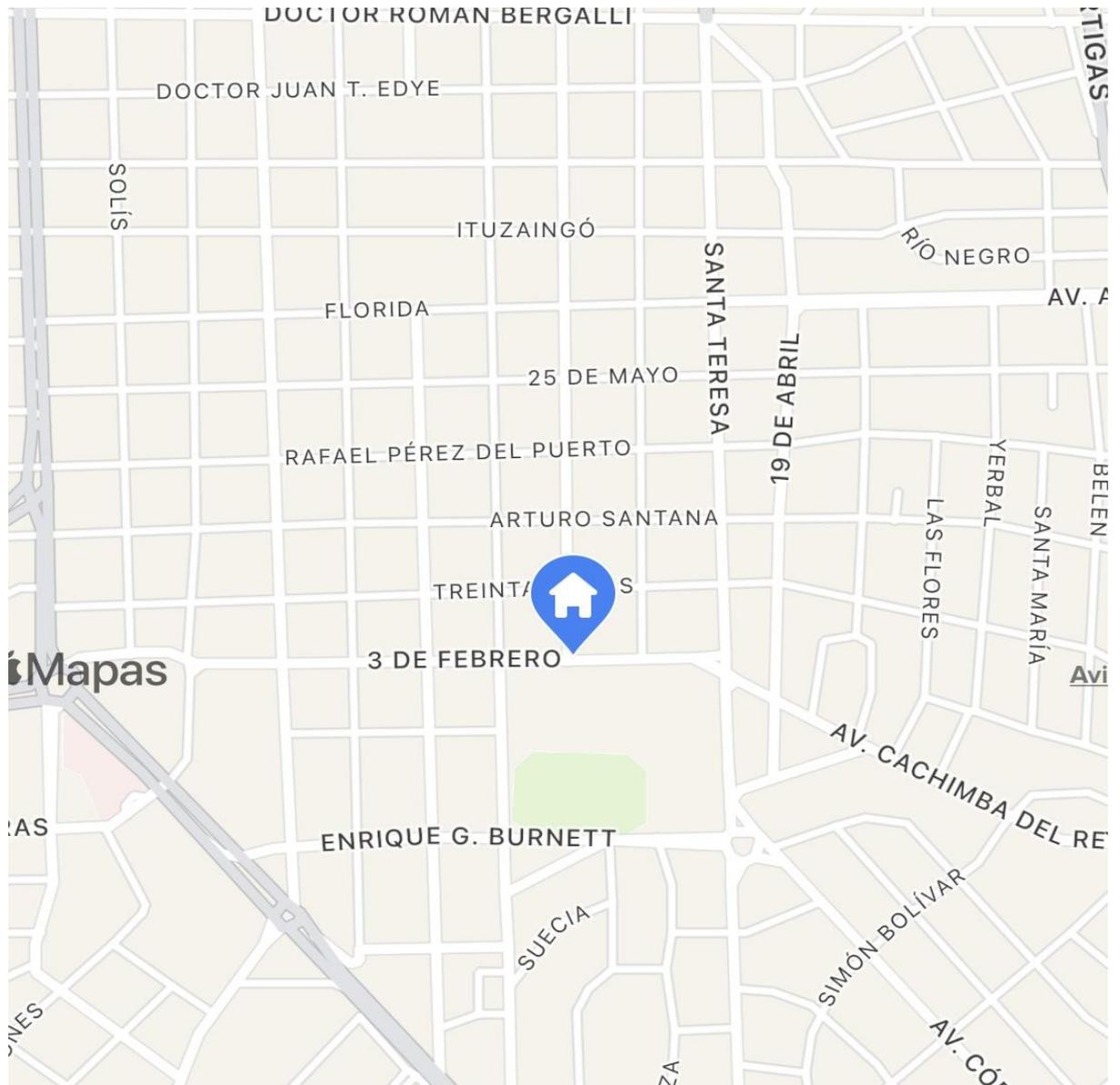


FIG11. Google maps local

1.1.10 Estudio del mercado

1.1.10.1 Tamaño y tendencia

Según los últimos datos disponibles, correspondientes al año 2017. Con un 65,9 % del total, las exportaciones de software tuvieron como principal destino EE. UU. y las siguientes posiciones en el ranking las ocuparon Argentina, Irlanda, México, Chile y España. En total, se exporta software a más de 52 países del mundo.

Los últimos datos indican que los servicios de software prestados por empresas radicadas en el extranjero a empresas uruguayas alcanzaron la cifra de 110 millones de dólares, lo que supone un aumento del 220 % en diez años.

Gracias a estas estadísticas, podemos identificar que estamos desarrollando en una industria de alto crecimiento y de gran facturación, tanto de mercado interno como de exportación.

1.1.10.2 Variables económicas

INFLACIÓN:

¿Qué es la inflación?

La inflación es definida como la variación o alza de los precios de los bienes y servicios en el mercado. En términos concretos, la inflación no es más que el desequilibrio o la diferencia que se da entre el precio de la mayor parte de los productos o servicios y la pérdida de valor del dinero en un país.

Actualmente en Uruguay estamos aproximadamente en el 10% anual de inflación.

¿Cómo nos afecta?

En primer lugar, en el aumento de los costos de algunos insumos y materiales, transporte o piezas, combustibles, desarrollos de terceros, servicios de software, etc. El incremento de costos puede inducir a aumentar los precios de los productos o servicios que ofrecemos. Por otra parte, a la hora de acceder a créditos o préstamos, es posible que las tasas aumenten.

En las cadenas productivas, dependemos de empresas grandes que pueden ser potenciales compradores. Por ello, somos susceptibles a las fluctuaciones de los principales indicadores de la macroeconomía.

PBI

¿Qué es el PBI?

El PBI es una medida económica. Este porcentaje nos revela que la salud de la economía se recupera y avanza. Si el PIB indica una evolución, tenemos parámetros para interpretar que la economía Uruguaya está creciendo, pues la productividad de nuestro país incrementa y por lo tanto, es una buena señal que indica que hay mayores probabilidades de “encontrar un buen trabajo, o que nos aumente el sueldo”. Al contar con un ambiente propicio económico, tendremos mayor confianza en la estabilidad de los ingresos. Lo anterior impulsa el desempeño económico de nuestro país.

¿Como nos afecta?

Las empresas utilizamos los pronósticos del PIB para evaluar la situación de la economía y tomar decisiones importantes en cuanto a la expansión de las actividades, por ejemplo, pueden pensar en invertir en nuevos desarrollos, expansiones o negocios y generar más fuentes de empleo. El PIB también funciona para tener idea de cómo se comportarán los precios y las tasas de interés en un futuro. En

cuanto al resto del mundo, le envía señales para saber si es oportuno invertir en nuestro país.

1.1.10.3 Aspectos regulatorios

En nuestra actividad, estaremos regulados por distintas leyes y decretos. Una de ellas es la Ley N° 18331, que regulará la protección y cuidado de datos que resulten de nuestro software.

Además, nuestra sociedad está amparada y regulada por la Ley de Sociedades Comerciales, Ley N° 16060 y por la Ley 16.637 & Decreto 244/018 – Software y servicios vinculados, también por el Decreto n° 64/020, de 17 de febrero de 2020. Especialmente, de la ley 16.060:

CAPITULO II - DE LAS SOCIEDADES EN PARTICULAR

SECCION IV - DE LAS SOCIEDADES DE RESPONSABILIDAD LIMITADA

Artículo 223

(Caracterización). En las sociedades de responsabilidad limitada el capital se dividirá en cuotas de igual valor, acumulables e indivisibles, que no podrán ser representadas por títulos negociables. La responsabilidad de los socios se limitará a la integración de sus cuotas.

El número de socios no excederá de cincuenta.

Si por cualquier circunstancia llegara a tener un número superior, deberá transformarse en sociedad anónima en el plazo de dos años, bajo sanción de disolución, salvo que en ese plazo el número de los socios se reduzca a cincuenta o menos.

1.1.10.4 Análisis FODA

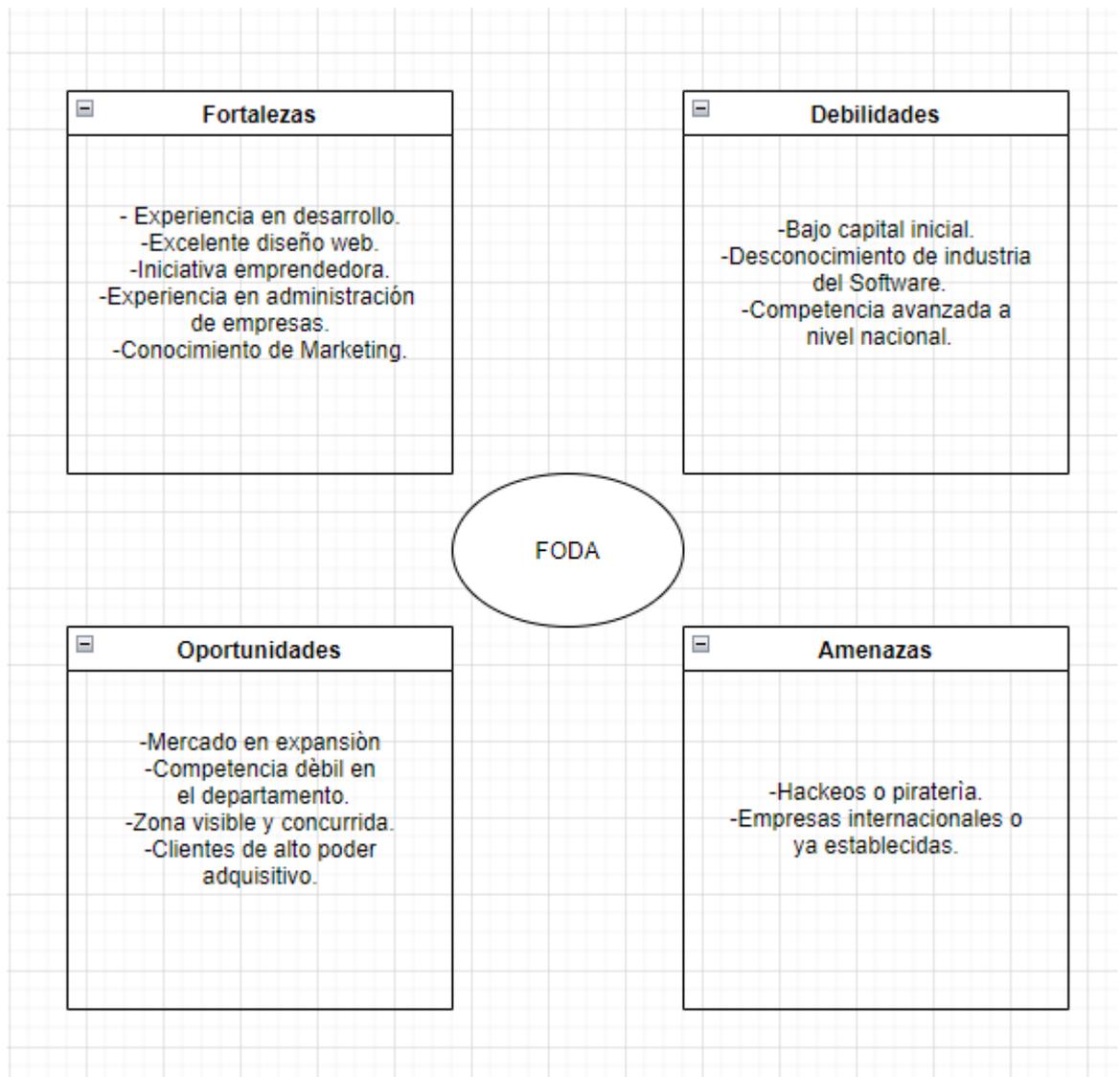


FIG12. Analisis F.O.D.A

1.1.11 Mercado objetivo

1.1.11.1 Determinación de la demanda

Nuestro Mercado objetivo son empresas medianas/grandes que busquen soluciones de software para gestionar todos los aspectos de su empresa.

1.1.11.2 Poder de compra de los consumidores

Apuntamos a empresas Medianas/Grandes que estén en periodo de crecimiento exponencial y necesiten soluciones de software e infraestructura digital.

1.1.12 Análisis de la competencia

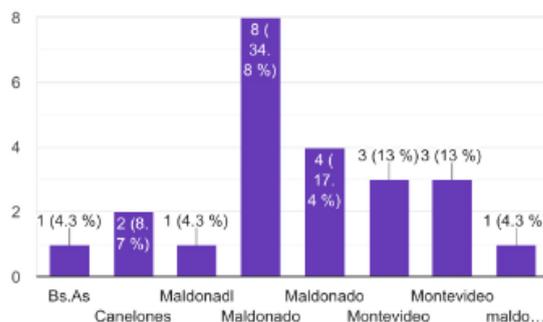
1.1.12.1 Situación competitiva actual

1.1.13 Plan de Marketing.

Encuestas:

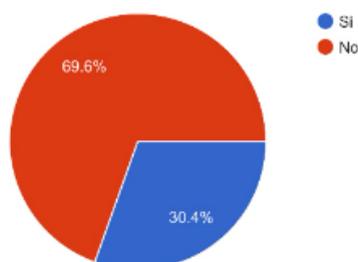
¿En que departamento resides?

23 respuestas



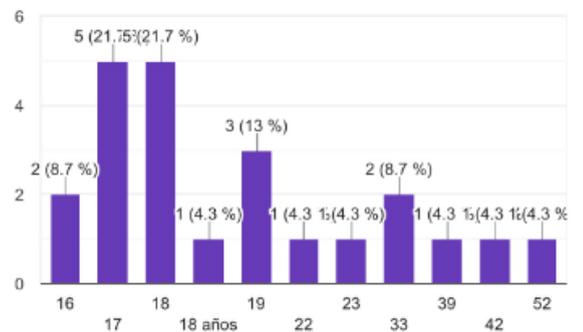
¿Te dedicas al software o trabajas en una empresa de la industria?

23 respuestas



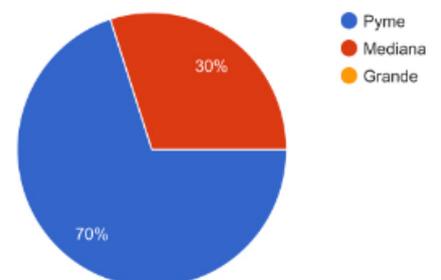
¿Cual es tu edad?

23 respuestas



¿En que tipo de empresa?

10 respuestas



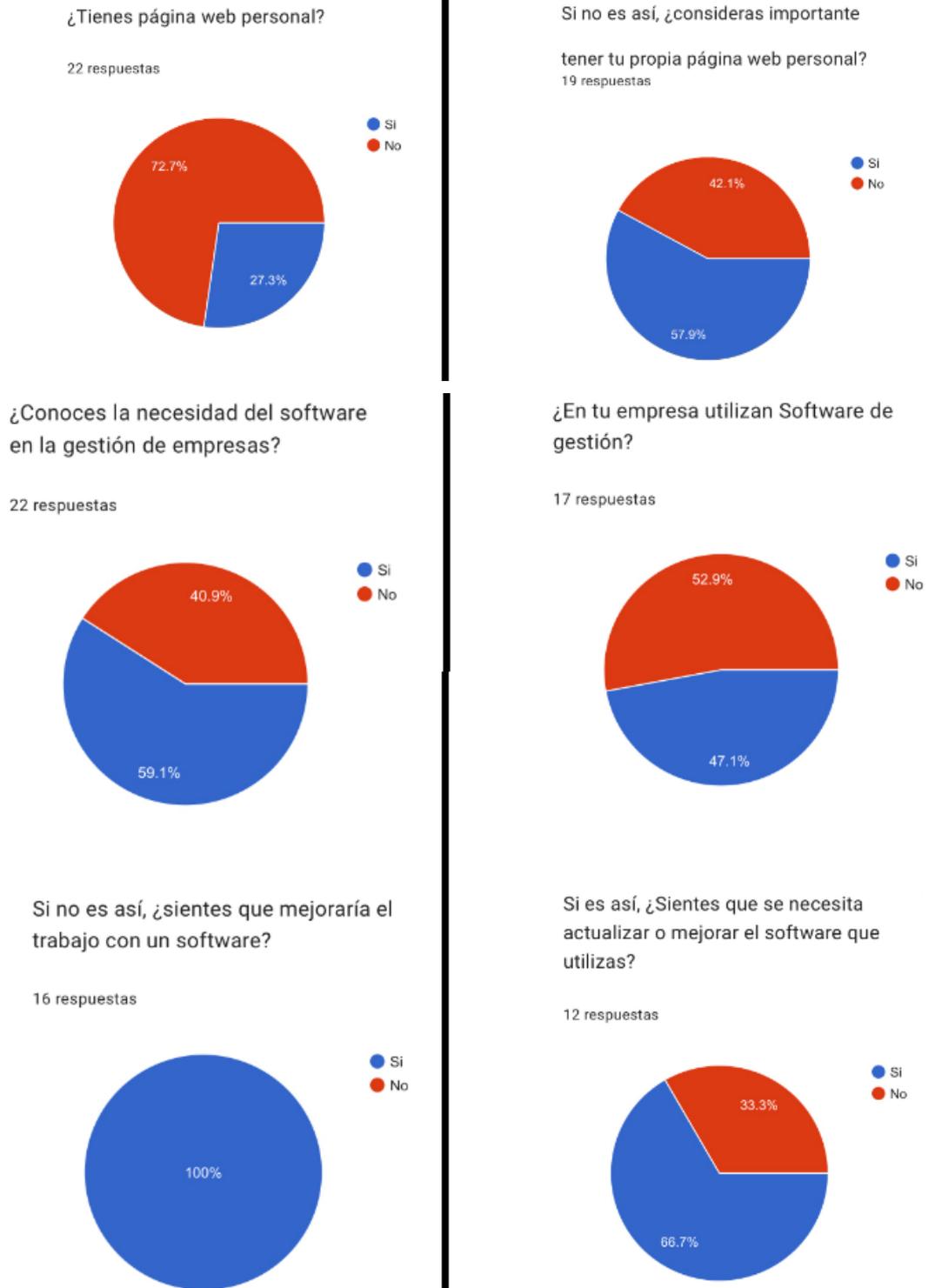


FIG13. Encuestas

1.1.14 Mezcla Comercial

1.1.15 Plan de inversiones

- Alquiler oficina, luz y agua: U\$25.000
- Licencias programas: U\$2000
- Servicio de limpieza oficina: U\$20.000
- Servicio de agua potable a domicilio: U\$1600
- 2 Dispensador de agua: U\$1000
- Cafetera dulce gusto: U\$8000
- Escritorio a medida, 1.5 x 10m: U\$10.000
- Mesa redonda: U\$25.000
- 4 cajoneras: U\$7.000
- 4 sillas de escritorio: U\$8.000
- 6 sillas: U\$6.500
- 4 mini mac: U\$85.000
- 1 Laptop: U\$30.000
- 8 monitores: U\$16.000
- 1 Televisión: U\$20.000
- Internet: U\$1300
- Aire acondicionado: U\$18.000
- Total: U\$143.000

1.1.16 Plan de Recursos

1.1.16.1 Humanos

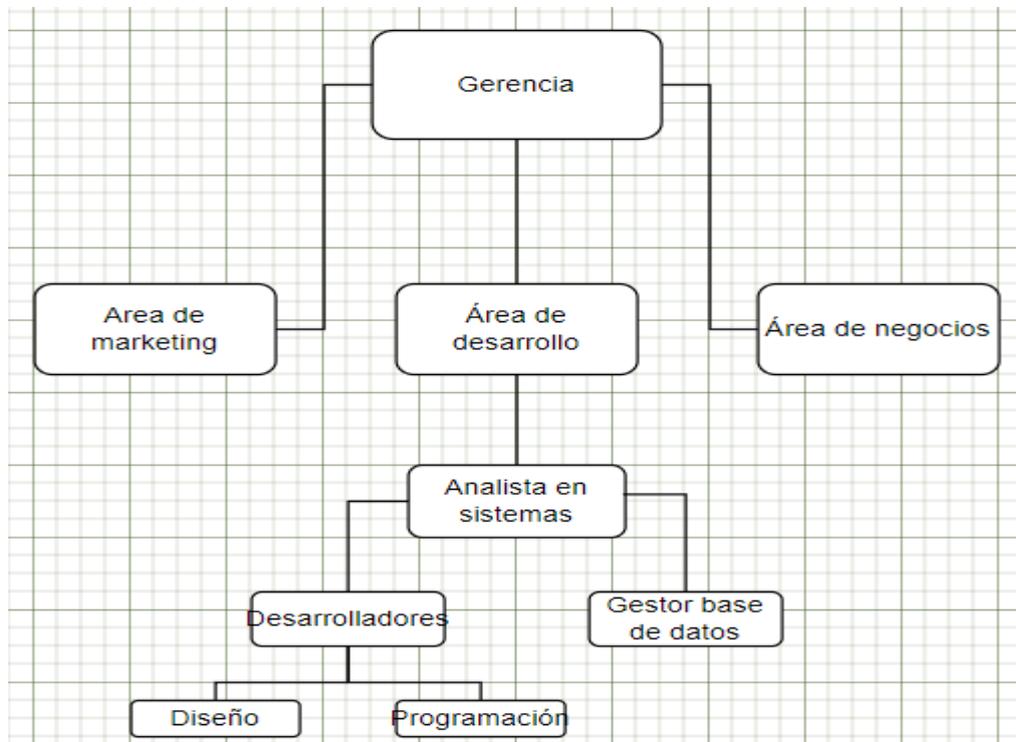


FIG14. Recursos humanos

1.1.16.2 Financieros

	Mayo	Junio	Julio	Agosto	Septiembre	Octubre
Saldo inicial	12.000	23.700	39.700	15.700	14.600	15.400
-Venta producto (6 pagos)	70.000	70.000	70.000	70.000	70.000	70.000
-Mantenimiento	8.000	8.000	8.000	8.000	8.000	8.000
Total ingresos	78.000	78.000	78.000	78.000	78.000	78.000
Egresos:	20.000	20.000	20.000	20.000	20.000	20.000

-Alquiler	6.300	7.000	5.500	7700	8100	6.500
-Servicios (Ose, Ute, Antel)	30.000	30.000	30.000	45.000	45.000	45.000
Retiros socios	10.000	5.000	6.500	6400	4100	4.600
Gastos varios						
Total egresos	66.300	62.000	102.000	79.100	77.200	76.100

Saldo Final	23.700	39.700	15.700	14.600	15.400	17.300
--------------------	--------	--------	--------	--------	--------	--------

2 Anteproyecto

2.1 Introducción.

En este proyecto vamos a realizar un software de gestión para un restaurante, para adquirir conocimiento y experiencia, brindando un buen producto ajustado a las necesidades del cliente.

2.2 Presentación del cliente.

El cliente al que le ofrecemos solución, comienza a forjarse en el año 2012 impulsado por su fundador, Oscar. Dedicó toda su vida a la industria gastronómica, y los últimos 20 años trabajaba en un parador de Punta Del Este. El parador cerró y Oscar se vió motivado a emprender por su camino, el sueño de toda su vida, un Restaurante Boutique de pesca.

Así, invirtiendo los ahorros de su vida, logró abrir un local gastronómico en la Península de Punta del Este. Es un restaurante modesto, con poco flujo y de mucha calidad.

Tiene 2 mozos fijos, 2 cocineros y un chef. Además, 3 mozos zafrales que se unen al equipo en temporada alta. 9 mesas y una cocina.

Gracias a la mente visionaria y emprendedora de su fundador, esta empresa ha salido adelante ante varias dificultades que se les han presentado. Por este motivo, es claramente reconocible que seguirán expandiéndose con los años venideros.

La sucursal está ubicada en la dirección: El trinquete y Juan Días de Solís, esquina. Punta del este, Uruguay.

El horario del local es: 12-16hs / 18-24hs.

Todos sus registros y sistemas de comandas están en papel, no digitalizados. El cliente busca digitalizar su negocio.

Nuestro cliente busca ofrecer una experiencia más moderna a sus clientes mediante el uso de menús digitales y otros cambios que están a verse, también busca ofrecer un sistema de delivery eficaz que ahorre la mayor cantidad de problemas posibles mediante la digitalización, con esta misma premisa se busca automatizar los procesos que sean convenientes, por ejemplo la manera en la que llega las órdenes a la cocina, tratando de ahorrar la mayor cantidad de tiempo y evitar problemas de interpretación entre cocina y los mozos. Para controlar el stock de productos se necesita llevar registro de las entregas periódicas de los proveedores.

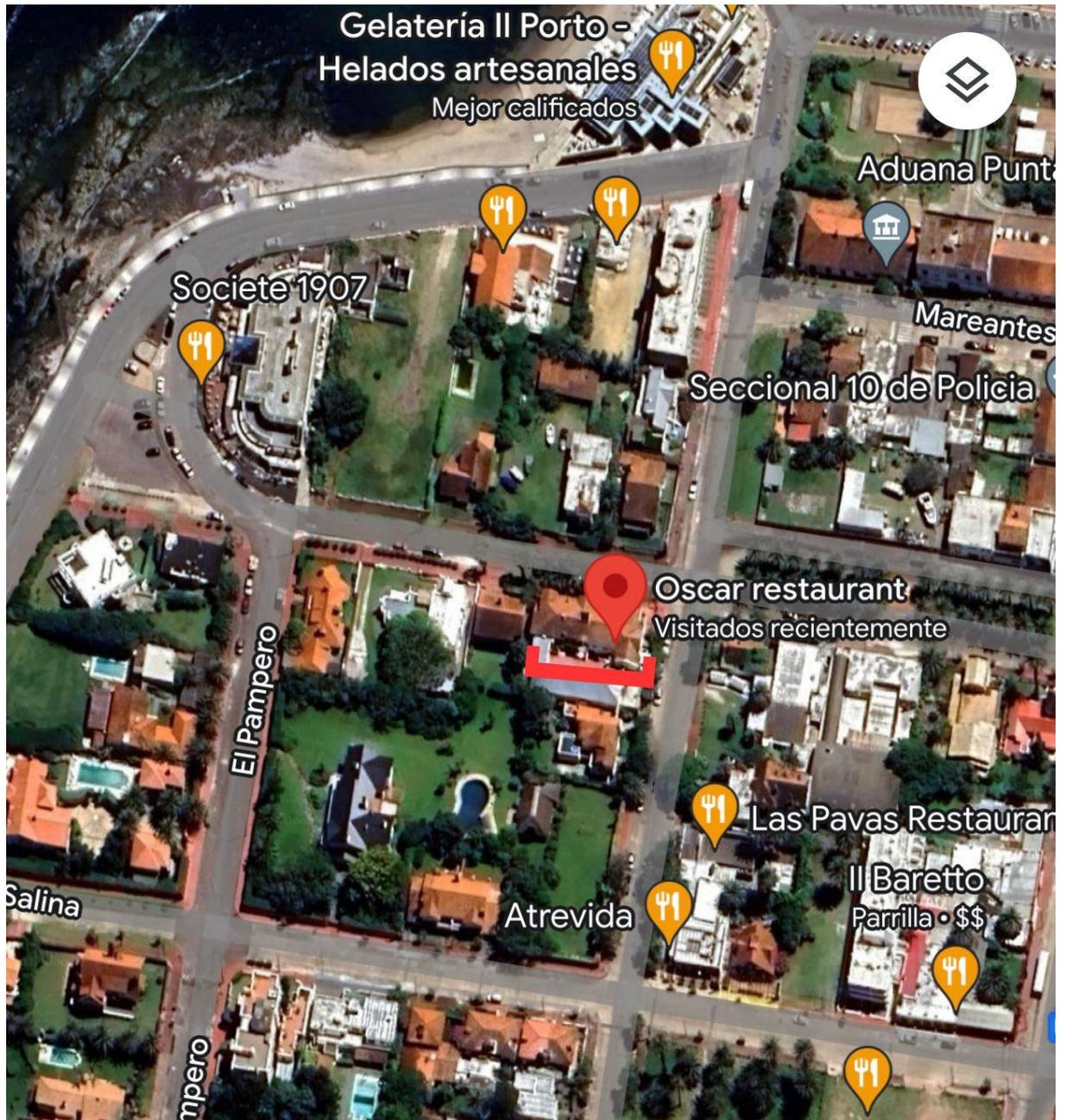


FIG15. Ubicación local

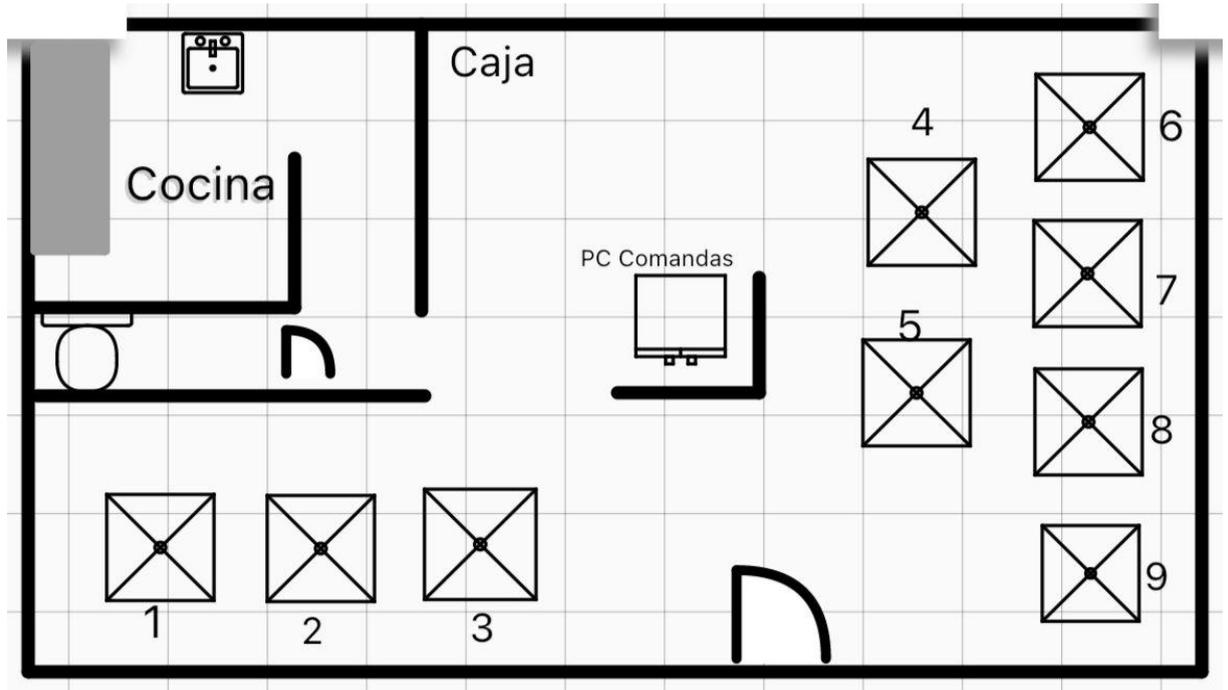


FIG16. Plano local

MENÚ

ENTRADAS Y ENSALADAS		POSTRES:	
Ensalada Mediterranea.....	\$ 850	Flan de naranjas.....	\$ 400
Ensalada de Salmón ahumada	\$ 850	Mousse de chocolate Belga.....	\$ 450
Rabas.....	\$ 750	Panqueque de manzanas.....	\$ 450
Chipirones a la leonesa.....	\$ 750		
Camarones al ajillo.....	\$ 950	BEBIDAS:	
Jamon crudo con melón.....	\$ 750	Agua mineral/Refrescos.....	\$ 200
PRINCIPALES:		VINOS:	
Brotola grille.....	\$ 850	Las Moras Sauvignon Blanc.....	\$ 700
Salmóm grille.....	\$ 950	Trapiche Reserva Malbec.....	
Lomo.....	\$ 950	o Chardonnay.....	\$ 1100
		Prima Donna Tannat , Merlot.....	
SALSAS PARA PESCADOS Y CARNES		o Sauvignon Blanc.....	\$ 950
De Hongos.....	\$ 180	CAFÉ ILLY.....	\$ 200
Roquefort.....	\$ 180	FREIXENET C. NEGRO.....	\$ 1400
Manteca negra c/alcaparras.....	\$ 180	FREIXENET C. NEVADA.....	\$ 1200
Oscar.....	\$ 280		
Sorrentinos de jamon y queso con salsa Basilisco.....			\$ 850
Acompañamientos: Papines al romero, puré de colabaza, verduras grilladas, papas fritas			

2.3.1 Recepción del cliente en el local

2.3.1.1 Descripción

El cliente es atendido por un mozo al ingresar y elige una mesa, o va a la asignada en caso de tener reserva. Si es takeaway, pasa directo por caja.

2.3.1.2 Diagrama de actividad

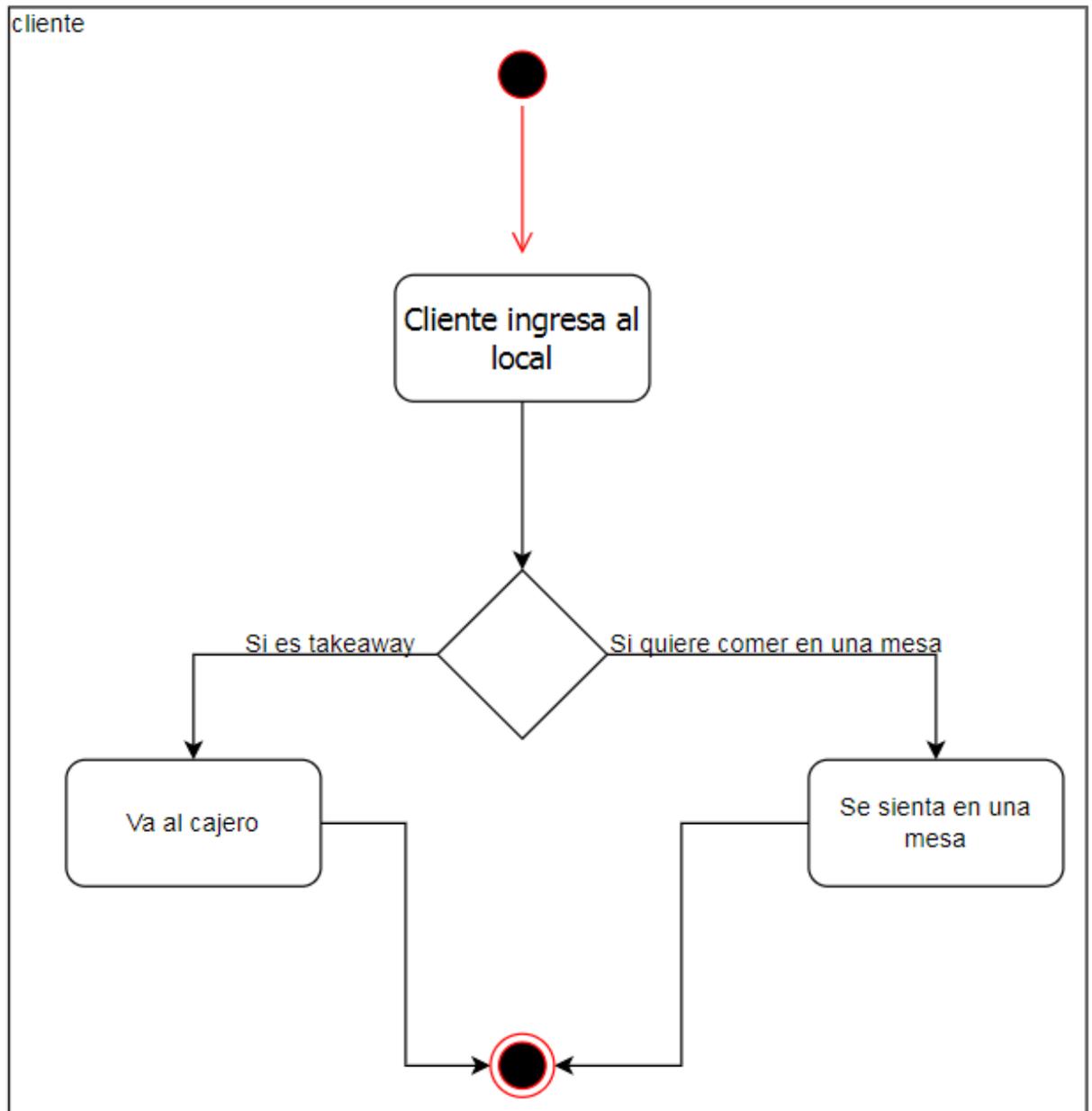


FIG19. Proceso 01

2.3.2 Atención al cliente

2.3.2.1 Descripción

El cliente es atendido por un mozo al ingresar y elige una mesa, o va a la asignada en caso de tener reserva. Si es takeaway, pasa directo por caja.

2.3.2.2 Diagrama de actividad

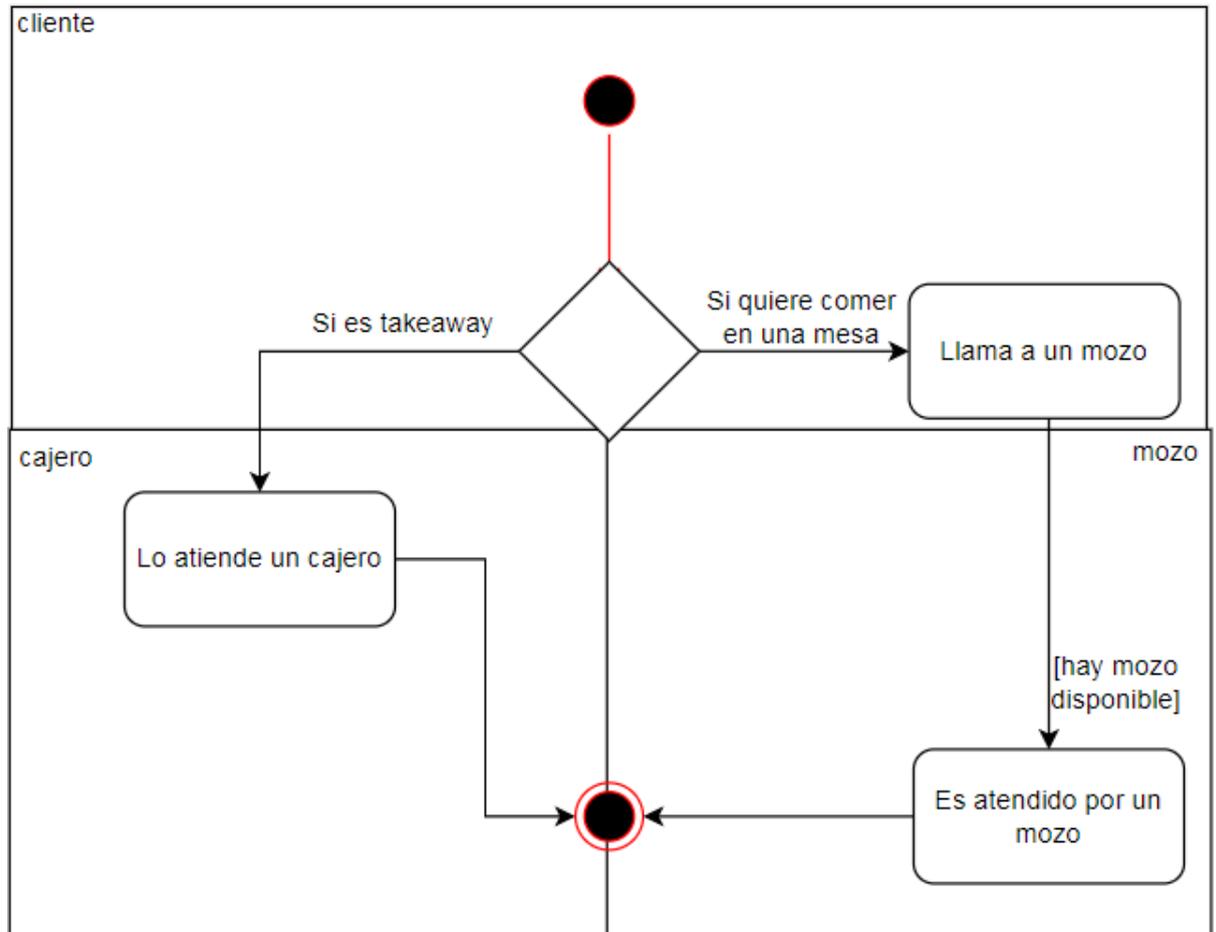


FIG20. Proceso 02

2.3.3 Registro de orden

2.3.3.1 Descripción

Mozo registra la mesa en la que el cliente está y también la orden, en caso de ser takeaway solo se registra la orden.

2.3.3.2 Diagrama de actividad

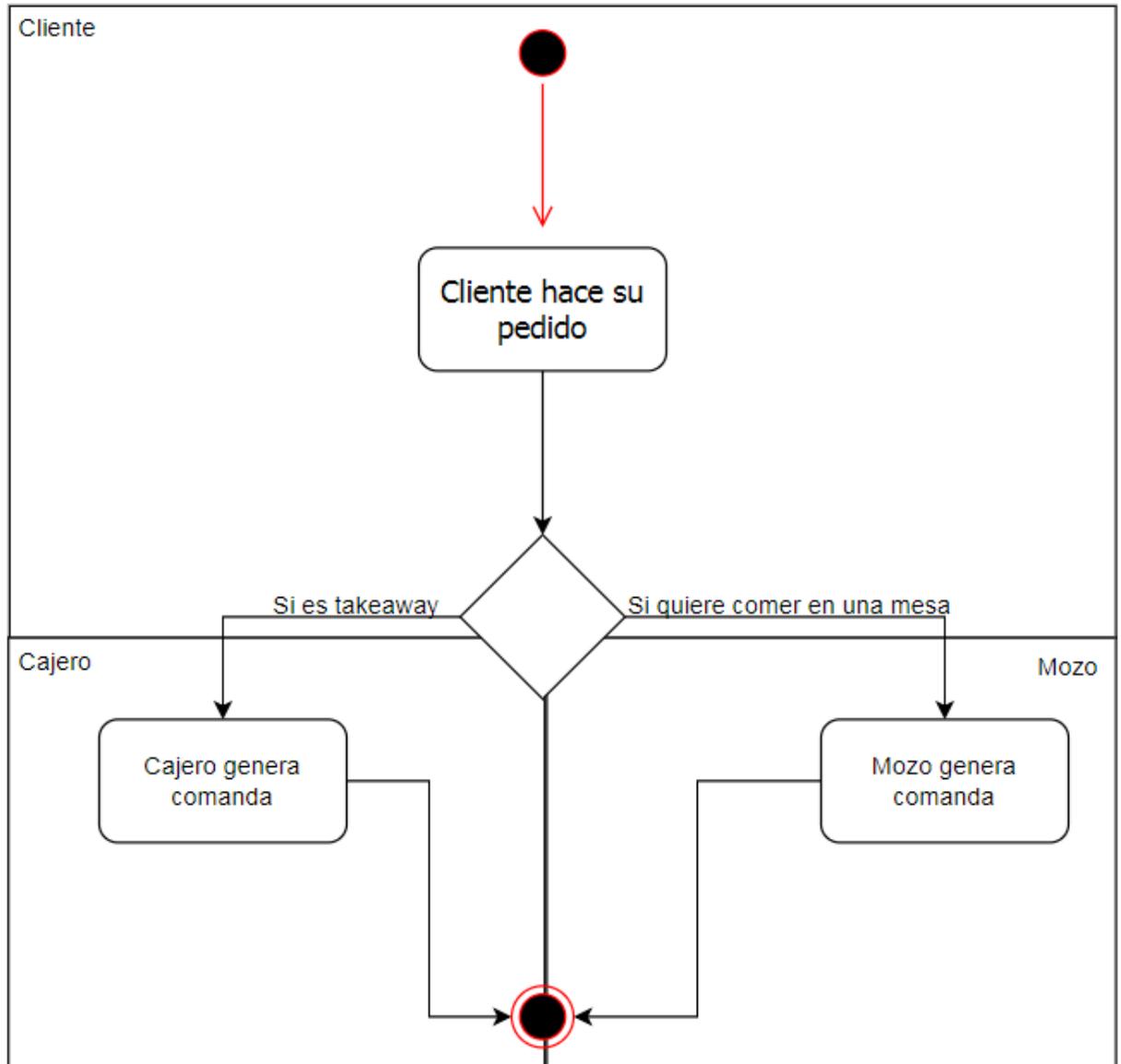


FIG21. Proceso 03

2.3.4 Envío de orden a la cocina

2.3.4.1 Descripción

El mozo o el cajero entrega orden a la cocina en formato papel.

2.3.4.2 Diagrama de actividad

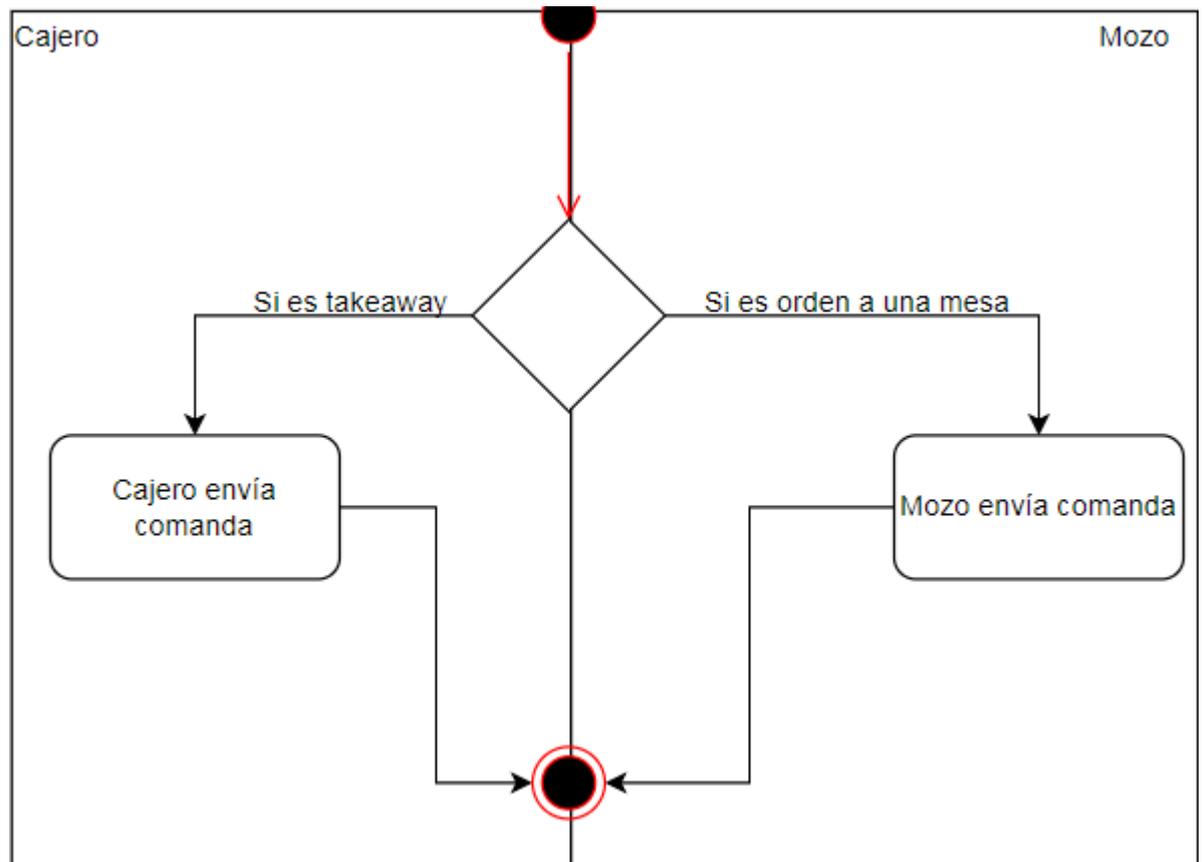


FIG22. Proceso 04

2.3.5 Elaboración de orden

2.3.5.1 Descripción

El chef recibe qué platos tiene que realizar y se fija si tiene el stock suficiente de productos para realizarlos, si tiene lo suficiente prepara los platos, si no manda a realizar un re stock y si no es posible esto le pide al cliente que cambie la orden.

2.3.5.2 Diagrama de actividad

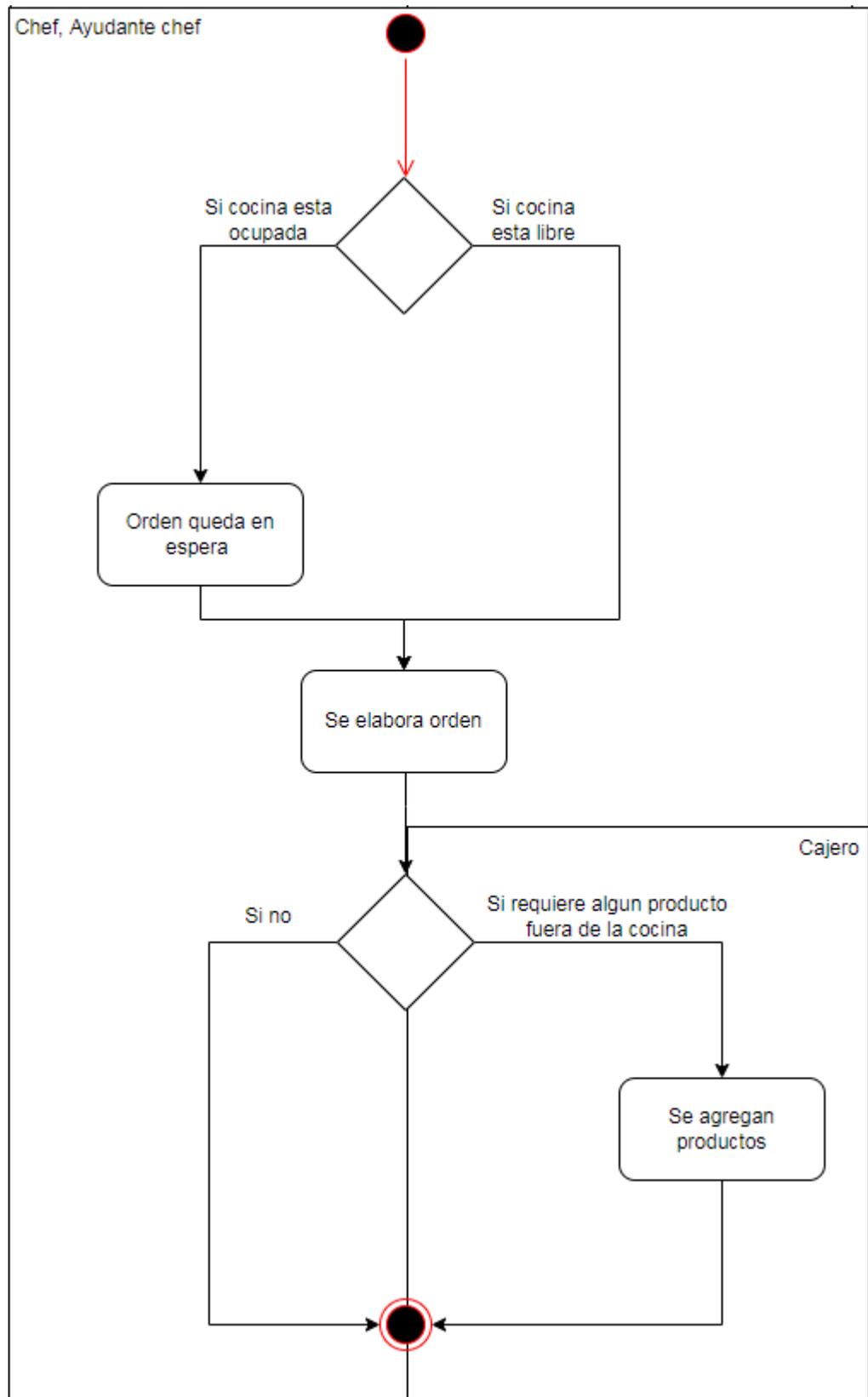


FIG23. Proceso 05

2.3.6 Entrega de orden

2.3.6.1 Descripción

El mozo lleva la orden a la mesa, el cajero entrega la orden en el mostrador.

2.3.6.2 Diagrama de actividad

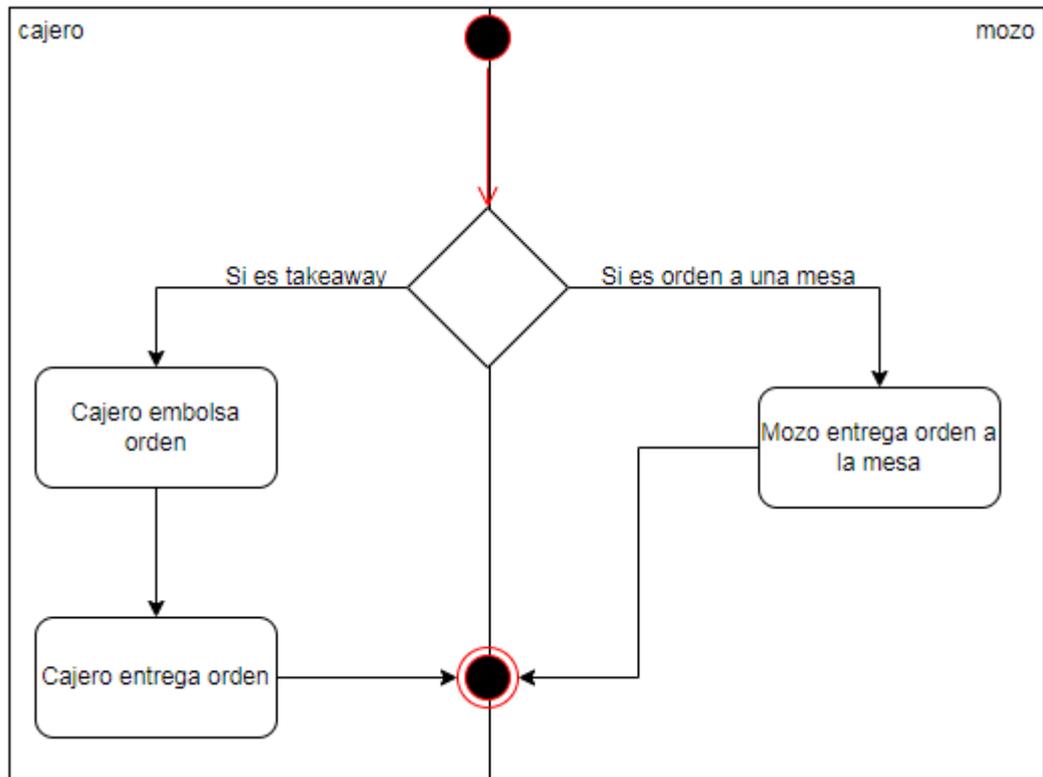


FIG24. Proceso 06

2.3.7 Ingreso de stock

2.3.7.1 Descripción

Empleado ingresa stock a las almacenes del restaurante.

2.3.7.2 Diagrama de actividad

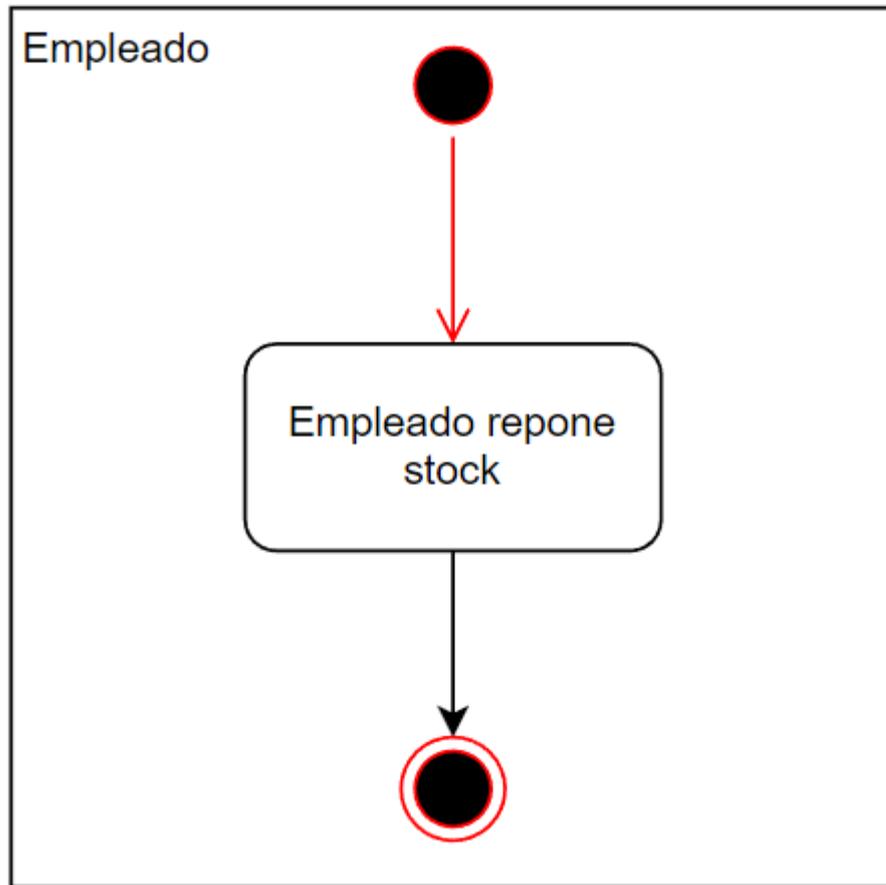


FIG25. Proceso 07

2.3.8 Pago de orden

2.3.8.1 Descripción

El mozo se dirige a la mesa del cliente y le pregunta el método de pago y el le cobra lo consumido generando un ticket

2.3.8.2 Diagrama de actividad

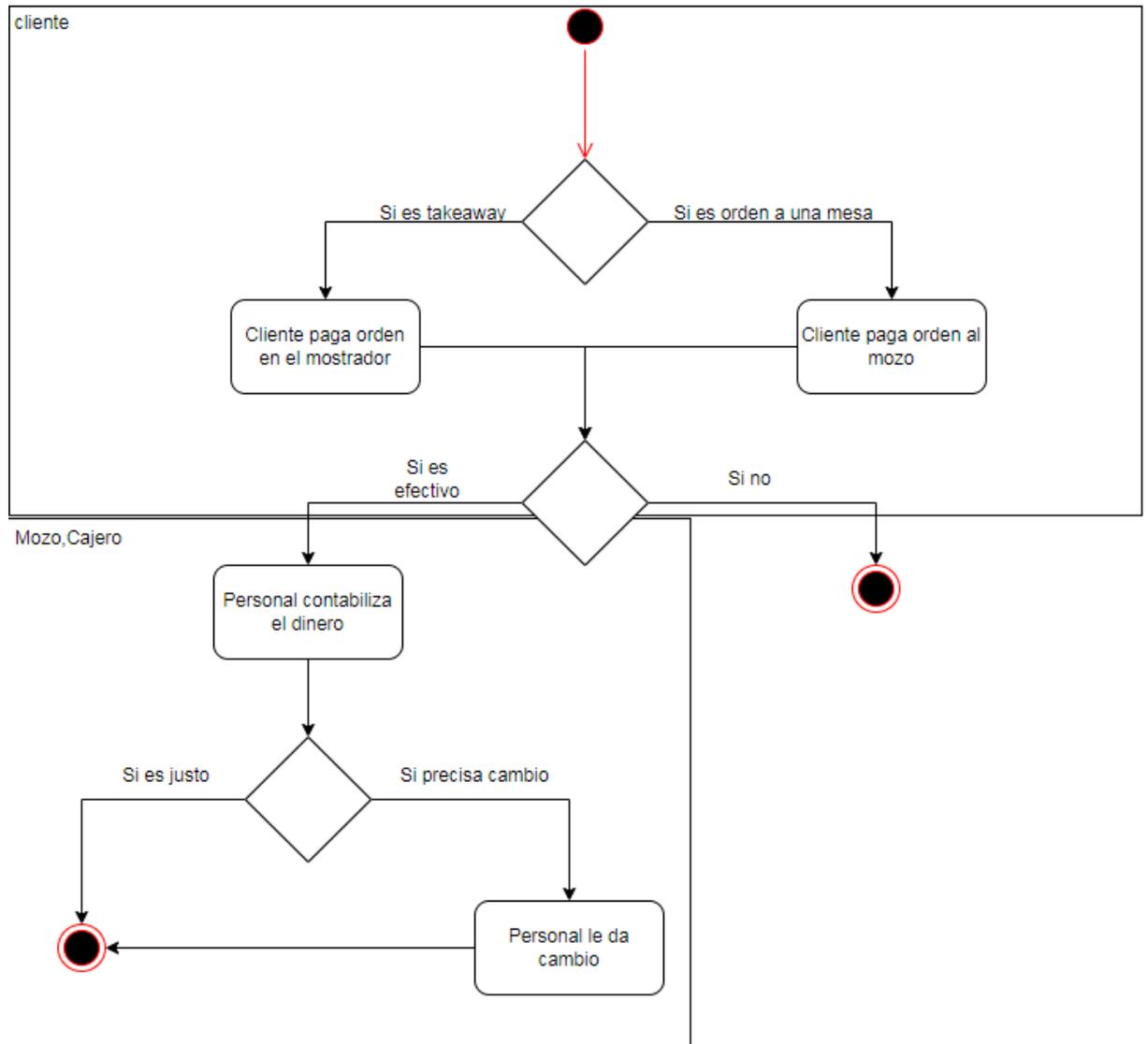


FIG26. Proceso 08

2.3.9 Generar ticket

2.3.9.1 Descripción

Quando el pago es registrado un ticket se genera e imprime registrando el precio total, los platos que pidió, el nombre del restaurante, fecha y impuestos

2.3.9.2 Diagrama de actividad

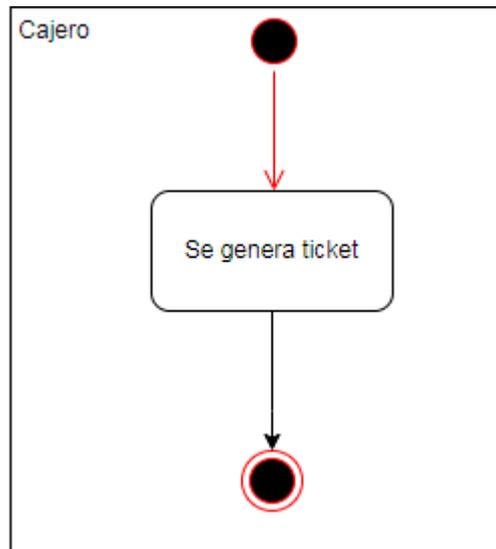


FIG27. Proceso 09

2.3.10 Registro de stock

2.3.10.1 Descripción

Gerente toma registro de los productos ingresados al stock.

2.3.10.2 Diagrama de actividad

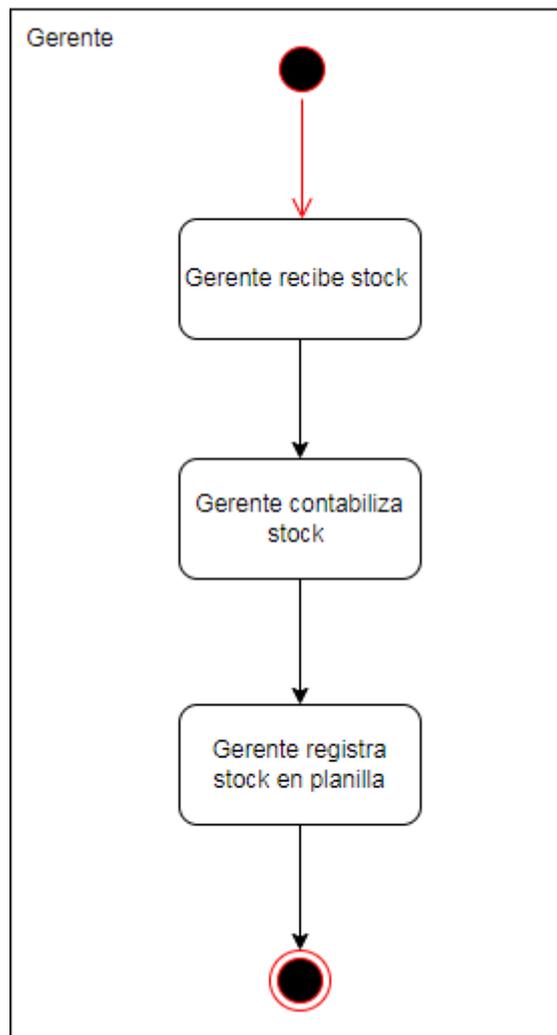


FIG28. Proceso 10

2.3.11 Modificación del menú

2.3.11.1 Descripción

El gerente en conjunto con el jefe de cocina se encarga de ajustar los precios y cambiar los platos del menú.

2.3.11.2 Diagrama de actividad

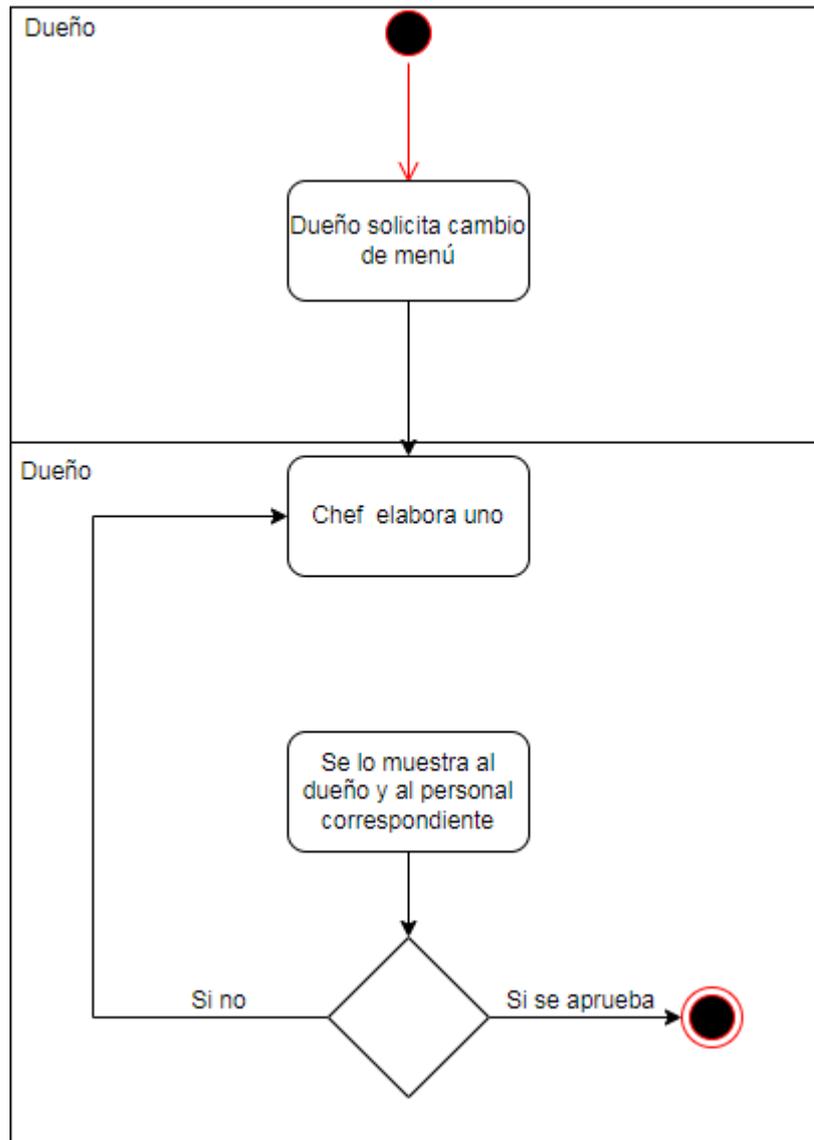


FIG29. Proceso 11

2.4 Actores involucrados

- Mozo
- Cliente
- Chef
- Gerente
- Cajero

2.4.1 Diagrama de actores

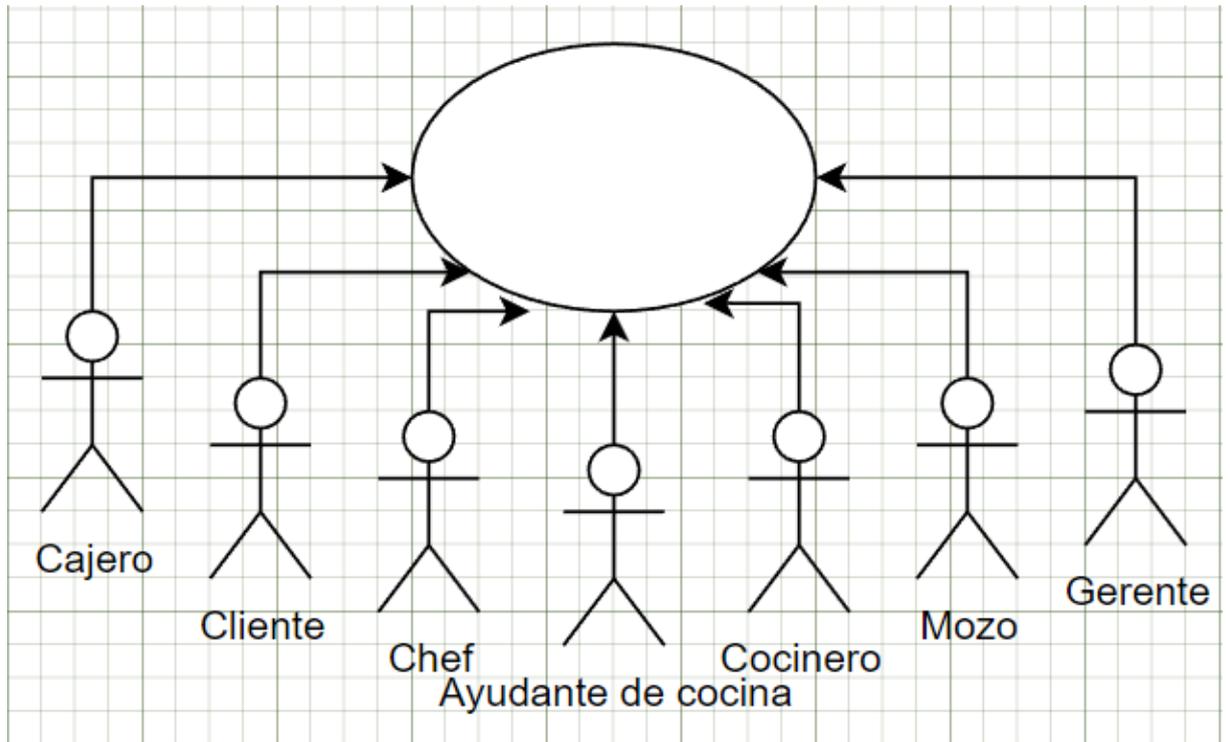


FIG30. Diagrama de actores

2.4.2 Descripción

- **Mozo**

Encargado de atender a los clientes del restaurante proporcionándoles alimentos, bebidas y asistencia durante la estancia.

- **Cliente**

El cliente es la persona o empresa receptora del servicio del Restaurante a cambio de dinero.

- **Chef(Jefe de cocina)**

Jefe de cocina del restaurante, a cargo de los cocineros, encargado de modificar el menú.

- **Gerente**

Persona que por oficio se encarga de dirigir, gestionar o administrar una empresa, en este caso el restaurante.

- **Cajero**

Persona que tiene por oficio llevar el control de caja y atender los pagos y cobros.

2.5 Descripción del entorno

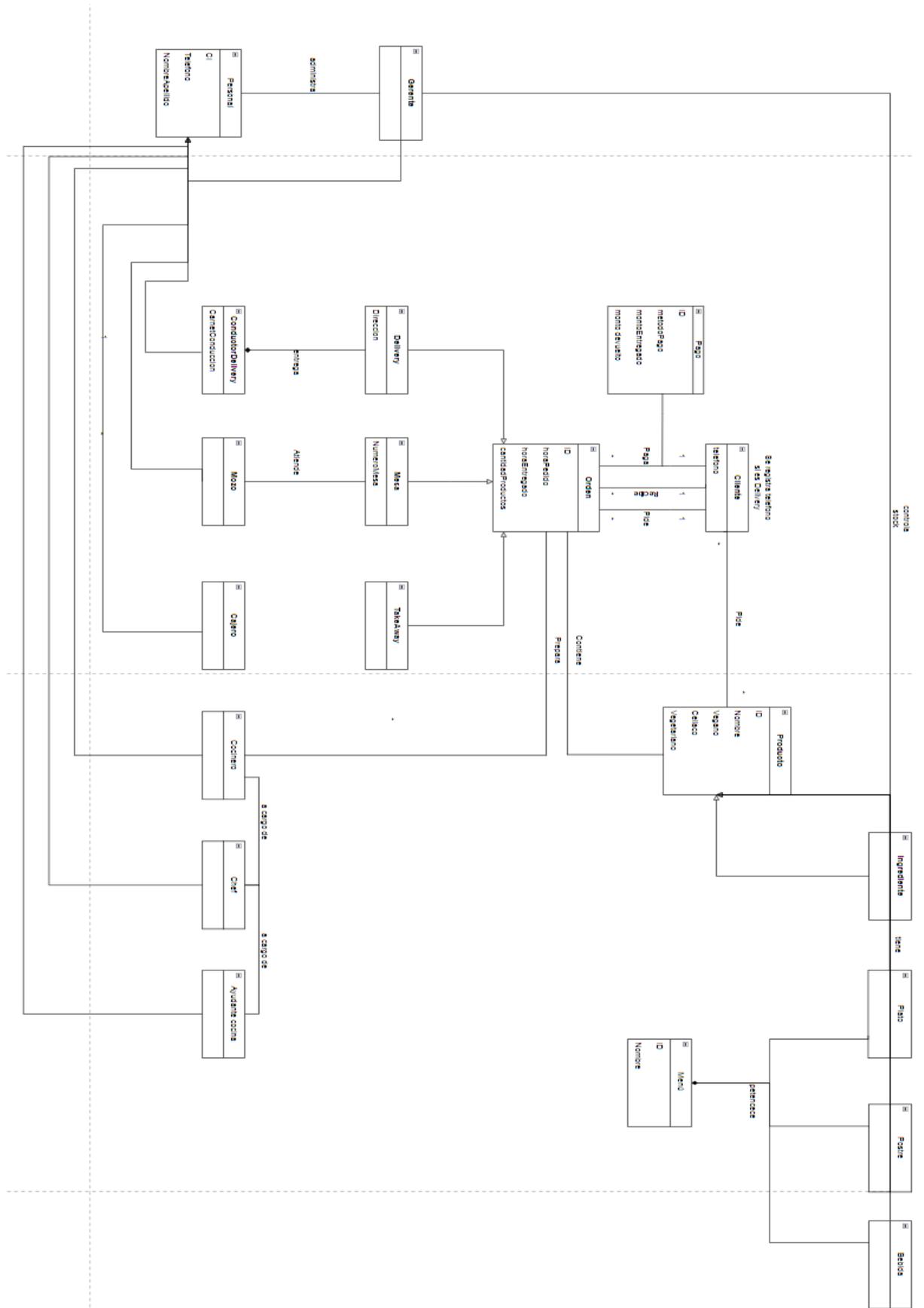


FIG31. UML

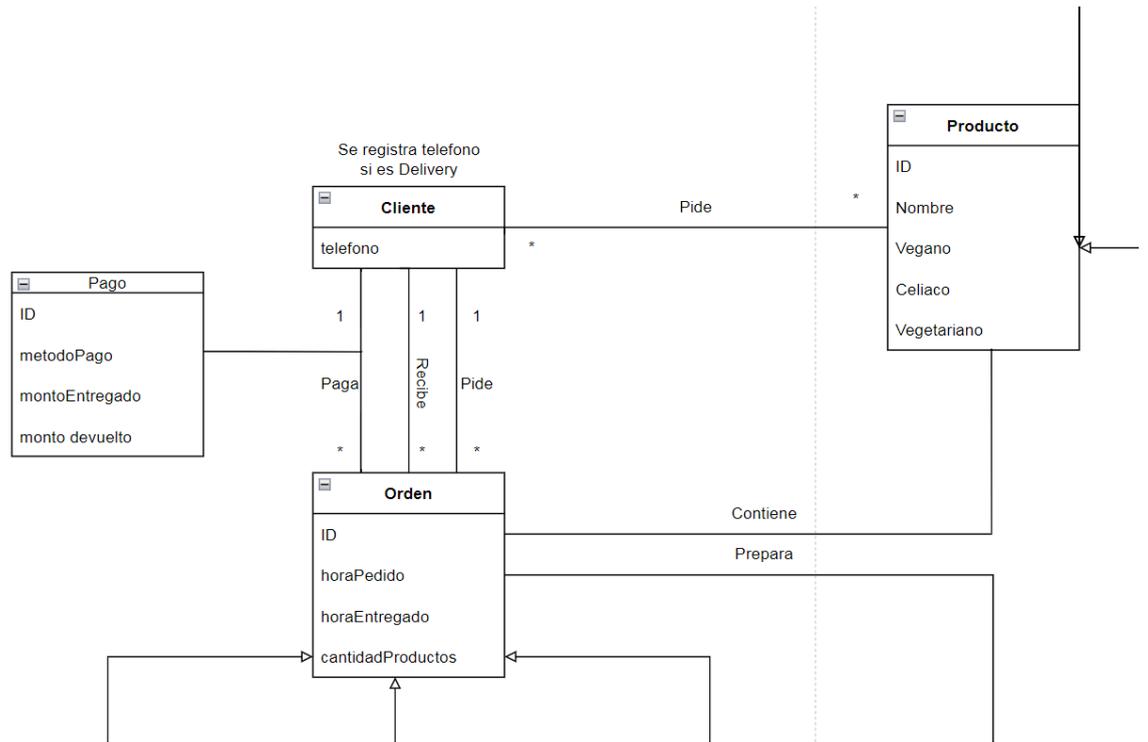


FIG32. UML 2

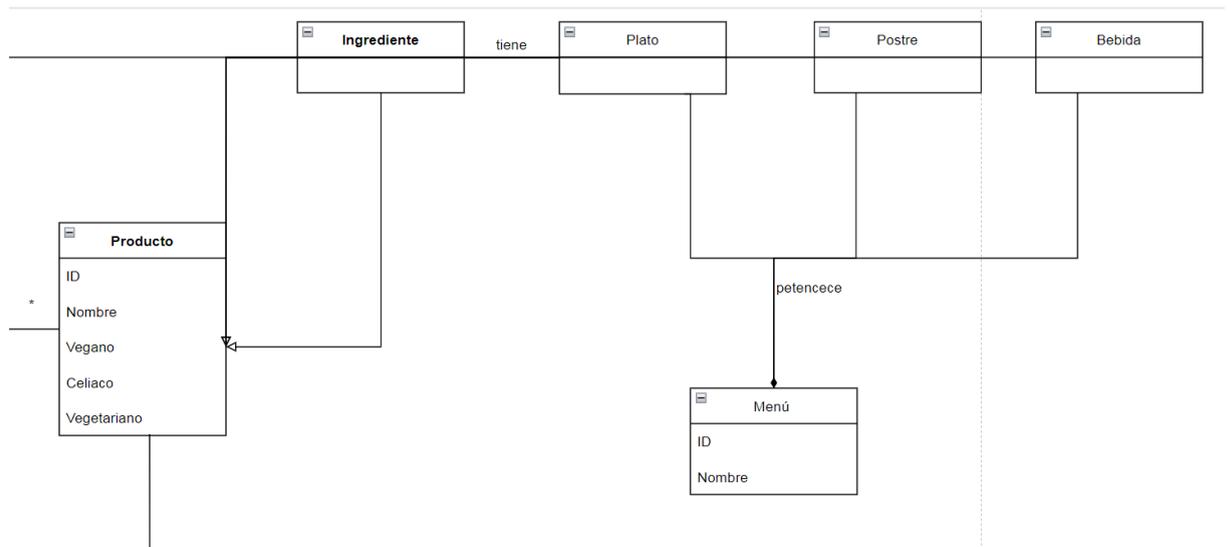


FIG33. UML 3

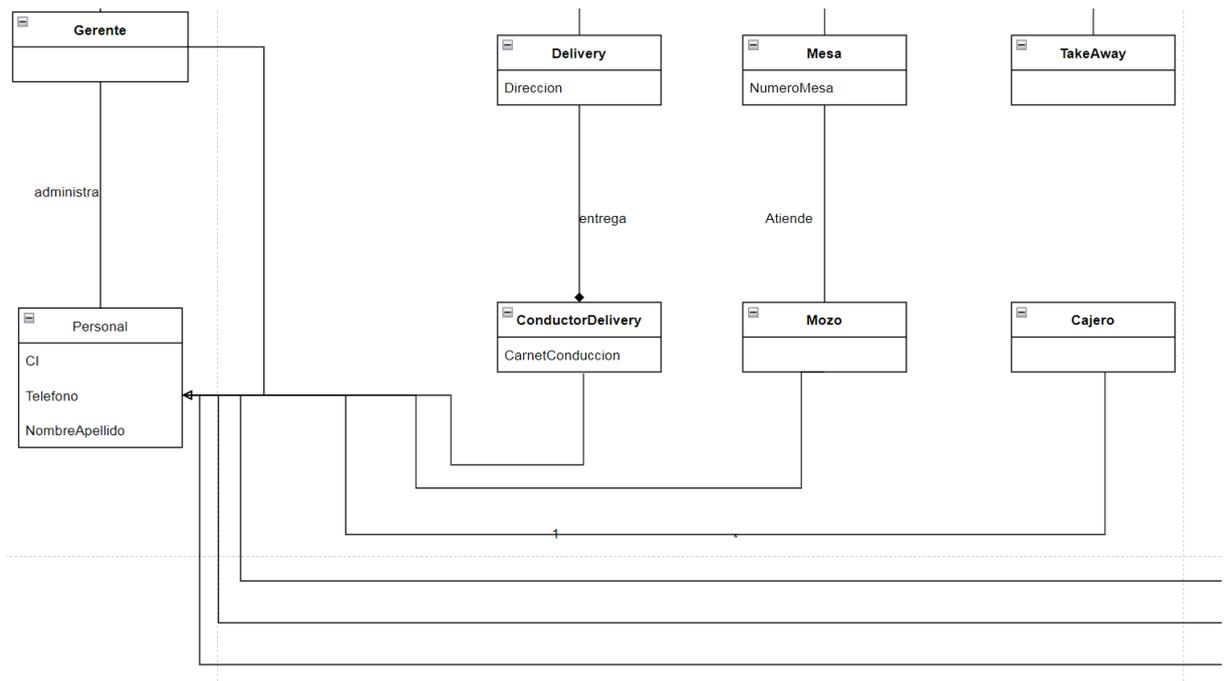


FIG34. UML 4

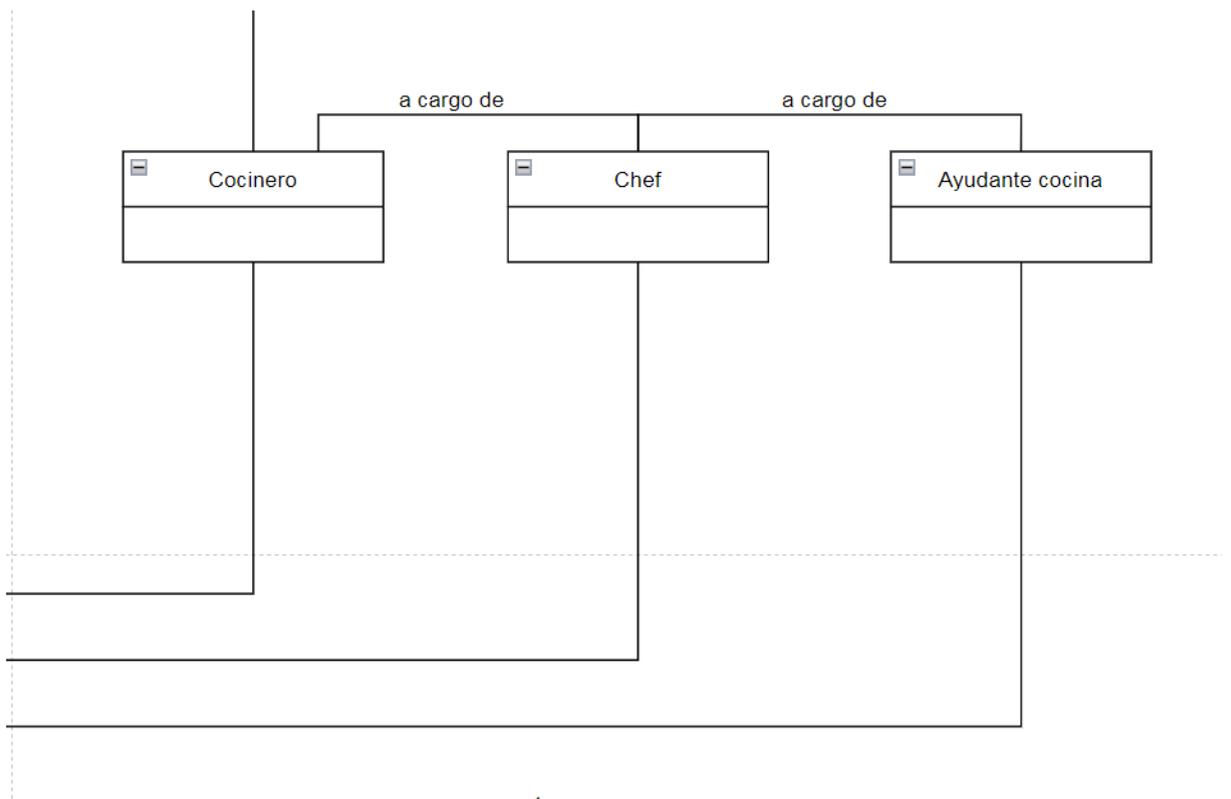


FIG35. UML 5

2.6 Lista de necesidades.

- NEC1.** Registro de comandas
- NEC2.** Registro de ventas
- NEC3.** Acceso fácil al menú
- NEC4.** Envío ágil y correcto de comandas a la cocina
- NEC5.** Control de acceso a la información

- NEC6.** Control de mesas
- NEC7.** Sistema de reservas
- NEC8.** Sistema de comandas
- NEC9.** Actualización menú actual
- NEC10.** Control de stock

2.7 Objetivos

- OB1.** Informatizar gestión de comandas.
- OB2.** Registrar ventas automáticamente.
- OB3.** Generar estadísticas en base a las ventas.
- OB4.** Digitalizar menú.
- OB5.** Crear un control de usuarios seguro mediante login.
- OB6.** Implementar abstracción y encapsulamiento de la información.
- OB7.** Crear tipos de usuarios con acceso a distintas funcionalidades del sistema.
- OB8.** Crear plano de las mesas que indique en qué estado están, (libre, ocupada, en espera de ser atendida).
- OB9.** Crear sistema de reservas.
- OB10.** Crear sistema de stock.

2.7.1 Matriz de trazabilidad de objetivos y necesidades

	NEC 1	NEC 2	NEC 3	NEC 4	NEC 5	NEC 6	NEC 7	NEC 8	NEC 9	NEC 10
OB1	x							x		
OB2		x						x		
OB3		x						x		
OB4			x						x	
OB5					x			x		
OB6					x			x		
OB7					x					
OB8	x					x				
OB9							x	x		
OB10										x

2.8 Lista de requerimientos

2.8.1 Actores del Sistema

Administrador: Podrá designar usuarios de tipo gerente. Además, contará con el poder total, y todos los privilegios y permisos del sistema.

Este tipo de usuario está designado para el dueño de la empresa y el personal de alto cargo que él mismo desee designar, como los encargados.

Usuario estándar:

- Gerente: Alta y baja de empleados .
- Mozo: Con el permiso de elaborar comandas, acceder al menú, acceder y modificar el estado de las mesas.
- Cajero: Con el permiso de elaborar comandas, cobrar cuenta.

2.8.2 Requerimientos funcionales

- RF1. Comprobar orden
- RF2. Alta orden
- RF3. Baja orden
- RF4. Modificar orden
- RF5. Listar orden
- RF6. Buscar orden
- RF7. Comprobar mesa
- RF8. Alta mesa
- RF9. Baja mesa
- RF10. Modificar mesa
- RF11. Listar mesa
- RF12. Buscar mesa
- RF13. Alta cliente
- RF14. Baja cliente
- RF15. Modificar cliente
- RF16. Listar cliente
- RF17. Buscar cliente
- RF18. Comprobar cliente
- RF19. Alta personal
- RF20. Baja personal
- RF21. Modificar personal
- RF22. Listar personal
- RF23. Comprobar personal
- RF24. Buscar personal
- RF25. Comprobar Pago
- RF26. Alta pago
- RF27. Modificar pago
- RF28. Baja pago
- RF29. Listar pago
- RF30. Buscar pago
- RF31. Comprobar producto

- RF32.** Alta producto
- RF33.** Baja producto
- RF34.** Modificar producto
- RF35.** Listar producto
- RF36.** Buscar Producto
- RF37.** Alta menú
- RF38.** Baja menú
- RF39.** Listar menú
- RF40.** Modificar menú
- RF41.** Comprobar menú
- RF42.** Buscar menú
- RF43.** Alta reserva
- RF44.** Baja reserva
- RF45.** Modificar reserva
- RF46.** Listar reserva
- RF47.** Comprobar reserva
- RF48.** Buscar reserva
- RF49.** Comprobar usuarios
- RF50.** Organizar órdenes
- RF51.** Alta estación
- RF52.** Baja estación
- RF53.** Modificar estación
- RF54.** Listar estación
- RF55.** Comprobar estación
- RF56.** Buscar estación
- RF57.** Alta metodo Pago
- RF58.** Baja metodo Pago
- RF59.** Modificar metodo Pago
- RF60.** Listar método Pago
- RF61.** Comprobar método Pago
- RF62.** Buscar metodo Pago
- RF63.** Alta usuario del sistema
- RF64.** Baja usuario del sistema
- RF65.** Modificar usuario del sistema
- RF66.** Listar usuarios del sistema
- RF67.** Comprobar usuario del sistema
- RF68.** Buscar usuario del sistema
- RF69.** Inicio de sesión
- RF70.** Ordenar productos por precio
- RF71.** Categorizar productos por menú
- RF72.** Categorizar productos por tipo
- RF73.** Categorizar productos por aptitud (celíaco, vegano, vegetariano)
- RF74.** Gestionar stock
- RF75.** Cerrar sesión
- RF76.** ver menú
- RF77.** Generar estadísticas en base a las ventas

RF1. Comprobar orden

El sistema permitirá comprobar el estado de las órdenes
Actores participantes: administrador, Gerente

RF2. Alta orden

El sistema permitirá registrar los datos de las órdenes.
De las cuales se almacenarán:

- Hora del pedido
- Hora de entrega
- Productos

Actores participantes: Gerente, Administrador, Mozo, Cajero.

RF3. Baja orden

El sistema permitirá dar de baja el registro de las órdenes y sus atributos.

Actores participantes: Gerente, Administrador, Mozo, Cajero.

RF4. Modificar orden

El sistema permitirá modificar los datos de las órdenes. Actores participantes: Gerente, Administrador, Mozo, Cajero.

RF5. Listar orden

El sistema permitirá listar los datos de las órdenes .

Actores participantes: Gerente, Administrador, Mozo, Cajero.

RF6. Buscar orden

El sistema permitirá buscar los datos de las órdenes .

Actores participantes: Gerente, Administrador, Mozo, Cajero.

RF7. Comprobar mesa

El sistema permitirá comprobar el estado de las mesas

Actores participantes: Gerente, Administrador, Mozo, Cajero.

RF8. Alta mesa

El sistema permitirá registrar los datos de las mesas.

De los cuales se almacenarán:

- Número
- Estado

Actores participantes: Gerente, Administrador, Mozo, Cajero.

RF9. Baja mesa

El sistema permitirá dar de baja el registro de las mesas y sus atributos.

Actores participantes: Gerente, Administrador, Mozo, Cajero.

RF10. Modificar mesa

El sistema permitirá modificar los datos de las mesas.

Actores participantes: Gerente, Administrador, Mozo, Cajero.

RF11. Listar mesa

El sistema permitirá listar los datos de los pedidos de mesas

Actores participantes: Gerente, Administrador, Mozo, Cajero.

RF12. Buscar mesa

El sistema permitirá listar los datos de los pedidos de mesa

Actores participantes: Gerente, Administrador, Mozo, Cajero.

RF13. Alta cliente

El sistema permitirá registrar los datos de los clientes.

De los cuales se almacenarán:

- Teléfono(Si es delivery)

Actores participantes: Gerente, Administrador, Cajero.

RF14. Baja cliente

El sistema permitirá dar de baja el registro de los clientes y sus atributos.

Actores participantes: Gerente, Administrador, Cajero.

RF15. Modificar cliente

El sistema permitirá modificar los datos de los clientes.

Actores participantes: Gerente, Administrador, Cajero.

RF16. Listar cliente

El sistema permitirá listar los datos de los clientes.

Actores participantes: Gerente, Administrador, Cajero.

RF17. Comprobar cliente

El sistema permitirá comprobar los datos de los clientes.

Actores participantes: Gerente, Administrador, Cajero.

RF18. Buscar cliente

El sistema permitirá listar los datos de los clientes.

Actores participantes: Gerente, Administrador, Cajero.

RF19. Alta personal

El sistema permitirá registrar los datos de los empleados.

De los cuales se almacenarán:

- Cédula
- Nombre
- Apellido
- Teléfono

Actores participantes: administrador, Gerente.

RF20. Baja personal

El sistema permitirá dar de baja al personal y sus atributos.

Actores participantes: administrador, Gerente.

RF21. Modificar personal

El sistema permitirá modificar los datos del personal.

Actores participantes: administrador, Gerente.

RF22. Listar personal

El sistema permitirá listar los datos del personal.

Actores participantes: Gerente.

RF23. Comprobar personal

El sistema permitirá comprobar datos del personal.

Actores participantes: administrador, Gerente.

RF24. Buscar personal

El sistema permitirá buscar los datos del personal.

Actores participantes: Gerente.

RF25. Comprobar Pago

El sistema permitirá comprobar información del pago.

Actores participantes: gerente, gerente, Cajero.

RF26. Alta pago

El sistema permitirá registrar los datos de los métodos de pago.

De los cuales se almacenarán:

- Método de pago
- Monto entregado
- Monto devuelto

Actores participantes: administrador, gerente, Cajero, Mozo.

RF27. Baja pago

El sistema permitirá dar de baja los pagos y sus atributos.

Actores participantes: administrador, gerente, Cajero.

RF28. Modificar pago

El sistema permitirá modificar los datos de los pagos.

Actores participantes: administrador, gerente.

RF29. Listar pago

El sistema permitirá listar los datos de los pagos.

Actores participantes: administrador, gerente, Cajero.

RF30. Buscar pago

El sistema permitirá buscar los datos de los pagos.

Actores participantes: administrador, gerente, Cajero.

RF31. Comprobar producto

El sistema permitirá comprobar disponibilidad de los productos

Actores participantes: administrador, gerente, Cajero.

RF32. Alta producto

El sistema permitirá registrar los datos de los productos.

De los cuales se almacenarán:

- Nombre
- Precio
- Apto celiaco
- Apto vegano
- Apto vegetariano

Actores participantes: administrador, gerente.

RF33. Baja producto

El sistema permitirá dar de baja los platos y bebidas y sus atributos.

Actores participantes: administrador, gerente.

RF34. Modificar producto

El sistema permitirá modificar los datos de los platos y bebidas.

Actores participantes: administrador, gerente.

RF35. Listar producto

El sistema permitirá listar los datos de los platos y bebidas.

Actores participantes: administrador, gerente.

RF36. Buscar producto

El sistema permitirá buscar los datos de los platos y bebidas.

Actores participantes: administrador, gerente.

RF37. Alta menú

El sistema permitirá registrar los datos de los menús.

De los cuales se almacenará:

- Nombre

Actores participantes: administrador, gerente.

RF38. Baja menú

El sistema permitirá dar de baja los menús y sus atributos.

Actores participantes: administrador, gerente.

RF39. Modificar menú

El sistema permitirá modificar los datos de los empleados.

Actores participantes: administrador, gerente.

RF40. Listar menú

El sistema permitirá listar los datos de los menús.

Actores participantes: administrador, gerente, Mozo, Cajero.

RF41. Buscar menú

El sistema permitirá buscar los datos del menú.

Actores participantes: administrador, gerente, Cajero.

RF42. Comprobar menú

El sistema permitirá comprobar el menú.

Actores participantes: administrador, gerente, Cajero.

RF43. Alta reserva

El sistema permitirá registrar los datos de los gerentes.

De los cuales se almacenará:

- cedula
- id
- teléfono Cliente

Actores participantes: administrador, Gerente, Cajero.

RF44. Baja reserva

El sistema permitirá dar de baja reservas.

Actores participantes: administrador, Gerente, Cajero.

RF45. Modificar reserva

El sistema permitirá modificar los datos de las reservas.

Actores participantes: administrador, Gerente, Cajero.

RF46. Listar reserva

El sistema permitirá listar los datos de las reservas

Actores participantes: administrador, Gerente, Cajero.

RF47. Buscar reserva

El sistema permitirá buscar los datos de las reservas.

Actores participantes: administrador, gerente, Cajero.

RF48. Comprobar reserva

El sistema permitirá comprobar el estado de las reservas.

Actores participantes: administrador, gerente, Cajero.

RF49. Comprobar usuario

El sistema permitirá comprobar los datos ingresados en el inicio de sesión, para permitir o no el acceso al sistema.

RF50. Organizar órdenes

El sistema permitirá ordenar las órdenes por tiempo de llegada

RF51. Alta estación

El sistema permitirá registrar los datos de las estaciones.
De los cuales se almacenará:

- id
- nombre

Actores participantes: administrador, Gerente, Cajero.

RF52. Baja estación

El sistema permitirá dar de baja estaciones.

Actores participantes: administrador, Gerente, Cajero.

RF53. Modificar estacion

El sistema permitirá modificar los datos de las estaciones.

Actores participantes: administrador, Gerente, Cajero.

RF54. Listar estación

El sistema permitirá listar los datos de las estaciones.

Actores participantes: administrador, Gerente, Cajero.

RF55. Buscar estacion

El sistema permitirá buscar los datos de las estaciones.

Actores participantes: administrador, gerente, Cajero.

RF56. Comprobar estación

El sistema permitirá comprobar el estado de las estaciones.

Actores participantes: administrador, gerente, Cajero.

RF57. Alta metodo Pago

El sistema permitirá registrar los datos de los métodos de pago.

De los cuales se almacenará:

- id
- nombre
- estadoPago

Actores participantes: administrador, Gerente, Cajero.

RF58. Baja metodo Pago

El sistema permitirá dar de baja métodos de pago.

Actores participantes: administrador, Gerente, Cajero.

RF59. Modificar metodo Pago

El sistema permitirá modificar los datos de los métodos de pago.

Actores participantes: administrador, Gerente, Cajero.

RF60. Listar método Pago

El sistema permitirá listar los datos de las métodos de pago

Actores participantes: administrador, Gerente, Cajero.

RF61. Buscar metodo Pago

El sistema permitirá buscar los datos de los métodos de pago.

Actores participantes: administrador, gerente, Cajero.

RF62. Comprobar método Pago

El sistema permitirá comprobar el estado de los métodos de pago.

Actores participantes: administrador, gerente, Cajero.

RF63. Alta usuarios del sistema

El sistema permitirá registrar los datos de todos los usuarios del sistema.

De los cuales se almacenará:

- id
- nombre
- estadoPago

Actores participantes: administrador, Gerente, Cajero.

RF64. Baja usuarios del sistema

El sistema permitirá dar de baja a todos los usuarios del sistema.

Actores participantes: administrador, Gerente, Cajero.

RF65. Modificar usuarios del sistema

El sistema permitirá modificar los datos de todos los usuarios del sistema.

Actores participantes: administrador, Gerente, Cajero.

RF66. Listar usuarios del sistema

El sistema permitirá listar los datos de todos los usuarios del sistema.

Actores participantes: administrador, Gerente, Cajero.

RF67. Buscar usuarios del sistema

El sistema permitirá buscar los datos de todos los usuarios del sistema.

Actores participantes: administrador, gerente, Cajero.

RF68. Comprobar usuarios del sistema

El sistema permitirá comprobar el estado de todos los usuarios del sistema.

Actores participantes: administrador, gerente, Cajero.

RF69. Inicio de sesión

El sistema permitirá ingresar sesión a cada tipo de usuario que se le permita con una contraseña o pin.

Actores participantes: administrador, Gerente, Cajero, mozo, Chef .

RF70. Ordenar productos por precio

El sistema permitirá ordenar los productos por precio en las listas

Actores participantes: administrador, Gerente, Mozo.

RF71. Categorizar productos por menú

El sistema permitirá ordenar los productos por menú en las listas

Actores participantes: administrador, Gerente, Mozo.

RF72. Categorizar productos por tipo

El sistema permitirá ordenar los productos por tipo en las listas

Actores participantes: administrador, Gerente, Cajero.

RF73. Categorizar productos por aptitud (celíaco, vegano, vegetariano)

El sistema permitirá ordenar los productos por aptitud los productos aptos para celíacos, veganos, vegetarianos en las listas

Actores participantes: administrador, gerente, Mozo.

RF74. Gestionar stock

El sistema permitirá gestionar el stock de los productos

Actores participantes: administrador, gerente.

RF75. Cerrar sesión

El sistema permitirá al usuario cerrar sesión.

RF76. Ver Menu

El sistema permitirá al cliente y al usuario ver el menú.

RF77. Generar estadísticas en base a las ventas

El sistema permitirá generar estadísticas en base a las ordenes y a lo que mas se pidio en el ultimo tiempo. .

2.8.3 Matriz de trazabilidad de objetivos y requerimientos funcionales

O	O	O	O	O	O	O	O	O	O
B	B	B	B	B	B	B	B	B	B
J	J	J	J	J	J	J	J	J	J
1	2	3	4	5	6	7	8	9	10

RF1	X							X	X	
RF2	X							X	X	
RF3	X							X	X	
RF4	X							X	X	
RF5	X							X	X	
RF6	X							X	X	
RF7	X							X		
RF8	X							X		
RF9	X							X		
RF10	X							X		
RF11	X							X		
RF12	X							X		
RF13	X								X	
RF14	X								X	
RF15	X								X	
RF16	X								X	
RF17	X								X	
RF18	X								X	
RF19	X									
RF20	X									
RF21	X									
RF22	X									
RF23	X									
RF24	X									
RF25	X	X								

RF26	X	X								
RF27	X	X								
RF28	X	X								
RF29	X	X	X							
RF30	X	X								
RF31	X			X						
RF32	X			X						
RF33	X			X						
RF34	X			X						
RF35	X			X						
RF36	X			X						
RF37	X			X						
RF38	X			X						
RF39	X			X						
RF40	X			X						
RF41	X			X						
RF42	X			X						
RF43	X								X	
RF44	X								X	
RF45	X								X	
RF46	X								X	
RF47	X								X	
RF48	X								X	
RF49	X				X	X	X			
RF50	X							X	X	

RF51	X									
RF52	X									
RF53	X									
RF54	X									
RF55	X									
RF56	X									
RF57	X	X								
RF58	X	X								
RF59	X	X								
RF60	X	X	X							
RF61	X	X								
RF62	X	X								
RF63	X				X	X	X			
RF64	X				X	X	X			
RF65	X				X	X	X			
RF66	X				X	X	X			
RF67	X				X	X	X			
RF68	X				X	X	X			
RF69	X				X	X	X			
RF70	X			X						
RF71	X			X						
RF72	X			X						
RF73	X			X						
RF74	X									X
RF75	X				X	X	X			

RF76	x									
RF77	x		x							

2.8.4 Requerimientos No Funcionales

RNF1. Todos los formularios deben validar la información que ingrese el usuario.

RNF2. Deben existir distintos tipos de usuarios con diferentes contraseñas.

RNF3. El sistema debe ser intuitivo y agradable.

RNF4. El sistema debe implementarse en un entorno web

RNF5. El sistema debe ser desarrollado en PHP.

RNF6. El sistema debe ser capaz de usarse tanto en PC como en celulares.

RNF7. El sistema deberá contar con una base de datos remota MySQL en la cual se registre la información de las entidades

RNF8. El sistema debe ser dividido en base a los roles y usuarios.

RNF9. El sistema debe contar con un usuario superior el cual pueda acceder a todas las funcionalidades y modificar todo.

2.9 Alcances y limitaciones

- **Alcances:**

El sistema a desarrollar va a cubrir las necesidades del cliente de digitalizar procesos y registro, gestión de comandas, gestión de stock, elaboración de gráficos e informes sencillos acerca de las ventas, menú en formato digital y formar la base para implementar servicio de delivery.

- **Limitaciones:**

El sistema no cubrirá cuestiones administrativas en el área de gestión de empleados, finanzas y formalidades legales.

2.10 Estudio de alternativa

En el mundo del alojamiento web, Linux es considerado como el mejor sistema operativo para servidores web. El sistema está disponible desde 1992 como software libre y, con los conocimientos necesarios, se puede ajustar a voluntad gracias a su estructura sencilla y modular. Los costes se originan cuando se emplean distribuciones con una oferta de asistencia técnica de pago. Con su fiabilidad, estabilidad y eficiencia, Linux se ha convertido en la mejor solución para servidores de correo electrónico y servidores web.

Linux constituye una alternativa excelente si lo que se quiere es usar PHP, Perl, Python, Ruby o MySQL. Si se buscan soluciones disponibles para páginas principales, aplicaciones como blogs, sistemas de gestión de contenidos o foros de discusión, el hosting de Linux cuenta con diversas aplicaciones de código abierto, mientras que el software para alojamiento web de windows está sujeto, en la mayoría de los casos, a pago.

En contraste con Windows, la filosofía de Linux consiste en mantener la estructura del sistema operativo lo más sencilla posible. Por ello, todos los componentes, incluso los equipos y los procesos, son un archivo y los ajustes en el núcleo del sistema se pueden llevar a cabo en cualquier momento. Hay muchas herramientas para la configuración y administración del sistema y estas están normalmente disponibles por líneas de comandos, pero también en GUI (interfaces gráficas de usuario) para casi todas las aplicaciones. Las mayores libertades están ligadas a una gran responsabilidad, lo que puede desbordar a los usuarios menos experimentados.

A la hora de valorar entre un sistema operativo u otro, lo primero que debemos de tener claro es que el sistema operativo de servidor Linux ofrece una serie de ventajas sobre Windows, siendo Linux mucha más estable, al tratarse de servidores estos deben de estar en todo momento encendidos, las 24 horas del día los 365 días del año por lo que es crítico que el sistema operativo en ningún momento se bloquee, Los sistemas Linux pueden ejecutarse durante años sin fallar o sin necesidad de reiniciar, un atributo crucial y que debemos de valorar considerablemente ya que el tiempo de inactividad puede ser desastroso para una empresa, otra de las grandes ventajas que conseguimos con una distribución Linux, es que también puede administrar y ejecutar una gran cantidad de procesos múltiples simultáneamente, y las actualizaciones se llevan a cabo sin problemas, sin la necesidad constante de reiniciar en Windows, básicamente se traduce en que a la hora de definir los requisitos de nuestro servidor podremos optar por una especificación más baja a nivel de hardware, ya que una distribución Linux no consume tantos recursos como una distribución Windows, además de todo lo ya comentado la seguridad es un factor importante, especialmente con la creciente ola de delitos cibernéticos, linux, debido a su diseño como sistema multiusuario, es mucho más seguro que Windows. Por esto elegiremos linux.

2.10.1 Análisis de los sistemas operativos a implementar

2.10.1.1 Server Ubuntu

Ubuntu Server es una variante de Ubuntu que sale con cada versión y está dedicada especialmente para su uso en servidores. El uso de Ubuntu como servidor se ha extendido mucho en los últimos años, sobretodo para el uso de servidores web, de un modo tanto particular como profesional.

ubuntu Server es un Sistema Operativo sin entorno gráfico (aunque podemos instalarlo) lo que quiere decir que todas las acciones se realizan mediante consola, y normalmente ni si quiera a través de el propio servidor, sino desde una conexión remota. El manejo de Ubuntu Server es muy similar al de cualquier otro Sistema Linux, pero con las particularidades de Ubuntu (como el sudo).

Cuando estamos instalando Ubuntu Server nos hace una serie de preguntas sobre que tipo de servicios queremos instalar, entre una lista de los más típicos, y nos pregunta los parámetros necesarios para su configuración. De este modo podemos instalar de una forma fácil y sencilla un servidor acorde a nuestras necesidades en unos pocos minutos.

Es muy habitual encontrarnos Ubuntu Server como sistema operativo en muchos de los VPS que podemos contratar en la mayoría de compañías, aunque también nos suelen dar a elegir otras distribuciones Linux.

2.10.1.2 Licenciamiento

Ubuntu es una plataforma de código abierto de confianza. Para mantener esa confianza, debemos administrar el uso de Ubuntu y los componentes que contiene con mucho cuidado. De esta manera, cuando las personas usen Ubuntu, o cualquier cosa que lleve la marca Ubuntu, pueden estar seguros de que cumplirá con los estándares que esperan. Su uso continuo de IP de Canonical implica su aceptación y reconocimiento de esta Política de derechos de propiedad intelectual.

Puede descargar, instalar y recibir actualizaciones de Ubuntu de forma gratuita.

Puede modificar Ubuntu para uso personal o comercial interno.

Puede redistribuir Ubuntu, pero solo donde no haya habido modificaciones.

Puede utilizar nuestros materiales de derechos de autor, patentes y diseños de acuerdo con esta Política de derechos de propiedad intelectual.

Puede estar seguro y confiar en la consistencia de la experiencia de Ubuntu.

Puede confiar en el estándar esperado de Ubuntu.

Ubuntu es un trabajo agregado; esta política no modifica ni reduce los derechos otorgados bajo licencias que se aplican a trabajos específicos en Ubuntu.

Relación con otras licencias

Ubuntu es un trabajo agregado de muchos trabajos, cada uno cubierto por su(s) propia(s) licencia(s). Con el fin de determinar qué puede hacer con trabajos específicos en Ubuntu, esta política debe leerse junto con la(s) licencia(s) de los paquetes correspondientes. Para evitar dudas, cuando cualquier otra licencia otorgue derechos, esta política no modifica ni reduce esos derechos bajo esas licencias.

2.10.1.3 Soporte

Microprocesador

Ubuntu acepta tres arquitecturas o familias de procesadores, que son la inmensa mayoría si no la práctica totalidad:

- Intel x86 y compatibles.
- AMD64, EM64T y compatibles.
- PowerPC y compatibles (hasta la versión 6.10 Edgy Eft).

Tarjetas de video

La mayoría de las tarjetas de video tienen compatibilidad Vesa / VGA por lo que en pocos casos perderemos la capacidad gráfica por completo. Sin embargo la capacidad para obtener el 100% del rendimiento de estas tarjetas depende mucho de la disponibilidad de los drivers. Para asegurarnos es recomendable escoger tarjetas de video de los siguientes fabricantes.

NVIDIA: Casi cualquier tarjeta de este fabricante tiene o ha tenido compatibilidad con Linux, las más recomendables:

- nVIDIA GeForce FX 5200
- nVIDIA GeForce 6200 LE
- nVIDIA GeForce 7600 LE
- nVIDIA GeForce 8400M GS LE (portátil)
- nVIDIA GeForce FX5500 AGP (referencia en pc coste es 14432)

Para tarjetas más antiguas o no listadas consulta la web oficial de NVIDIA en la sección de Download drivers.

ATI: Ocurre lo mismo con este fabricante. Algunas tarjetas compatibles:

- ATI Radeon 9250
- ATI Radeon 1550 Pro
- ATI Radeon X1200 Series

Tarjetas de TV

- Genius Video Wonder Pro II
- Woxter USB TV Total siguiendo este manual de instalación y consiguiendo aquí el firmware.
- Pinnacle PCTV Hybrid Pro PCI (Con Kaffeine en Ubuntu 8.04)
- Easy TV USB TDT Stick siguiendo este manual de instalación. (Probada con Kaffeine en Ubuntu 8.04, 8.10 y 9.04)
- Hauppauge WinTV NOVA-T Stick chipset 7060 y 7070.

Tarjetas de sonido

- Creative SoundBlaster Live! 24-bit
- Creative SoundBlaster Connect USB

Tarjetas de red inalámbricas

- DWL-G510 (rev.c1, rev.c2, requieren instalación manual y solo funcionan en modo ad-hoc y managed)
- Conceptronic C54Ri
- Realtek Semiconductor Co., Ltd. RTL8101E/RTL8102E PCI Express Fast Ethernet controller

Tarjetas de red

- Tarjeta de red Conceptronic PCI card 10/100 (referencia pc coste 56361)

Placas base

Casi cualquier tarjeta de este fabricante tiene o ha tenido compatibilidad con Linux,

- Asus M2NPV-VM
- Gigabyte GA-P35-DSR3 (Revision 2.1)
- Gigabyte GA-MA69VM-S2 (Revisión 1.0)

2.10.1.4 Requisitos

Los requisitos para una versión server linux son mínimos debido a que no utiliza el entorno gráfico, pero para que actúe como servidor dependiendo del tráfico que tengamos puede requerir más.

Mínimo (Consola)

- 256 MB de memoria
- 2 Gb de espacio en HDD (Incluido swap)
- AMD o Intel Procesador de 64-32bits
- Incluido AMD Optaron y Intel EM64T Xeon, para versiones de 64.

Mínimo (Gráfico)

- 512 MB de memoria
- 4 Gb de espacio en HDD (Incluido swap)
- AMD o Intel Procesador de 64-32bits
- Tarjeta Gráfica VGA, monitor con resolución de 800x600

Mínimo Recomendado

Requisitos recomendados para levantar un servicio webmail para 25 usuarios. Teniendo en cuenta que cada usuario tiene 1gb de espacio asignado para almacenamiento de correo electrónico, deberíamos contar con la siguiente arquitectura:

- 27 Gb de HDD (25gb para usuarios + 2gb de SO) Con lo cual con un disco standard de 40Gb nos alcanza para mantener a los 25 usuarios y además realizar políticas de mantenimiento como (agregar usuarios, realizar back-up comprimidos de los mail, entre otros).
- 512Mb-1Gb de memoria ram nos alcanzaría para que el sistema responda sin problemas en caso de contar con mail "pesados" o simultaneidad de acceso, aunque esto también estará condicionado por los recursos de la red.
- Procesador Intel o AMD 64 o 32 bit, en nuestro caso usamos un Intel Dual-Core 2.4Ghz 32bits

2.10.1.5 Seguridad

Por qué Ubuntu es seguro:

- Ubuntu distingue limpiamente entre usuarios normales y usuarios administrativos.
- El software para Ubuntu se guarda en línea en un repositorio seguro, el cual no contiene software falso o malicioso.

- Software de código abierto como Ubuntu permite que los fallos de seguridad sean fácilmente detectados.
- Los parches de seguridad para software de código abierto como Ubuntu son liberados frecuente y rápidamente.
- La mayoría de los virus diseñados principalmente para sistemas basados en Windows no afectan a sistemas Ubuntu.

2.10.1.6 Gestión de los usuarios

Básicamente, hay dos tipos de usuarios en los sistemas basados en GNU/Linux o Unix: el administrador (root) y el resto de usuarios.

Nos vamos a centrar en sistemas basados en GNU/Linux, en concreto utilizaremos Ubuntu. Sin embargo, la mayor parte de lo explicado se puede utilizar en cualquier sistema GNU/Linux o Unix.

El usuario administrador

El usuario root es el usuario utilizado para administrar el sistema. Este usuario puede realizar cualquier tarea de administración sobre el sistema. Es decir, este usuario puede hacer cualquier cosa con el sistema.

Este “poder” es peligroso, porque cualquier fallo ocasionado con la cuenta de root puede afectar a la seguridad de todo el sistema. Hay que tener en cuenta, que cualquier aplicación que se ejecute con la cuenta de root, tendrá los mismos permisos que root, por tanto podrá hacer cualquier cosa en el sistema. Por tanto es necesario tomar una serie de medidas de seguridad para evitar problemas con el uso de la cuenta de root.

Veamos algunas de estas medidas de seguridad:

- Usar la cuenta de root solo cuando sea necesario. Para ello, el administrador del sistema deberá contar con una cuenta de usuario normal, que será la que utilice para trabajar con el sistema. Cuando necesite realizar alguna tarea de administración, se cambiará a la cuenta de root para llevarla a cabo. Una vez terminada la tarea de administración, vuelve a su cuenta normal.
- No permitir inicio de sesión con la cuenta root. Aunque esta medida no es necesaria, si está siendo utilizada cada vez más. De esta forma, se obliga a que el administrador del sistema tenga que tener una cuenta de usuario normal y acceder al sistema a través de ésta. Es una manera de forzar el cambio a root solo cuando sea necesario.

Cambio de usuario al usuario root

El sistema proporciona una serie de mecanismos para poder cambiar al usuario root desde una cuenta de usuario normal o para ejecutar comandos como usuario root. Por supuesto, solo podrá hacer este cambio aquel o aquellos usuarios que queramos que tengan permisos de administración.

Para seleccionar qué usuario puede llevar a cabo tareas de administración como root, hay dos técnicas de protección básicas: conocer la contraseña de root e iniciar sesión como root o configurar el sistema para que ciertos usuarios puedan ejecutar comandos como si fueran root.

La primera opción, proporcionar la contraseña de root e iniciar sesión, conlleva proporcionar cierta información sensible a uno o varios usuarios y, sobre todo, facilita la vulnerabilidad del sistema. Al trabajar como root de forma continua, se facilita la posibilidad de cometer un error humano a la hora de trabajar con la cuenta al aumentar el tiempo que estamos en el sistema como root.

La segunda opción se basa en mantener una lista con los usuarios que pueden ejecutar tareas como root. De tal forma que, cuando un usuario quiera realizar una operación de administración, se comprueba si ese usuario pertenece o no a la lista de usuarios con permisos de administración. Si pertenece, se comprueba si realmente es quien dice ser, pidiendo una contraseña. Esta opción es más fácil de mantener y permite realizar, hasta cierto punto, seguimiento de las

id	Muestra información de identificación de una cuenta de usuario
passwd	Cambia la contraseña de una cuenta de usuario
adduser	Añade un usuario al sistema Añade un usuario a un grupo.
addgroup	Añade un grupo al sistema
deluser	Elimina un usuario del sistema Elimina un usuario de un grupo
delgroup	Elimina un grupo del sistema
usermod	Modifica la cuenta de un usuario en el sistema.

operaciones realizadas.

FIG36. Comandos para agregar usuarios

2.10.1.7 Gestión de red

En ubuntu se usa NetworkManager para la gestión de red, es un sistema de configuración y control de red dinámico que intenta mantener los dispositivos y conexiones de red activos y activos cuando están disponibles. Finalmente, NetworkManager proporciona una API rica a través de D-Bus que permite que las aplicaciones consulten y controlen la configuración y el estado de la red

2.10.1.8 Particularidades

- Versátil. Puede instalarse en equipos de sobremesa, ordenadores portátiles o en servidores (para gestionar almacenamiento web, por ejemplo).
- Diferentes entornos de escritorio. La apariencia de Ubuntu puede variar haciendo uso de una serie de entornos de escritorio. Cada uno proporciona una serie de herramientas como la administración de archivos, visores de imágenes y documentos PDF, y similares.
- Incluye controladores Nvidia. Ubuntu incorpora controladores para el principal fabricante de tarjetas gráficas, lo que aumenta el rendimiento de los equipos donde se instala, a la vez que reduce el tiempo de configuración del sistema tras su instalación.
- Arranque y apagado veloz. Ubuntu es un sistema que se caracteriza por iniciarse o apagarse en muy poco tiempo.

2.10.1.9 Server DEBIAN

El Proyecto Debian es una asociación de personas que han hecho causa común para crear un sistema operativo (SO) libre. Este sistema operativo que hemos creado se llama Debian.

Un sistema operativo es un conjunto de programas y utilidades básicas que hacen que su computadora funcione. El centro de un sistema operativo es el núcleo (N. del T.: kernel). El núcleo es el programa más importante en la computadora, realiza todo el trabajo básico y le permite ejecutar otros programas.

Los sistemas Debian actualmente usan el núcleo de Linux o de FreeBSD. Linux es una pieza de software creada en un principio por Linus Torvalds y desarrollada por miles de programadores a lo largo del mundo. FreeBSD es un sistema operativo que incluye un núcleo y otro software.

Sin embargo, se está trabajando para ofrecer Debian con otros núcleos, en especial con el Hurd. El Hurd es una colección de servidores que se ejecutan sobre un micronúcleo (como Mach) para

implementar las distintas funcionalidades. El Hurd es un software libre producido por el proyecto GNU.

Una gran parte de las herramientas básicas que completan el sistema operativo, vienen del proyecto GNU; de ahí los nombres: GNU/Linux, GNU/kFreeBSD, y GNU/Hurd. Estas herramientas también son libres. Desde luego, lo que la gente quiere es el software de aplicación: herramientas que los ayuden a realizar lo que necesiten hacer, desde editar documentos, ejecutar aplicaciones de negocios hasta divertirse con juegos y escribir más software. Debian viene con más de 59000 paquetes (software precompilado y empaquetado en un formato amigable para una instalación sencilla en su máquina), un gestor de paquetes (APT), y otras utilidades que hacen posible gestionar miles de paquetes en miles de ordenadores de manera tan fácil como instalar una sola aplicación. Todos ellos de forma gratuita.

Es un poco como una torre. En la base está el núcleo. Encima se encuentran todas las herramientas básicas. Después está todo el software que usted ejecuta en su computadora. En la cima de la torre se encuentra Debian — organizando y encajando todo cuidadosamente para que todo el sistema trabaje junto.

2.10.1.10 Licenciamiento

A algunas personas les gusta ayudar a otras. Muchas escriben programas para aprender más acerca de los computadores. Más y más personas están buscando maneras de evitar los precios inflados del software. Un grupo creciente contribuye como un agradecimiento por todo el excelente software libre y gratuito que ha recibido de otros. En las instituciones académicas muchos crean software libre para ayudar a obtener los resultados de sus investigaciones en un uso más amplio. Las empresas ayudan a mantener el software libre para poder observar cómo se desarrolla éste, no hay una manera más rápida de obtener una nueva característica que implementarla por diversión

Debian está tan comprometido con el software libre que pensamos que sería útil que ese compromiso se encontrara formalizado en algún tipo de documento. Por ello nació el Contrato Social.

Aunque Debian cree en el software libre, existen casos en los que la gente quiere o necesita disponer de software que no es libre en sus máquinas. Siempre que sea posible, Debian respaldará esto. Hay un número creciente de paquetes que tienen como única misión instalar software que no es libre en un sistema Debian

2.10.1.11 Soporte

Debian funcionará en casi todos los ordenadores personales, incluyendo la mayoría de los modelos más antiguos. Cada nueva versión de Debian generalmente soporta un mayor número de arquitecturas de ordenadores. Si quiere una lista completa de las que se soportan actualmente, vea la documentación para la versión estable.

Casi todo el hardware común está soportado. Si desea asegurarse de que todos los dispositivos conectados a su máquina están soportados hay una wiki llamada [InstallingDebianOn](#)

Existen algunas compañías que hacen difícil el soporte al no publicar las especificaciones de su hardware. Esto quiere decir que a lo mejor no pueda usar su hardware con GNU/Linux. Algunas compañías proporcionan drivers que no son libres, pero eso es un problema, porque la compañía podría quebrar o dejar de dar soporte a su hardware. Recomendamos que sólo adquiera hardware de fabricantes que proporcionen controladores libres para sus productos.

2.10.1.12 Requisitos de instalación

Lo primero que necesitamos es una CPU. Y la mínima para poder instalar este Linux es contar con un Pentium 4 a 1 GHz. Los responsables de Debian recomiendan un Intel i5, o equivalente en AMD, para que vaya fluido, pero para instalarlo, y que funcione, basta con mucho menos.

Otro de los requisitos para poder instalar este SO en el PC es tener, como mínimo, 256 MB de memoria RAM. Si lo comparamos con el mínimo de RAM de Windows 11 (que son 4 GB), la diferencia es notable. Eso sí, lo recomendable, para funcionar bien, es 1 GB.

Y, por último, el espacio libre en el disco duro mínimo es de 10 GB, aunque se recomienda tener al menos 20 GB para que el sistema no tenga problemas y pueda escribir sus propios archivos sin impedimentos. Y, si lo instalamos en un SSD, mucho mejor.

2.10.1.13 Seguridad

La experiencia ha demostrado que la seguridad por medio de la oscuridad nunca funciona. La revelación pública permite encontrar soluciones mejores, y en un menor plazo de tiempo, a los problemas de seguridad. Esta página muestra la situación de Debian respecto a diversos agujeros de seguridad conocidos que podrían, potencialmente, afectar al sistema operativo Debian.

El proyecto Debian coordina muchos avisos de seguridad con otros agentes del software libre, con el resultado de que estos avisos se publican el mismo día que se hace pública la vulnerabilidad. Además, tenemos nuestro propio equipo de auditoría de seguridad. El equipo de auditoría de seguridad revisa el archivo y busca fallos de seguridad no corregidos.

Debian también participa en los esfuerzos de estandarización de seguridad:

- Los avisos de seguridad de Debian son compatibles con CVE (revise las referencias cruzadas).
- Debian está representado en el foro del proyecto Open Vulnerability Assessment Language (n.t. Lenguaje abierto de detección de vulnerabilidades)

Las distribuciones de Debian no son vulnerables a todos los problemas de seguridad. El sistema de seguimiento de problemas de seguridad de Debian recolecta toda la información sobre el estado de vulnerabilidad de los paquetes de Debian. Permite hacer búsquedas por nombre de CVE o por paquete.

2.10.1.14 Particularidades

Estas son las particularidades de DEBRIAN:

- Debian es software libre.
- Debian está hecho de software libre y de código abierto y siempre será 100% libre. Cualquiera es libre de usarlo, modificarlo y distribuirlo. Esta es nuestra principal promesa a nuestros usuarios. Además, es gratuito.
- Debian es estable y seguro.
- Debian es un sistema operativo basado en Linux adecuado para un amplio rango de dispositivos incluyendo portátiles, ordenadores de escritorio y servidores. Proporcionamos una configuración predeterminada razonable para cada paquete así como actualizaciones de seguridad con regularidad durante su ciclo de vida.
- Debian tiene un soporte de hardware extenso.
- La mayoría del hardware ya está soportado por el núcleo Linux, lo que significa que está soportado en Debian también. Hay disponibles controladores no libres para el hardware si fueran necesarios.
- Debian ofrece un instalador flexible.
- Nuestro CD «en vivo» es para cualquier persona que quiera probar Debian antes de instalarlo. También incluye el instalador Calamares que facilita instalar Debian desde el sistema «en vivo». Usuarios con más experiencia pueden usar el instalador de Debian,

que dispone de más opciones de personalización, incluyendo la posibilidad de usar una herramienta automatizada de instalación en red.

- Debian proporciona actualizaciones de versión sin complicaciones.
- Es fácil mantener nuestro sistema operativo actualizado, ya sea una actualización completa a una versión superior o una actualización de un solo paquete.
- Debian es la base de muchas otras distribuciones.
- Muchas de las distribuciones Linux más populares como Ubuntu, Knoppix, PureOS o Tails están basadas en Debian. Proporcionamos todas las herramientas por lo que todo el mundo puede extender los paquetes de software del archivo de Debian con sus propios paquetes si lo necesitan.
- El proyecto Debian es una comunidad.
- Cualquiera puede ser parte de nuestra comunidad; no necesita ser una persona que programe o administre sistemas. Debian tiene una estructura de gobierno democrático. Al tener todos los miembros de Debian iguales derechos, no puede ser controlada por una sola empresa. Tenemos desarrolladores en más de 60 países y Debian en sí se traduce a más de 80 idiomas.

2.10.1.15 Server FEDORA

En primer lugar, Fedora es un sistema operativo basado en Linux creado en 2003 como una distribución comunitaria llamada Proyecto Fedora. Fue lanzado junto con Red Hat Enterprise Linux, que iba a ser la próxima distribución oficial de Linux después de que se suspendiera Red Hat Linux. Su objetivo principal era servir como fuente de innovaciones de software para las versiones de Hat Enterprise Linux. Esta conexión entre las dos distribuciones de Linux explica el origen del nombre de Fedora: 'fedora' se refiere al característico sombrero fedora utilizado en el logotipo "Shadowman" de Red Hat.

2.10.1.16 Licenciamiento

Fedora contiene software distribuido bajo varias licencias gratuitas y de código abierto y apunta a estar a la vanguardia de las tecnologías libres

2.10.1.17 Soporte

Linux es capaz de soportar toda clase de componentes y periféricos. Fedora incluye todos los controladores proporcionados por los

proyectos que son responsables del soporte físico de Linux. El proceso de instalación de Fedora automáticamente detecta los componentes físicos compatibles con su equipo.

Algunos modelos específicos de componentes aún no están soportados bajo Linux, o requieren controladores que son propiedad del fabricante. Si Fedora no detecta automáticamente un dispositivo, puede que necesite instalar controladores adicionales o de soporte lógico incorporado.

Conexiones de red

Los módems y conexiones de red inalámbricas requieren una configuración adicional después de completar el proceso de instalación.

2.10.1.18 Requisitos de instalación

Requerimientos mínimos del sistema

- 6GB de espacio de disco libre
- 2GB de RAM

Requerimientos recomendados del sistema

- 6GB más el espacio de disco requerido por el sistema operativo por huésped. Para la mayoría de sistemas operativos se recomiendan más de 6GB de espacio libre.
- Un núcleo de procesador o hiper-hilo para cada CPU virtualizada y para el hipervisor.
- 2 GB de RAM, y RAM adicional para los huéspedes virtualizados.

Requerimientos de KVM

- Un procesador Intel con las extensiones Intel VT e Intel 64 o
- Un procesador de AMD con extensiones AMD-V y AMD64.

Soporte de almacenaje

Los métodos existentes de almacenamiento de huéspedes son: archivos en almacenaje local, particiones de discos duros físicos

- LUN físicamente conectados
- Particiones LVM
- iSCSI
- LUNs basados en canal de fibra.

2.10.1.19 Seguridad

Esta edición es mantenida por una comunidad de evaluadores de seguridad y desarrolladores. Viene con el sencillo y rápido entorno de

Escritorio Xfce y con un menú personalizado que permite tener todos los instrumentos indispensables para seguir una ruta de prueba para verificaciones de seguridad o para rescatar a un sistema corrupto. La imagen viva se ha diseñado para hacer posible la instalación de software mientras se ejecuta, y si se usa desde una memoria USB creada con el Creador LiveUSB mediante la herramienta de recubrimiento, podrá instalar, actualizar el software y grabar los resultados de las pruebas en forma permanente.

2.10.1.20 Particularidades

Mejor administración de energía

Linux jamás ha hecho un excelente trabajo en la dirección de la batería del computador portátil. Favorablemente, Linux no consume mucha energía por lo que habitualmente no acaba con la vida útil de una batería nueva. Pero, los beneficios de gestión de energía rediseñadas por Fedora 15 le proporcionan un mejor rendimiento y durabilidad de la batería que otros tipos Linux.

Esto fue comprobado al ser ejecutado por IntecBattery Mark 1.1 en Windows XP en un computador virtual de VirtualBox de marca Lenovo ThinkPad R61, con un procesador Intel Core 2 Duo de 2.2GHz T7500 y 2GB de RAM. Con el sistema Mint 11, la batería de seis celdas duró dos horas y cincuenta y cinco minutos. Eso es lo suficiente bueno para que una batería dure tres años o más.

Con Fedora 15, se logró llegar mucho más allá de las 3 horas de uso, haciéndola más duradera. Pero siempre dependerá de su sistema interno, recuerda que fedora es el sistema más comprensivo del planeta. Cualquier organismo que ofrezca una vida útil a una batería por medio de este sistema obtendrá la mejor ventaja del sistema operativo.

Mejor software para el usuario final

Fedora usa el LibreOffice, como opción principal en vez de OpenOffice. Mientras que la ramificación de LibreOffice se adapta mejor a la versión mejorada de OpenOffice, se busca que este sistema funcione correctamente y cada vez más rápido y eficaz.

Aunque tiene numerosas correcciones de errores mínimos y posee una compatibilidad mucho mejor con el formato de registro de Microsoft Office. Este sistema operativo es de gran importancia para su computador portátil.

Además, Fedora contiene Firefox 4, que sin duda es el mejor sistema en comparación a la serie de Firefox 3.x. Por otro lado, hubiera sido

más eficaz si se hubiera utilizado Chrome 11. Pero, sin ningún problema Linux, logró más adelante instalar Chrome caracterizándolo como el navegador establecido que lleva muchos años de uso.

Firewall dinámico

Esta es una exclusiva beta, por lo que se debe instalar de forma manual y solo los directores de red logran encontrarla tan emocionante, aunque numerosas veces todavía utiliza su esquema de administrador de red, para así garantizar un mejor potencial.

El firewall dinámico admite cambiar la disposición del firewall sin necesidad de reiniciar el firewall. Por ejemplo, consigue configurar el firewall para modificar sus reglas, según sea preciso para los aparatos virtuales o redes privadas virtuales.

También logra usar para apertura el firewall para una solicitud de red determinada, como manifestar una impresora local o un servicio de Windows para luego cerrar el puerto una vez que haya culminado con ese procedimiento. Esto posee posibilidades existentes que les agrada a todos los usuarios. Si todo marcha bien, se transformara en el firewall establecido en Fedora 15 y posteriormente, en la red Hat Enterprise Linux (RHEL).

Integrado con GNOME 3

GNOME 3 es la próxima reproducción de GNOME con una interfaz optimizada para el usuario. Suministra un escritorio totalmente nuevo y actual que ha sido planteado por los mismos beneficiarios estableciendo técnicas actuales. Cuando decidieron implementar este nuevo sistema se pensó en un escritorio totalmente nuevo. Aunque la serie 2.x de GNOME fue muy reconocida y la mayoría de los usuarios se adaptaron a ella, aun así quieren una nueva versión originando el GNOME 3 de Fedora.

Este sistema funcionó bien y no tuvo que ser adaptado a otro sistema para ser usado. El término de la Unidad de Ubuntu, es un shell esencialmente diferente que se sitúa encima de GNOME. Unity está predestinado a todos los usuarios de Windows y Mac que jamás han utilizado Linux. También se encuentra diseñado para convertirse casualmente en una interfaz de comprimido. Cuando utilizas GNOME, lo que observas es solo un cambio insignificante en tu computador en comparación con otros sistemas más complejos.

En resumen, GNOME 3.0 es un sistema nuevo, pero para los beneficiarios es un paso atrás. Ya que Fedora 15 es la primera comercialización principal que contiene GNOME 3 de manera predeterminada. Eso fue un gran error. Lo mismo ocurrió con Fedora

4 en su primera versión. GNOME 3 no ofrece un escritorio ideal. Por ese motivo que los usuarios prefieren otra opción de fedora

2.10.2 Alternativas de diseño

2.10.2.1 Escritorio

2.10.2.1.1 Descripción

Una aplicación de Escritorio es la que es desarrollada para el entorno de dispositivos de escritorio como las PC. Y comúnmente no requiere más infraestructura que la misma PC donde es alojada, tanto para alojar datos o información, sumar funcionalidad y demás.

Es una opción viable para uso personal o de pequeños negocios donde se utiliza únicamente una estación de trabajo. También aplicaciones que no necesitan almacenar o acceder a información de forma remota.

2.10.2.1.2 Ventajas

Bajo costo de desarrollo y evita el gasto en una infraestructura dedicada para mantener el servicio activo. El cliente puede ya contar con el equipamiento necesario para comenzar a utilizar la aplicación, el cual es únicamente su PC. Al funcionar en un entorno cerrado y sin necesidad de comunicarse con módulos exteriores, se convierte en la alternativa virtualmente más segura.

2.10.2.1.3 Desventajas

No comparte datos e información, solo puede usarse un dispositivo para acceder a la información particular que desee manipularse mediante la aplicación. Establece un límite sobre cuáles son los dispositivos donde podrá ser utilizada la aplicación, reduciendo la cantidad de posibles clientes.

2.10.2.2 Web

2.10.2.2.1 Descripción

Una aplicación Web es almacenada en un servidor web, y es accedida y ejecutada dentro de un navegador como Mozilla Firefox, Google Chrome o Safari

Al momento que un usuario accede a la URL específica del recurso, se descarga una parte de la aplicación desde el servidor web al dispositivo.

Es la denominada capa de Presentación o UI (Interfaz de Usuario), y aporta el aspecto visual a la aplicación, comúnmente es creada utilizando HTML, CSS y JavaScript. La interfaz de usuario se comunica por medio del protocolo HTTP y una interfaz de Transferencia de Datos, con la capa de Lógica del Negocio o procesamiento de datos.

La interfaz de transferencia de datos aporta seguridad, evitando la comunicación directa entre la interfaz de usuario y el procesamiento de datos. Ya que todo el código que se descarga al dispositivo del cliente es de fácil acceso, se requiere que no contenga información que pueda comprometer la seguridad del sistema. Algunas de las más usadas en proyectos modernos pueden ser REST o GraphQL. La capa de procesamiento de datos se encarga de proporcionar funcionalidad a la aplicación, se mantiene a la escucha de peticiones por parte de la interfaz de usuario para, comúnmente, devolver datos. Es creada usando lenguajes de programación como Java, PHP, C# o Python, entre otros.

También será la encargada de comunicarse con la capa de Persistencia de Datos o Sistema Gestor de Bases de Datos (SGBD) alojado en un servidor de bases de datos.

2.10.2.2.2 Ventajas

Aunque requiere de un equipo capacitado para implementar la pila de tecnologías necesarias, una aplicación de este tipo reduce el tiempo y los costos de desarrollo si la comparamos con esfuerzos multiplataforma que utilicen otras arquitecturas. Ya que se deberá crear una única aplicación que podrá ser accedida desde cualquier dispositivo que cuente con un navegador web. También permite una rápida implementación de actualizaciones, e inmediata para todos los usuarios sin importar la plataforma o sistema operativo que utilicen. Las aplicaciones web no necesitan ser descargadas a través de una tienda como la App Store de iPhone o Play Store en Android.

2.10.2.2.3 Desventajas

Las aplicaciones web no ofrecen el mejor rendimiento entre las distintas arquitecturas por distintas razones. La necesidad de recuperar constantemente los datos desde una ubicación remota. La dependencia absoluta del estado de la conexión a Internet. Y el tiempo de procesamiento del navegador para interpretar y ejecutar código no compilado.

Tampoco permite una correcta utilización del hardware y características del SO en el dispositivo del usuario. Además, tienen un costo inicial y de mantenimiento del hardware requerido para alojar la aplicación y sus datos.

Es necesario un conjunto equipamiento de hardware dedicado para mantener la aplicación activa, esto significa un gasto inicial y de mantenimiento extra. También se puede contratar un servicio que lo proporcione, así se logra evitar el coste inicial pero el de mantenimiento será mayor.

2.10.2.3 *Mixto*

2.10.2.3.1 Descripción

Se denomina aplicación Mixta a la que es desarrollada para trabajar en varios sistemas operativos de forma nativa. Se trata de varias aplicaciones creadas cada una para el entorno de cada sistema operativo en que se desee su utilización. Su funcionamiento es muy similar al de una aplicación Nativa, debe ser descargada e instalada en los dispositivos, y ofrecerá un buen nivel de rendimiento e integración.

Aunque utilizará siempre una base de datos externa ya que son varios los dispositivos que harán uso de ella.

2.10.2.3.2 Ventajas

Acceso multiplataforma y la posibilidad de utilizar la máxima capacidad de procesamiento de los dispositivos donde sea instalada.

2.10.2.3.3 Desventajas

Costo elevado del desarrollo y mantenimiento de las aplicaciones debido a que cada una podría utilizar herramientas y tecnologías distintas. Por lo que requiere de personal específico y dedicado a cada tarea, multiplicado por el número de plataformas a las cuales se quiera llegar.

2.10.3 Opciones de arquitectura

2.10.4 Arquitectura 1

PC sin conexión a red local ni internet

esquema lógico



FIG37. Esquema lógico de la arquitectura 1

Particularidades

Para este caso solo es necesaria una estación de trabajo compatible y en funcionamiento. No requiere de conexión a Internet u otros dispositivos, las herramientas a utilizar son únicamente de software.

Ventajas:

- Este esquema es el más económico de los aquí mencionados.
- Cuenta con una gran seguridad para la base de datos.

Desventajas:

- No se puede acceder a la base de datos desde otra PC
- Solo permite crear aplicaciones de escritorio.

2.10.5 Arquitectura 2

Red local sin conexión a Internet, y con servidor de bases de datos local e independiente.

Esquema lógico

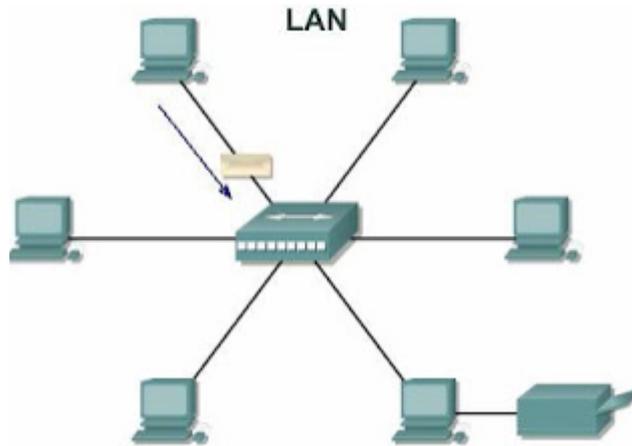


FIG38. Esquema lógico de la arquitectura 2

Particularidades

Las estaciones de trabajo y el servidor de bases de datos dedicado deben tener tarjetas de red para conectarse a la red local y poder comunicarse entre sí.

Ventajas:

- Permite la creación de aplicaciones híbridas.
- Sencilla de gestionar.

Desventajas:

- Impide la creación de aplicaciones con conexión a Internet.

2.10.6 Arquitectura 3

Red local con acceso a Internet, Firewall, Access Point y servidor de bases de datos local e independiente.

Esquema lógico

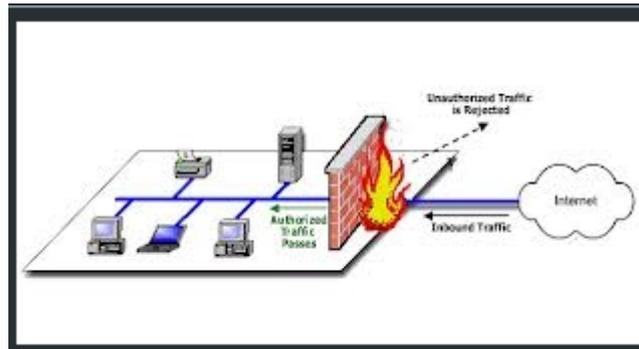


FIG39. Esquema lógico de la arquitectura 3

Particularidades

Las estaciones de trabajo, el servidor de bases de datos y servidor de Firewall dedicado deben tener tarjetas de red para conectarse a la red local y poder comunicarse con el resto de dispositivos. El Firewall se ejecuta en un servidor independiente para centralizar las configuraciones y el tráfico de la red.

En este caso se implementa un dispositivo modulador para conectar la red local con el Internet.

Ventajas:

- Permite la creacion de aplicaciones híbridas.
- Sencilla de gestionar.

Desventajas:

- Requiere configuración de red y firewall.
- La conexión a Internet produce una mayor inseguridad para la base de datos.

2.10.7 Arquitectura 4

Red local con Firewall, Access Point y servidor web, y un servidor de bases de datos remoto

Esquema lógico

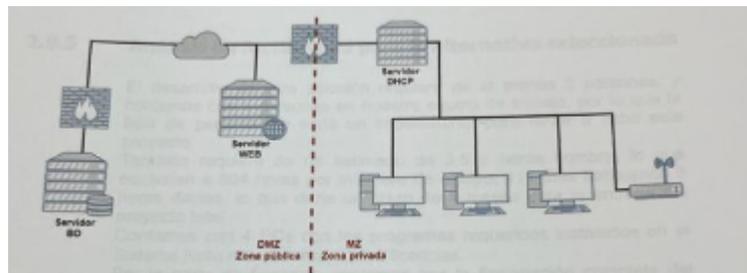


FIG40. Esquema lógico de la arquitectura 4

Particularidades

Particularidades

Las estaciones de trabajo, el servidor de bases de datos y servidor de Firewall dedicado deben tener tarjetas de red para conectarse a sus redes locales. Los Firewall se ejecutan en servidores

independientes para centralizar las configuraciones y tráfico de la red. Para mantener el servidor web accesible es necesario obtener una dirección IP estática y registro DNS.

En este caso se implementa un dispositivo modulador para conectar la red local con el Internet.

Ventajas:

- Cuenta con conexión a Internet.
- Tiene una gran seguridad.
- Permite acceso remoto a la base de datos.

Desventajas:

- Muy dificultosa de gestionar.
- Muy costosa de implementar.

2.10.8 Selección de la alternativa

Se eligió la alternativa web por su alta compatibilidad entre dispositivos, y la capacidad de proveer al cliente de soporte técnico remotamente, lo cual lo hace mas rapido y facil

2.10.9 Análisis de factibilidad para la alternativa seleccionada

El desarrollo de esta solución requiere de al menos 3 personas, y contamos con 4 personas en nuestro equipo de trabajo, por lo que la falta de personal no sería un impedimento para llevar a cabo este proyecto.

También requiere de un estimado de 3276 horas, lo que equivalen a 819 horas por miembro de equipo, 136 días trabajando diarias, lo que daría un plazo de 4 meses para la entrega del proyecto total.

Contamos con 4 PCs con los programas requeridos instalados en el Sistema.

Por la parte de finanzas, contamos con la financiación completa del proyecto, por lo que por el lado económico no existe ninguna limitación.

Debido a todo lo anteriormente dicho, podemos afirmar que el desarrollo y puesta en funcionamiento de la solución en la empresa cliente es totalmente viable.

2.10.9.1 Operativa

La implementación del sistema en el establecimiento repercutirá beneficiosamente en la preferencia de clientes por el local debido a una mayor agilidad y rapidez a la hora de las interacciones con el cliente y entre el personal, también se podrá ofrecer un acceso al manu mas facil y remota, sin necesidad de estar en el restaurante, de

una manera más cómoda que solo la foto del menú, ya que el usuario podrá ver fotos del plato y acceder a datos más detallados, la inversión en el software se hace solo una vez, aunque podrá requerir mantenimiento, el programa no tiene fecha de vencimiento.

Todo lo anterior va a suponer una gran mejora en la experiencia para el cliente y para los empleados con un software completamente adaptado al funcionamiento del establecimiento y que va a contemplar cada detalle, a diferencia de un software por licencia más genérico, los cuales muchas veces carecen del soporte técnico necesario para una aplicación comercial, en la cual se va a depender de un sistema robusto y confiable para evitar pérdidas de tiempo

2.10.9.2 Técnica.

La implementación desde el lado técnico no va a requerir grandes inversiones, los empleados podrán emplear sus terminales o como sea de preferencia del cliente, el local tendrá que contar con una pc e internet, con lo cual ya cuenta.

El límite en términos de tamaño en bytes de la base de datos es de 5 gb, el tamaño máximo de una fila es de 65.535 bytes, por lo cual, evidentemente, no va a ser nunca necesario buscar otra alternativa, de paga o de cualquier tipo.

2.10.9.3 Legal

El sistema se registrará bajo las normativas de DGI, protección de datos:

Ley N° 18331:

- Toda persona física o jurídica tendrá derecho a solicitar la rectificación, actualización, inclusión o supresión de los datos personales que le corresponda incluidos en una base de datos, al constatar error o falsedad o exclusión en la información de la que es titular.

¿Qué son los datos personales?:

- Un dato personal es cualquier tipo de información que nos pueda identificar directamente o nos hace identificables, ya sea nuestro nombre, dirección, teléfono, cédula de identidad, RUT, huella digital, número de socio, número de estudiante, una fotografía o hasta el ADN.

Ley N° 19.438

Declara que exceptúa a UTE, ANTEL y OSE, de las limitaciones para brindar información a los gobiernos departamentales.

2° de la Ley N° 19.179.

Regulación del formato para el procesamiento y almacenamiento de información digital por parte de determinados organismos y empresas.

2.10.9.4 Económica

El costo del programa va a depender de una estimación, la estimación por puntos de función, en la cual se tendrá en cuenta el tamaño del mismo (líneas de código), para poder aproximarse a un precio.

Para identificar el peso de cada funcionalidad del sistema se utiliza la siguiente tabla la cual determinará la complejidad de cada función según el estándar de la IFPUG(International Function Point Users Group).

Tipo de función de usuario	Nivel de complejidad	Nº	*	Peso	=	Total
Entradas	Baja			3		0
	Media	21		4		84
	Alta			6		0
Salidas	Baja	4		4		16
	Media	7		5		35
	Alta	2		7		14
Consultas	Baja	16		3		48
	Media			4		0
	Alta			6		0
Archivos	Baja	4		7		28
	Media	20		10		200
	Alta	2		15		30
Interfaces	Baja			5		0
	Media			7		0
	Alta			10		0
Número de Puntos Función sin ajustar:						455

FIG41. Cálculo puntos de función

Factores de Influencia en la Dificultad del Sistema	Grado
1. Comunicaciones de datos	1
2. Procesamiento distribuido	0
3. Objetivos de rendimiento	3
4. Configuración de uso intensivo	0
5. Tasas de transacción rápidas	0
6. Entrada de datos en línea	5
7. Amigabilidad en el diseño	4
8. Actualización de datos en línea	0
9. Procesamiento complejo	1
10. Reusabilidad	4
11. Facilidad de instalación	3
12. Facilidad operacional	0
13. Adaptabilidad	0
14. Versatilidad	4
Total GI	25

FIG42. Calculo factores de influencia en dificultad del sistema

PUNTOS FUNCIÓN = PUNTOS FUNCIÓN SIN AJUSTAR * (0'65 + 0'01 * TOTAL GI) =	
409.5	Total puntos
3276	Horas totales
819	Horas por integrantes
10	\$ precio por hora
32760	\$ total precio contando horas totales

FIG43. Calculo puntos de función

Los parámetros indican el tipo de funcionalidad, lo que va a hacer el sistema, cada parámetro tiene 3 niveles de complejidad y estos mismos tienen a su vez un peso el cual es la unidad de medida del esfuerzo, conociendo la cantidad de veces que se repite cada parámetro se calcula el total, que es sumar todo evidentemente, y se multiplica por el peso para obtener la suma total de los puntos por función.

También el local deberá invertir en una computadora, monitor y periféricos, en caso de querer hostear la página de manera local, el local ya cuenta con internet de fibra óptica básico de antel y el precio de la instalación viene incluido en el precio del software por lo cual

esa sería la única inversión fuera del desarrollo, sin tener en cuenta el soporte y mantenimiento.

En caso de querer proveer a sus empleados de equipos para usar el software, tablets por ejemplo, se deberá invertir en eso también, aunque es un gasto que el propietario deberá tomar en consideración

2.11 Gestión de riesgos

2.11.1 Identificación de riesgos

n°	riesgos	categoría	ocurrencia	incidencia
R1	Pérdida de datos	Proyecto	Baja	Alta
R2	Incumplimiento de los tiempos planificados	Proyecto	Media	Media
R3	Baja de algún miembro, temporal o definitiva	Proyecto	Baja	Alta
R4	Baja de cliente	Proyecto	Baja	Media
R5	Disconformidad por parte del cliente	Proyecto	Media	Baja
R6	Falta de tiempo para cumplir requerimientos	Proyecto	Media	Alta
R7	Programar un requerimiento sin una	Proyecto	Alta	Media

	investigación previa			
R8	Cambios importantes en fases finales del proyecto por correcciones	Proyecto	Baja	Alta
R9	Errores en el código	Programación	Alta	Baja
R10	No conexión a internet	Proyecto	Baja	Baja
R11	Dificultades con el S.O instalado	Sistemas operativos	Baja	Medio
r12	Problemas al conectar la base de datos	Base de datos	Baja	Media
R13	Problemas con la documentación	Proyecto	Media	Media
R14	Ausencia de un profesor	Proyecto	Baja	Alta
R15	Cliente no conforme con nuestro diseño	ADA	Baja	Media
R16	Base de datos incompleta o redundante	Base de datos	Media	Media
R17	Incapacidad de completar los requerimientos de código	Programación	Media	Alta

	por falta de conocimiento			
R18	Cambio de lenguaje	Programación	Baja	Alto
R19	Pcs poco potentes	Sistemas operativos	Media	Baja
R20	Condiciones climáticas	Proyecto	media	Baja

2.11.2 Gestión de Riesgos

N°	Riesgo	Disminución	Anulación	Plan de contingencia
R1	Pérdida de datos	Cuidar y organizar los datos	Realizar copias de seguridad	Ejecutar copias de seguridad
R2	Incumplimiento de los tiempos planificados	No dejar para último momento trabajos	Ajustar fechas	Acelerar procesos y tiempos
R3	Baja de algún miembro, temporal o definitiva	Avisar con tiempo	Tratar que pueda seguir con el trabajo de manera virtual	Reestructurar a organizativa
R4	Baja de cliente	Llevarse bien con el equipo de trabajo	Tratar de convencerlo	Conseguir un nuevo cliente
R5	Disconformidad por parte del cliente	Entregar y mostrar parcialmente	Pedir reseñas al cliente	Ajustar y modificar conforme a

			periódicamente y ajustar según su conformidad	las solicitudes
R6	Cambios importantes en fases finales del proyecto	Prever con anticipación	Cambiar y actualizar de a poco	Acelerar procesos y cambios
R7	Programar un requerimiento sin una investigación previa	No realizar un requerimiento de la nada	Realizar el correcto análisis	Actualizar o modificar requerimiento
R8	Hardware defectuoso	Verificar que funcione antes de la ejecución	Tener disponible otros dispositivos	Reparación o cambio de hardware
R9	Errores en el código	Revisiones periódicas	Piezas de código	Reparar el código
R10	No conexión a internet	Verificar conexiones	Tener varias disponibles	Utilizar otra conexión
R11	Dificultades con el S.O instalado	Verificar utilidades y robustez	Tener varias versiones y sistemas disponibles	Actualizar o reparar S.O
R12	Problemas al conectar la base de datos	Realizar con anticipación las conexiones	No anulado	Reconectar a otra base de datos
R13	Problemas con la documentación	Documentar todos los procesos	Mostrar a profesores y ejecutar correcciones.	Modificar y actualizar documentación.
R14	Miembro enfermo	Avisar con anticipación	Reorganizar procesos	Equipo realiza la parte del

				enfermo
R15	Cliente no conforme con nuestro diseño estetico	Mostrar parcialmente	Modificar parcialmente e con sus correcciones	Actualizar diseño
R16	Base de datos incompleta o redundante	Rellenar en base a MER válidos	Realizar los MER de todos los procesos	Rehacer base de datos
R17	Incapacidad de implementar los requerimientos funcionales	Planificar un calendario de implementación lógico en cuanto al tiempo	Racionalizar los requerimientos que deben ir correctamente	Implementar los más importantes
R18	Cambio de lenguaje	Hacer un análisis de lenguajes que nos sirven	Elegir lenguajes sin cambio al inicio del proyecto	Reorganizar en consecuencia al nuevo lenguaje
R19	Falta de hardware en etapa avanzada	Hacer lista de hardware necesario	Prever hardware al inicio del proyecto	Ajustar software a la realidad del hardware
R20	Condiciones climáticas	No disminución	No anulado	Programar horarios y reorganizar

2.12 Plan de proyecto

2.12.1 Definición del proceso

2.12.1.1 Metodología

Elegimos la POO ya que es la más adecuada para nuestro proyecto en el cual nos vamos a encontrar con errores frecuentemente, vamos a trabajar con relativamente muchas clases, con sus correspondientes atributos, métodos y tabla generada en una base de datos, es un paradigma muy utilizado para soluciones parecidas a las de nuestro

proyecto, por lo cual es más fácil encontrar información y nos permite la reutilización de código. Trabajamos con un MVC para independizar el funcionamiento, lo cual nos facilita el mantenimiento en caso de que haya cualquier tipo de error y nos ofrece maneras más sencillas de probar el funcionamiento del sistema.

2.12.1.2 Ciclo de vida elegido

Nuestro software se va a implementar con un esquema de desarrollo incremental iterativo, ya que es lo que mejor se adapta a un proyecto en el cual se exigen entregas periódicas y cada módulo del software tiene más o menos importancia.

El desarrollo del software se va a dividir en iteraciones, en las cuales se va a cumplir con la implementación de ciertos requerimientos funcionales. De esta manera podemos formar una base del software con lo más esencial del problema, el sistema de comandos y un control de usuarios mediante sesiones y postergar para las siguientes interacciones funcionalidades menos importantes del software, o cuestiones estéticas que no cumplan un rol esencial y son menos tediosas de programar.

En cada iteración recibiremos tanto feedback del cliente como de los docentes, por lo cual tenemos un tiempo en el cual podemos planificar cambios o redirigir el desarrollo sin tener la dificultad de cambiar un software ya desarrollado en casi su totalidad, nos va a permitir conocer si es posible la implementación de funcionalidades de acorde al tiempo límite que tenemos.

Dado que cada iteración debe dar como resultado requisitos terminados, se minimiza el número de errores que se producen en el desarrollo y se aumenta la calidad.

2.12.1.3 Integrantes y roles

Los integrantes somos:

- Dylan Rodriguez - Base de datos y programación
- Brian Matos - Documentación
- Diego Pozzi - Diseño y programación
- Facundo Rodriguez - Documentación y gestión empresarial

2.12.1.4 Descripción de las herramientas

- **Draw.io 19.0.2**

Utilidad: Se utiliza para realizar todo lo relacionado a diagramas, bocetos y maquetado.

- **Visual Code Studio 2022**

Utilidad: Editor de código utilizado por todos los integrantes del grupo.

- **Google sheets**

- Utilidad: Estructuración del cronograma.
- **Asana**
Utilidad: Formación de base del cronograma.
- **MySql 8.0.29**
Utilidad: Sistema gestor de base de datos.
- **Windows 10**
Utilidad: Sistema operativo de implementación y de desarrollo.
- **Xampp 8.1.6**
Utilidad: Iniciar base de datos y host local de la página.
- **Documentos de google**
- **Google Drive**
Utilidad: almacenamiento compartido de los archivos del proyecto.

2.12.1.5 Plan de SQA

El proyecto constará con todos los procedimientos para asegurar su calidad.

Dividido en 3 instancias, Análisis, Diseño y Desarrollo.

En el análisis nos encargaremos de recopilar toda la información posible acerca de nuestro cliente y en base a eso analizar el problema en sí. se definirán las necesidades, objetivos, requerimientos y se hará el UML correspondiente.

En el diseño se tendrá en cuenta lo recopilado en el análisis y se hará el diseño de interfaces, el MER para la base de datos y se actualizará el UML agregándole algunos métodos que se tendrán que implementar.

El desarrollo se dividirá en 3 iteraciones las cuales van a ir en orden de importancia en cuanto los requerimientos a implementar, las cuales van a ser a su vez probadas en cada instancia.

Todo lo anterior va a ser gestionado mediante la Gestión de Proyecto, la cual va a contar con los siguientes puntos:

- Plan de Proyecto
- Gestión de Riesgos
- Plan de iteración

Y se va a asegurar de que todos los problemas o desviaciones que surjan sean documentadas.

En cuanto a la documentación, deberá:

- Ser consistente, no pueden haber elementos contradictorios.
- Ser no ambigua, todo término referido al área de aplicación debe estar definido en un glosario.
- Ser verificable, debe ser posible verificar siguiendo un método definido, si el producto final cumple o no con cada requerimiento.

- Estar acompañada de un detalle de los procedimientos adecuados para verificar si el producto cumple o no con los requerimientos.
- Incluir requerimientos de calidad del producto a construir.

Con el fin de que en caso de necesitarlo, un grupo independiente de desarrollo pueda encarar el proyecto o que un miembro del grupo pueda consultar a la documentación y entenderla sin problemas.

Matriz	Dylan	Brian	Diego	Facundo
Desarrollar/Documentar el plan SQA	x	x		
Revisar el plan SQA	x	x		
Aprobar el plan SQA	x	x		
Planificación del proyecto, seguimiento y supervisión de procesos	x			
Proceso de análisis de requerimientos del sistema	x	x	x	x
Desarrolla/Documentar requerimientos del sistema	x	x	x	x
Revisar requerimientos del sistema	x		x	x
Aprobar requerimientos del sistema	x		x	x
Desarrollar/Documentar diseño de software	x		x	
Revisar diseño de software	x		x	
Aprobar diseño de software	x			
Desarrollar/Arreglar código	x	x	x	x
Revisión de código	x			

Matriz	Dylan	Brian	Diego	Facundo
Desarrollar/Documentar el plan SQA	x	x		
Revisar el plan SQA	x	x		
Aprobar el plan SQA	x	x		
Planificación del proyecto, seguimiento y supervisión de procesos	x			
Proceso de análisis de requerimientos del sistema	x	x	x	x
Desarrolla/Documentar requerimientos del sistema	x	x	x	x
Revisar requerimientos del sistema	x		x	x
Aprobar requerimientos del sistema	x		x	x
Desarrollar/Documentar diseño de software	x		x	
Revisar diseño de software	x		x	
Integrar software	x			

2.12.1.5.1 Estándares definidos y convenciones

- **ABML:** ALTA BAJA MODIFICACIÓN LISTADO hace referencia a la implementación de los 4 puntos dentro de un sistema.
- **CRUD:** CREATE READ UPDATE DELETE, lo mismo que lo anterior pero en inglés.
- **DB:** DATABASE, base de datos.
- **Login:** Ingreso de un usuario dentro de un sistema.
- **Lamp:** Es un sistema gestor de base de datos, el de Linux en concreto.
- **MER:** Modelo entidad relación, es un diagrama el cual permite diseñar la base de datos.
- **UML:** Unified Modeling Language, es un estándar genérico definido para el diseño de software.
- **PHP:** Es un lenguaje de programación, aplicado a servidores sobre todo.

- **CSS:** Es un lenguaje de estilo que se usa para definir las propiedades visuales de HTML.
- **HTML:** Es un lenguaje de etiquetas, utilizado para crear páginas web.
- **RF:** Requerimiento funcional.
- **RNF:** Requerimiento no funcional.
- **C.U:** Caso de uso.

Convenciones

- Se va a utilizar la notación de Camell Case, por lo tanto, el primer carácter de todas las palabras se va a escribir en Mayúsculas y los otros caracteres
 - en minúsculas.
- Se van a usar nombres entendibles y descriptivos para nombrar a las variables. Se van a permitir abreviaciones que se puedan interpretar.
- Las variables globales van a ser declaradas con un guión bajo al principio, las locales no.
- No se podrán usar palabras reservadas por el lenguaje de programación.
- Cuando se trate de variables tipo boolean se debe de usar el prefijo "es".
`private bool esValido`
`private bool esActivo` .
- El nombre de los archivos físicos deberá de coincidir con el de la clase.
- Los comentarios deben de estar al mismo nivel que el código
- Las llaves ({}) deben estar en el mismo nivel que el código fuera de las llaves.
- Se usará TAB para las sangrías.
- Se va a usar una línea en blanco como separador para dos grupos lógicos de código y deberá de haber solo una línea en blanco para separar dos métodos dentro de una clase.

2.12.1.5.2 Plan de testing

El testing es fundamental para definir los objetivos de calidad en un desarrollo de sistemas, encargándose de definir aspectos como por ejemplo los módulos o funcionalidades sujeto de verificación, tipos de pruebas, entornos, recursos asignados, entre otros aspectos. Por lo tanto, hemos definido varias estrategias para el testing de nuestro sistema.

Analizar los requerimientos funcionales

Para elaborar un plan de pruebas de software lo primero que debemos hacer es analizar todos los requerimientos funcionales que hemos marcado.

Deberemos analizar toda la información de requisitos, especificaciones y diseño funcional, requisitos no funcionales, casos de uso,

Identificar las funcionalidades nuevas a probar

A partir de la documentación del análisis de requisitos, debemos identificar e incluir en el plan de pruebas de software en la lista de las funcionalidades.

Identificar las funcionalidades de sistemas existentes que deben probarse

Debemos identificar las funcionalidades existentes que están siendo impactadas por el desarrollo de alguna forma, considerando todos los componentes afectados en todas las capas de la arquitectura de software.

Existen dos situaciones que podemos encontrar al identificar estas funcionalidades:

Funcionalidades modificadas de cara al usuario: Por ejemplo, si una funcionalidad está siendo modificada agregando más pantallas o cambios a su flujo de proceso, debe ser incluida en el plan de pruebas de software.

Funcionalidades modificadas en sus componentes internos: Son funcionalidades no modificadas de cara al usuario, manteniendo la misma interfaz gráfica y flujo de procesos, sin embargo, si se modifican componentes internos que comparten con otras funcionalidades del sistema, en las capas de lógica de negocio o acceso a datos.

Estas deben incluirse en el plan de pruebas de software para determinar a partir de ellas pruebas de regresión a realizar.

Definir la estrategia de pruebas

Consiste básicamente en seleccionar cuáles son los tipos de pruebas de software que debemos realizar.

Por ejemplo los tipos de pruebas de software definidos por el ISTQB.

Para esto definimos usar estrategias de caja negra y caja blanca.

Caja negra es el método en el cual el elemento es estudiado desde el punto de vista de las entradas que recibe y las salidas o respuestas

que produce, sin tener en cuenta su funcionamiento interno. Estas pruebas son realizadas desde la interfaz gráfica.

Por otro lado, el método de pruebas White box (caja blanca) es el cual mira el código y la estructura del producto que se va a probar y usa ese conocimiento para la realización de las pruebas.

Este método se usa en la fase de Unit testing, aunque también puede ocurrir en otras fases como en las pruebas de sistema o de integración. Para la ejecución de este método es necesario que el tester o la persona que vaya a usar el método tengan amplios conocimientos de la tecnología y arquitectura usada para desarrollar el programa

Criterios de aceptación o rechazo:

Para definir los criterios de aceptación o rechazo, es necesario definir el nivel de tolerancia a fallos de calidad. Si la tolerancia a fallos es muy baja puede definirse como criterio de aceptación que el 100% de los casos de prueba estén sin incidencias. Lograr este margen en todos los casos de prueba principales y casos bordes será muy difícil, y podría comprometer los plazos del proyecto (incrementa los riesgos), pero asegura la calidad del producto.

Por otra parte, puede ser que la intención sea realizar un Soft Launch, o un mínimo producto viable, en ese caso se podría definir como criterio de aceptación el 100% de los casos de prueba principales (considerados clave) y 20% de casos de prueba no principales (casos borde).

Una vez logradas las condiciones, se darán por aceptadas las pruebas y el desarrollo de software.

Criterios de inicio o reanudación:

Definen las condiciones que deben cumplirse para dar inicio o reanudar las pruebas. Por ejemplo, en el caso de inicio la condición podría ser la instalación de los componentes de software en el ambiente y que los casos de pruebas de verificación de ambiente sean exitosos.

Para el caso de la reanudación las condiciones están relacionadas, se determina a partir de cuáles criterios de suspensión se presentaron para detener las pruebas. Una vez que estas condiciones ya no existan (sean solventadas) procedemos con la reanudación.

Criterios de suspensión:

El criterio de suspensión puede ser poco exigente, por ejemplo solo ocurriendo si se bloquean por incidencia todos los casos de prueba.

2.12.1.6 Plan de SCM

Como medida de seguridad, respaldando los datos, y de organización entre los miembros del equipo, las iteraciones o cambios que se hagan en el software se subiran a GitHub, se utilizaran las siguientes notaciones:

El primer número hace referencia a la primera iteración, tal que 1.XX es la primera iteración, 2.XX la segunda, etc.

El segundo número hace referencia a las versiones del software, tal que 1.5 sería una versión temprana de 2.0.

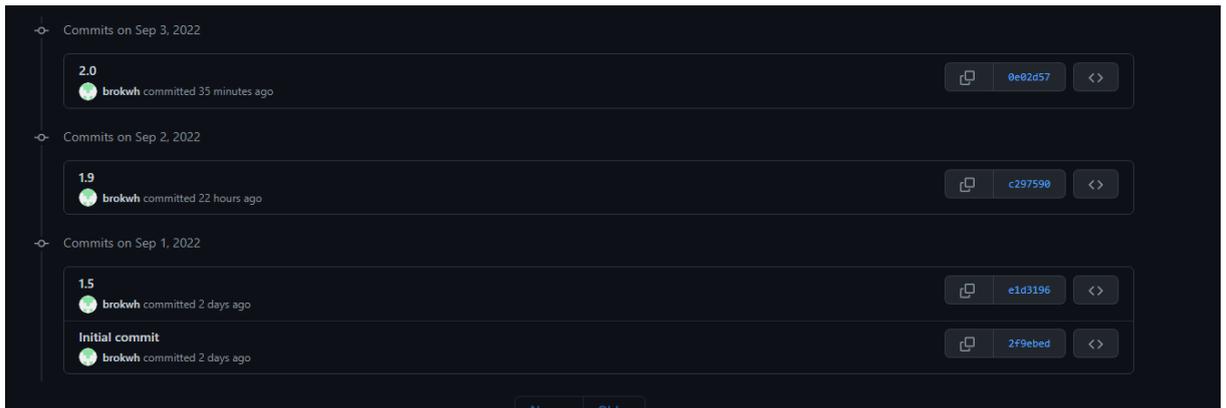


FIG44. *plan SCM*

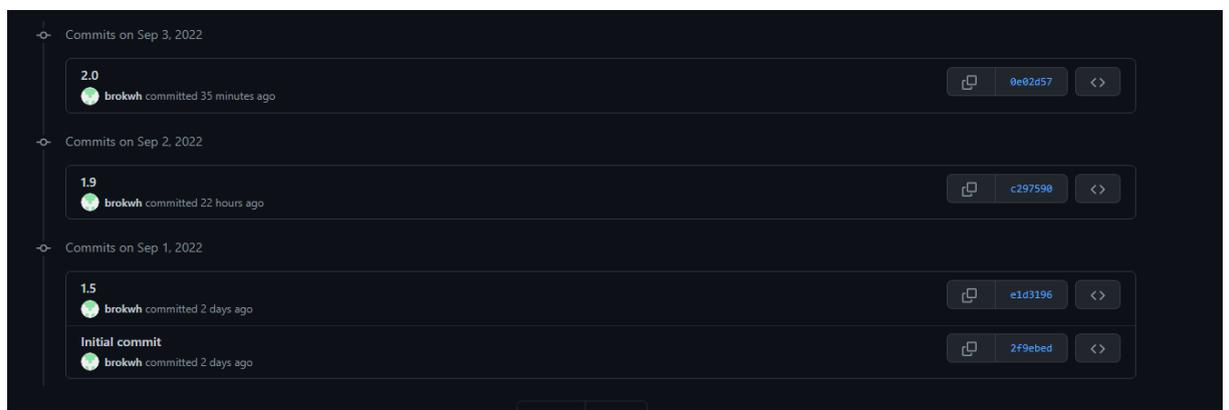


FIG45. *plan SCM*

Informe GIT

El control de variantes, además conocido como "control de código fuente", es la práctica de rastrear y gestionar los cambios en el código de programa. Los sistemas de control de variantes son herramientas de programa que ayudan a los grupos de programa a gestionar los cambios en el código fuente todo el tiempo. Mientras los ámbitos de desarrollo se aceleran, los sistemas de control de variantes ayudan a los grupos de programa a laborar de manera más inmediata e inteligente. Son en especial útiles para los grupos de desarrolladores, debido a que les ayudan a minimizar la época de desarrollo.

El programa de control de variantes ejecuta un seguimiento de cada una de las modificaciones en el código en un tipo particular de base de datos. Si se comete un error, los desarrolladores tienen la posibilidad de ir hacia atrás en la era y equiparar las variantes anteriores del código para contribuir a solucionar el error, al tiempo que se minimizan las interrupciones para todos los miembros del equipo.

A diferencia de ciertos programas de programa de control de variantes, Git no se deja mentir por los nombres de los archivos en el momento de decidir cuál debe ser el almacenamiento y el historial de variantes del árbol de archivos; en vez de ello, se concentra en el contenido del propio documento. En conclusión, los archivos de código fuente cambian de nombre, se parten y se reorganizan a menudo. El formato de objeto de los archivos del repositorio de Git emplea una mezcla de codificación delta (que almacena las diferencias de contenido) y compresión, y guarda explícitamente el contenido de los directorios y los objetos de metadatos de las variantes. Ciertos otros sistemas de control de variantes carecen de custodia contra las modificaciones escondidas llevadas a cabo con posterioridad, algo que puede dar por sentado una grave vulnerabilidad de estabilidad de la información para cualquier organización que se base en el desarrollo de programa.

Git disfruta de una vasta base de usuarios y de un enorme apoyo a causa de la sociedad, al ser la alternativa más conocida, la documentación es mucha, debido a que incluye libros, tutoriales y sitios web especializados, así como podcasts y tutoriales en vídeo.

Por todo lo mencionado nosotros mismos vamos a utilizar github para conservar un orden entre variantes y además de centralizar donde estamos haciendo un trabajo.

2.12.1.7 Plan de capacitación

Se va a explicar a cada uno de los trabajadores la forma en la que interactúa con el software, especificar que el sistema será intuitivo y fácil de utilizar. Se capacitará a los empleados del restaurantes en una modalidad presencial en la cual se expondrá el uso del software en un en un escenario simulando una jornada laboral habitual del local en la cual se pondrán y se explicaran todos los casos de uso del software, a su vez se le entregará al personal un manual en formato

PDF el cual podrá ser entregado a cada uno de los trabajadores como y cuando lo desee el propietario.

2.12.1.8 Plan de implementación

2.12.1.8.1 Iteración 1 (AGOSTO)

- Interfaz definida - Semana 1
- Base de datos - Semana 1
- Login - Semana 1
 - Requerimientos a implementar
 - Inicio de sesión
- ABML usuarios - Semana 2
 - Requerimientos a implementar
 - Listado usuarios
 - Alta usuarios
 - Baja usuarios
 - Modificar usuarios
- ABML productos - Semana 3
 - Requerimientos a implementar
 - Listado productos
 - Alta productos
 - Baja productos
 - Modificar productos

2.12.1.8.2 Iteración 2 (SEPTIEMBRE)

- **ABML comandas - Semana 1**
 - Requerimientos a implementar
 - Listado productos
 - Alta productos
 - Baja productos
 - Modificar productos
- **Gestión de mesas - Semana 2 y 3**
 - Requerimientos a implementar
 - Listado productos
 - Alta productos
 - Baja productos
 - Modificar productos
- **Reservas - Semana 4**
 - Requerimientos a implementar
 - Listado productos
 - Alta productos
 - Baja productos
 - Modificar productos
- **Menú - Semana 4**
 - Requerimientos a implementar
 - Listado productos
 - Alta productos
 - Baja productos
 - Modificar productos

2.12.1.8.3 Iteración 3 (OCTUBRE)

- cliente, personal, estacion ,pago, MetodoPago
- Rectificación - Semana 1 y 2, 3
- Implementación - Semana 4

2.12.1.9 Plan de implementación de SO

2.12.1.9.1 Hardware base

Al ser un sistema de bajo nivel, en lo que a almacenar y procesar información concierne, no se va a requerir un hardware muy potente, con casi cualquier PC de gama baja se podrá hostear tanto la página como la base de datos.

Un ejemplo de PC de gama baja sería:

- Un i3 de cualquier generación
- 4gb de RAM
- Motherboard acorde
- Fuente confiable
- Disco de 128gb

No se va a requerir tarjeta gráfica dedicada, al no trabajar con nada visual o de diseño.

2.12.1.9.2 Configuración del BIOS setup

En el BIOS se verificarán las siguientes características:

- Que cumpla con los requisitos mínimos para instalar el SO
- Que la temperatura esté dentro de un rango aceptable
- Que el fan del CPU no se encuentre en modo automático
- Que los voltajes de la motherboard y del CPU sean aceptables
- Que los discos estén en buen estado y tengan buen funcionamiento

Luego de las verificaciones se pasará a configurar el booteo, indicando la iniciación desde el pendrive en donde se encuentra la imagen ISO.

2.12.1.9.3 Instalación del SO servidor LAMP

Utilizaremos un servidor LAMP para un sistema operativo Linux ya que ofrece una mayor estabilidad, seguridad y flexibilidad ya que los servidores de Linux están por delante de los servidores de Windows por lo que Linux permite optimizar el uso de todos los recursos y se pueden adaptar de una manera precisa a las necesidades y preferencias de cualquier tipo de compañía. Generalmente pueden pasar años sin que aparezcan fallos, también permite manejar con fluidez un mayor número de procesos. Al igual que la gran libertad de uso que ofrece Linux por lo que podemos modificar y utilizar todo lo que queramos para satisfacer nuestras necesidades. La aplicación que desarrollaremos será web.



FIG46. código QR video instalación de lamp.

Según el perfilamiento de los usuarios tenemos: gerente, administrador, mozo, delivery, cajero, cocina.

Gerente: Se encarga de dirigir, gestionar y administrar el restaurante. Va a poder modificar los menús, productos y órdenes.

Administrador: Es el usuario con más permisos que va a ser el encargado de gestionar los usuarios y los permisos

Cajero: Registra y controla todos los métodos de pagos que se efectúan, a su vez puede generar comandas.

Mozo: Genera comandas y modifica el estado de las mesas.

Cocina: Puede ver las comandas.

2.12.1.9.4 Plan de seguridad

Para la seguridad de nuestro software utilizamos un login para limitar el acceso entre los usuarios, también utilizamos sistemas de respaldo GITHUB, discos duros y pendrive para tener el mayor nivel de seguridad a nivel de respaldo.

Como medida de seguridad, respaldando los datos, y de organización entre los miembros del equipo, las iteraciones o cambios que se hagan en el software se subiran a GitHub, se utilizaran las siguientes notaciones:

El primer número hace referencia a la primera iteración, tal que 1.XX es la primera iteración, 2.XX la segunda, etc.

El segundo número hace referencia a las versiones del software, tal que 1.5 sería una versión temprana de 2.0.

2.12.1.9.5 Instalación del SO servidor red

```
#!/bin/bash
```

```
echo -e "\n\nActualizando paquetes\n"  
sudo apt-get update -y && sudo apt-get upgrade -y
```

```
echo -e "\n\n Apache2 Web server\n"  
sudo apt-get install apache2 apache2-doc apache2-mpm-prefork  
apache2-utils libexpat1 ssl-cert -y
```

```
echo -e "\n\nInstalando PHP & Requerimientos\n"  
sudo apt-get install libapache2-mod-php7.0 php7.0 php7.0-common  
php7.0-curl php7.0-dev php7.0-gd php-pear php7.0-mcrypt  
php7.0-mysql -y
```

```
echo -e "\n\nInstalando MySQL\n"  
sudo apt-get install mysql-server mysql-client -y
```

```
echo -e "\n\nPermisos para /var/www\n"  
sudo chown -R www-data:www-data /var/www  
echo -e "\n\n Permissions have been set\n"
```

```
echo -e "\n\nHabilitando módulos\n"  
sudo a2enmod rewrite  
sudo phpenmod mcrypt
```

```
echo -e "\n\nReiniciando Apache\n"  
sudo service apache2 restart
```

```
echo -e "\n\nLAMP instalado!"
```

```
exit 0
```

2.12.1.9.6 Servicios

Un servicio es un programa que se ejecuta en segundo plano, fuera del control interactivo de los usuarios del sistema, ya que carecen de una interfaz. Esto con el fin de proporcionar aún más seguridad, pues algunos de estos servicios son cruciales para el funcionamiento del sistema operativo.

Por otro lado, en sistemas como Unix o Linux, los servicios también se conocen como daemons. A veces el nombre de estos servicios, o daemons, terminan con la letra d. Por ejemplo, sshd es el nombre del servicio que maneja SSH.

- **apache2:**
La función esencial del servidor Apache es servir las webs alojadas en el servidor a los diversos navegadores como Chrome, Firefox, Safari,...

Apache consigue que la comunicación entre el servidor web y el cliente web (usuario que solicita la información) sea fluida y constante. Haciendo que cuando un usuario haga una petición HTTP a través de navegador para entrar a una web o URL específica, Apache devuelva la información solicitada a través del protocolo HTTP.

En Apache podemos aplicar una alta personalización a través de su sistema modular, de forma que podemos activar o desactivar diversas funcionalidades a través de los módulos de Apache.

Estos módulos de Apache hay que usarlos con cautela ya que pueden afectar a la seguridad y funcionalidades del servidor web.

Todos los parámetros que se incluyan en `httpd.conf` serán validados previo arranque de Apache, esto es, al ejecutar `apachectl start` (o `apachectl restart`); la única excepción a esto son los renglones que inicien con el signo: `#`, ya que estos indican un comentario (al igual que otros archivos en sistemas `*nix`).

El registro de errores del servidor, cuyo nombre y ubicación se especifica en la directiva `ErrorLog`, es el más importante de todos los registros. Apache enviará cualquier información de diagnóstico y registrará cualquier error que encuentre al procesar peticiones al archivo de registro seleccionado. Es el primer lugar donde tiene que mirar cuando surja un problema al iniciar el servidor o durante su operación normal, porque con frecuencia encontrará en él información detallada de qué ha ido mal y cómo solucionar el problema.

mySQL:MySQL es el sistema de gestión de bases de datos relacional más extendido en la actualidad al estar basada en código abierto. Desarrollado originalmente por MySQL AB, fue adquirida por Sun Microsystems en 2008 y esta su vez comprada por Oracle Corporation en 2010, la cual ya era dueña de un motor propio InnoDB para MySQL.

MySQL es un sistema de gestión de bases de datos que cuenta con una doble licencia. Por una parte es de código abierto, pero por otra, cuenta con una versión comercial gestionada por la compañía Oracle.

deberemos saber dónde encontrar el fichero de configuración. Este fichero en Linux se suele encontrar en `/etc/mysql/my.conf` aunque dependerá de tu distribución, lo más común será encontrarlo en esta ubicación.

El motor de SQL Server registra en el archivo `/var/opt/mssql/log/errorlog` en las instalaciones de Linux y Docker. Debe estar en modo 'superusuario' para navegar por este directorio.

- **PHP:**
PHP es un lenguaje de scripting de código abierto utilizado principalmente para el desarrollo web y el scripting del lado del servidor (backend). En términos sencillos:
 - PHP puede instalarse en un servidor para ejecutar scripts (por ejemplo, piezas de código para añadir formularios a tu sitio).
 - Se ejecuta del lado del servidor y no del lado del cliente, por lo que funciona en el backend y no en el navegador.
 - Se integra bien con HTML, lo que lo hace muy adecuado para el desarrollo web.

El fichero de configuración (php.ini) es leído al arrancar PHP. En las versiones en que PHP funciona como módulo de servidor, esto sucede únicamente cuando se inicia el servidor. En las versiones CGI y CLI, esto ocurre en cada ejecución.

Podemos usar el comando `php --info` para localizar la ubicación del archivo del registro de errores en PHP en el módulo de apache. Para Windows, use el comando `php --info | findstr /r /c:"error_log"` comando en el símbolo del sistema. En el caso de Linux, use el comando `php --info | grep error`. Estos comandos generarán la ubicación del registro de errores en la terminal.

PHPmyadmin:PhpMyAdmin es una aplicación web que sirve para administrar bases de datos MySQL de forma sencilla y con una interfaz amistosa. Se trata de un software muy popular basado en PHP.

Todos los datos configurables se colocan en `config.inc.php` en el directorio de nivel superior de phpMyAdmin. Si este archivo no existe, consulte la sección Instalación para crear uno. Este archivo solo necesita contener los parámetros que desea cambiar de su valor predeterminado correspondiente en `library/config.default.php` (este archivo no está diseñado para cambios).

2.12.1.9.7 Shell script administrador

-backup base de datos:

```
#!/bin/sh
```

```
#
```

```
#Realizar backup de mysql como usuario root, a todas la bases de  
datos existentes, y guardar la salida en el archivo  
/respaldos/respaldo_db.sql.
```

```
mysqlDump -uroot -p[password] --all-databases >  
/respaldos/respaldo_db.sql
```

```
#
#Generar un archivo tar del archivo respaldo_db.sql.
tar -zcvf /respaldos/respaldo_db_$(date +%d%m%y).tgz
/respaldos/*.sql
#
#
# Directorios y/o archivos que queremos respaldar.
backup_files="/home /var/www /etc/apache2 /etc/bind"
#
# Directorio en el cual queremos guardar nuestros respaldos.
dest="/respaldos"
#
# creando un archivo tar para respaldar.
day=$(date +%d%m%y)
hostname=$(hostname -s)
#Este archivo llevara el nombre del equipo seguido de la fecha en la
cual se realiza el respaldo.
archive_file="$hostname-$day.tgz"
#
# Imprimir mensajes de estado
echo "Respaldando $backup_files a $dest/$archive_file"
date
echo
#
# Respaldando los archivos con tar
tar czf $dest/$archive_file $backup_files /respaldos/respaldo_db_*
#
# Mensajes de estado
echo
echo "Respaldo finalizado"
date
-hacer un respaldo automático:
#sudo crontab -e
# m h dom mon dow command
15 2 * * * /home/aline/respaldo.sh respaldo
-respaldo de configuración que mana un archivo al correo
#/sbin/bash

# ::: == Variables == :::

export FECHA=`date +%d/%m/%Y\::[%X]`; # Fecha en Formato
dia/mes/DD
export LOGNOM="back-dhcp.log" # Nombre fichero LOG
```

```
export LOGDIR="/tmp"; # Directorio Temporal de trabajo

# Correo del Administrador
export CORREO_ADM="Administrador@dominio.com.mx"

# Directorio de configuración
export LDHCP="/etc/dhcp3";

#####
##### PROCEDIMIENTOS #####
#####

#####
### Comprime y compacta el fichero #####
#####

compactar()
{
## Se compacta el fichero a respaldar para enviarlo como correo
adjunto
#
# Si existe se elimina
if [ -e $LOGDIR/dhcpd.tar.gz ];
then
/bin/rm -rf $LOGDIR/dhcpd.tar.gz
fi
#
echo " ";
echo "----- ";
echo "Se envia adjunto del fichero de configuración. Favor de
Verificar. Gracias"
## Se compacta en directorio temporal

/bin/tar czvf $LOGDIR/dhcpd.tar.gz $LDHCP/dhcpd.conf

##
} ##EOF

#####
##Envio el LOG y fichero compresado por correo ##
#####

correo()
{
```

```
## Envío de Correo Con Adjunto incluido

cat $LOGDIR/$LOGNOM | mutt -s "Respaldo de Configuración
DHCP" ${CORREO_ADM} -a $LOGDIR/dhcpd.tar.gz

} ##EOF

#####
### P R I N C I P A L ###
#####

echo "*" Reporte Generado el dia: ${FECHA} " >>
$LOGDIR/$LOGNOM
echo " " >> $LOGDIR/$LOGNOM

## Llamado al procedimiento para comprimir
compactar >> $LOGDIR/$LOGNOM

## Envio del correo LOG y Adjunto:
correo > /dev/null 2>&1 || :

exit 0
```

2.13 Cronograma de trabajo

Name	Materia	Responsable	Fecha inicio	Fecha entrega	Duración	Estado
• Normas de trabajo del equipo	General	D.Rodriguez, B.Matos, D.Pozzi	sáb 30/4/22	dom 1/5/22	2 días	hecho
• Bitácora y Acta de Reunión.	General	D.Rodriguez, B.Matos, D.Pozzi	mar 3/5/22	mié 4/5/22	2 días	hecho
• Inscripciones al equipo(nota de conformación de equipo)	General	Dylan Rodriguez	mié 4/5/22	jue 5/5/22	2 días	hecho
• Nombre de la Empresa y Logo.	General	Facundo Rodriguez	jue 5/5/22	vie 6/5/22	2 días	hecho
• Emplazamiento y	Formación	D.Rodriguez	vie 6/5/22	lun	2 días	hecho

ubicación geográfica	empresarial			9/5/22		
• Ramo o actividad principal	Formación empresarial	B.Matos, D.Pozzi, F.Rodriguez	sáb 7/5/22	lun 9/5/22	2 días	hecho
• Definición misión, visión, objetivos y valores de la organización	Formación empresarial	Facundo Rodriguez	dom 8/5/22	lun 9/5/22	2 días	hecho
• Forma jurídica (contrato social según tipo de sociedad)	Formación empresarial	B.Matos, D.Pozzi, F.Rodriguez	lun 9/5/22	mar 10/5/22	2 días	hecho
• Plan de proyecto	Gestión de proyecto	Todos	mar 24/5/22	mié 25/5/22	2 días	hecho
• Cronograma de trabajo	General	D.Rodriguez, B.Matos, D.Pozzi	vie 27/5/22	lun 30/5/22	2 días	hecho
• Descripción del cliente	Análisis y diseño de aplicaciones	Todos	lun 30/5/22	mar 31/5/22	2 días	hecho
• Descripción del problema (Identificación de al menos 10 procesos)	Análisis y diseño de aplicaciones	Todos	jue 2/6/22	vie 3/6/22	2 días	hecho
• Diagramas de actividad de los procesos identificados.	Análisis y diseño de aplicaciones	Todos	dom 5/6/22	lun 6/6/22	2 días	hecho
• Actores involucrados	Análisis y diseño de aplicaciones	Todos	mié 8/6/22	jue 9/6/22	2 días	hecho
• Descripción del entorno (Modelo conceptual)	Análisis y diseño de aplicaciones	Todos	sáb 11/6/22	lun 13/6/22	2 días	hecho
• Lista de necesidades (del cliente)	Análisis y diseño de aplicaciones	Todos	mar 14/6/22	mié 15/6/22	2 días	hecho
• Alcance y limitaciones	Análisis y diseño de aplicaciones	Todos	vie 17/6/22	lun 20/6/22	2 días	hecho

• Lista de requerimientos (funcionales y no funcionales)	Análisis y diseño de aplicaciones	Todos	lun 20/6/22	mar 21/6/2 2	2 días	hecho
• Análisis de los sistemas operativos a seleccionar en terminales y servidores.	Sistemas operativos	D.Pozzi	jue 23/6/22	vie 24/6/2 2	2 días	hecho
• Justificación técnica de la elección del SO de los servidores.	Sistemas operativos	D.Pozzi, B.Matos	dom 26/6/22	lun 27/6/2 2	2 días	hecho
• Carátula del Equipo	General	Facundo Rodriguez	mié 29/6/22	jue 30/6/2 2	2 días	hecho
• Estudio de alternativas	Sistemas operativos	D.Pozzi	sáb 2/7/22	lun 4/7/22	2 días	hecho
• Maquetación de las interfaces dados los requerimientos dados.	Programación web	D.Rodriguez, D.Pozzi	mar 5/7/22	mié 6/7/22	2 días	hecho
• Implementación del formulario de LOGIN y Menú Principal	Programación web	D.Pozzi, F.Rodriguez, D.Rodriguez	vie 8/7/22	lun 11/7/22	2 días	hecho
• Gestión de riesgos.	Gestión de proyecto	D.Rodriguez	lun 11/7/22	mar 12/7/2 2	2 días	hecho
• Trámites necesarios en los diferentes organismos estatales para su formación	Formación empresarial	D.Rodriguez, B.Matos	jue 14/7/22	vie 15/7/2 2	2 días	hecho
• Trámites específicos según actividad	Formación empresarial	Todos	dom 17/7/22	lun 18/7/2 2	2 días	hecho
• Identificación y descripción del entorno de la	Formación empresarial	Todos	mié 20/7/22	jue 21/7/2 2	2 días	hecho

empresa						
• Objetivos (como empresa que desarrolla el SW)	Análisis y diseño de aplicaciones	Todos	sáb 23/7/22	lun 25/7/22 2	2 días	hecho
• Matrices necesidades-objetivos y objetivos-requerimientos	Análisis y diseño de aplicaciones	Todos	mar 26/7/22	mié 27/7/22 2	2 días	hecho

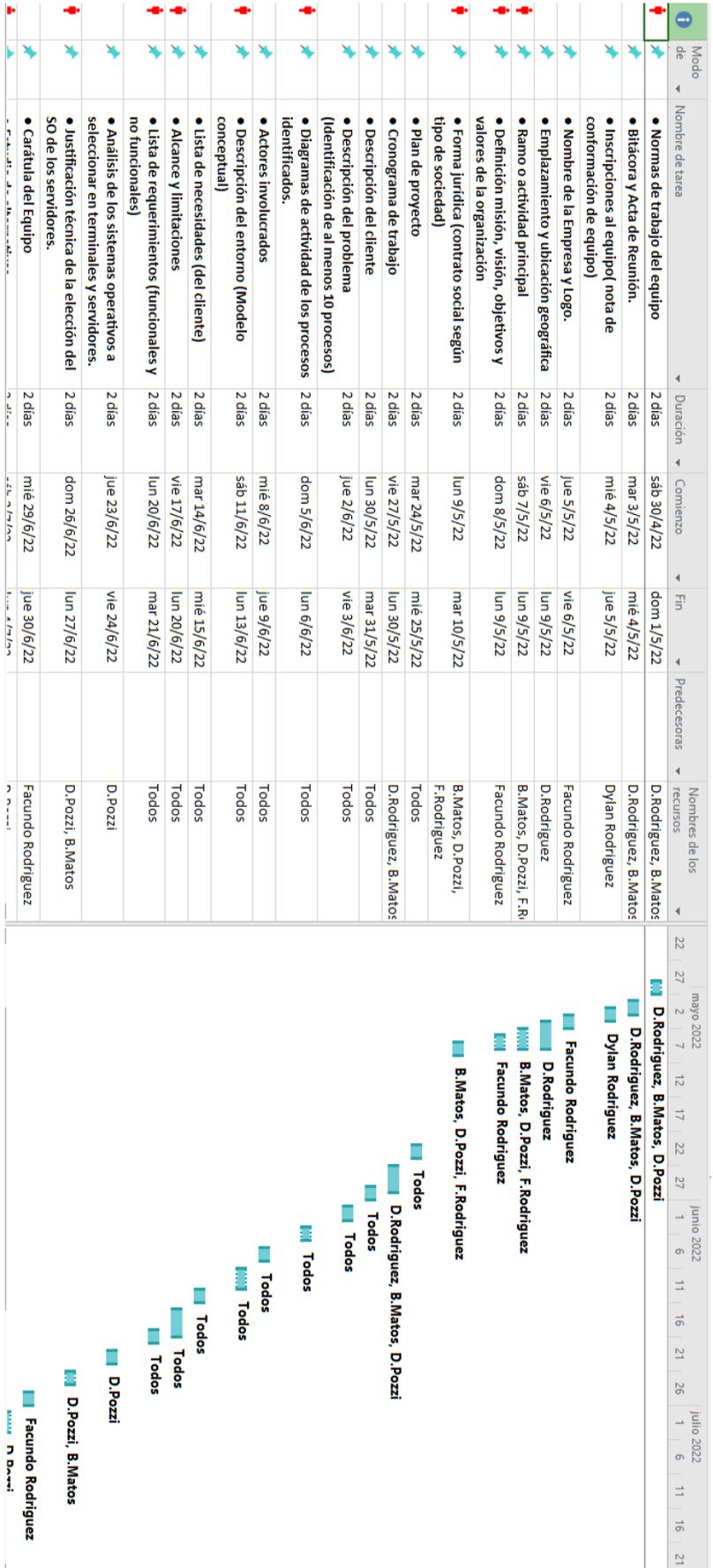


FIG47. diagrama de Gantt cronograma 1(parte 1)



FIG48. diagrama de Gantt cronograma 1(parte 2)

Planificación segunda entrega:

Nombre	Materia	Responsable	Fecha inicio	Fecha entrega	Duración	Estado
• Listado de todos los archivos de la página	programación web	D.Rodriguez, F. Rodriguez	20/7	21/7	1 día	hecho
• Página principal y categorías	programación web	D.Rodriguez	21/7	22/7	1 día	hecho
• Control de accesos implementado	programación web	D.Rodriguez, F. Rodriguez	22/7	23/7	1 día	hecho
• Implementación C.R.U.D usuarios y platos	programación web	D.Rodriguez	23/7	24/7	1 día	hecho
• Informe referente a hosting y dominio web	programación web	B.Matos	24/7	25/7	1 día	hecho
• Justificación de plataformas y programas	Sistemas operativos	F. Rodriguez	25/7	26/7	1 día	hecho
• Instalación del servidor LAMP	Sistemas operativos	D.Pozzi	26/7	27/7	1 día	hecho
• Configuración de los servicios	Sistemas operativos	B.Matos, F. Rodriguez	27/7	28/7	1 día	hecho
• Determinar las cuentas de usuario según perfilamiento	Sistemas operativos	D.Rodriguez	28/7	29/7	1 día	hecho
• Respaldo y recuperación de datos	Sistemas operativos	B.Matos	29/7	30/7	1 día	hecho
• Seguridad	Sistemas operativos	B.Matos	30/7	31/7	1 día	hecho
• MER corregido	Base de datos	D.Pozzi, F. Rodriguez	31/7	1/8	1 día	hecho
• MER normalizado	Base de datos	B.Matos	1/8	2/8	1 día	hecho
• Borrador de instalación de base de datos	Base de datos	D.Rodriguez	2/8	3/8	1 día	hecho
• Archivo con inserción de datos de pruebas	Base de datos	B.Matos	3/8	4/8	1 día	hecho

mínimos para las tablas creadas						
• Corrección de posibles errores en etiquetas html	Diseño web	B.Matos, F. Rodriguez	4/8	5/8	1 día	hecho
• Optimización de código html	Diseño web	D.Pozzi	5/8	6/8	1 día	hecho
• Relacionar diferentes tipos de colores para una interfaz de mayor calidad	Diseño web	B.Matos, F. Rodriguez	6/8	7/8	1 día	hecho
• Edición de imagen para la personalización de logos ,iconos ,fondos, etc	Diseño web	D.Rodriguez, F. Rodriguez	7/8	8/8	1 día	hecho
• Diseño responsivo para diferentes dispositivos	Diseño web	F. Rodriguez	8/8	9/8	1 día	hecho
• Diseño en css para cada apartado necesario	Diseño web	B.Matos	9/8	10/8	1 día	hecho
• Identificación y descripción del entorno de la empresa	formación	D.Pozzi, F. Rodriguez	10/8	11/8	1 día	hecho
• Estudio de mercado	formación	B.Matos, F. Rodriguez	11/8	12/8	1 día	hecho
• Mercado objetivo	formación	D.Rodriguez	12/8	13/8	1 día	hecho
• Análisis de la competencia	formación	B.Matos	13/8	14/8	1 día	hecho
• Situación de competitiva actual	formación	B.Matos, F. Rodriguez	14/8	15/8	1 día	hecho
• Justificación de alternativa seleccionada en s.o	A.D.A	D.Pozzi	15/8	16/8	1 día	hecho
• Análisis de factibilidad 1	A.D.A	B.Matos	16/8	17/8	1 día	hecho
• Análisis de factibilidad 2	A.D.A	D.Rodriguez	17/8	18/8	1 día	hecho
• Metodología	A.D.A	B.Matos, F. Rodriguez	18/8	19/8	1 día	hecho

● Ciclo de vida elegido	A.D.A	B.Matos	19/8	20/8	1 día	hecho
● Integrantes y roles	A.D.A	D.Pozzi	20/8	21/8	1 día	hecho
● Descripción de herramientas	A.D.A	B.Matos	21/8	22/8	1 día	hecho
● Plan SQA	A.D.A	D.Rodriguez	22/8	23/8	1 día	hecho
● Estándares definidos y convenciones	A.D.A	B.Matos	23/8	24/8	1 día	hecho
● Plan de testing	A.D.A	F. Rodriguez	24/8	25/8	1 día	hecho
● Plan SCM e informe sobre GIT	A.D.A	D.Pozzi	25/8	26/8	1 día	hecho
● Plan de capacitación	A.D.A	B.Matos	26/8	27/8	1 día	hecho
● Plan de implementación	A.D.A	D.Rodriguez	27/8	28/8	1 día	hecho
● Acta de reunión actualizada	A.D.A	B.Matos, F. Rodriguez	28/8	29/8	1 día	hecho
● Bitácora actualizada	A.D.A	B.Matos, F. Rodriguez	29/8	30/8	1 día	hecho
● Cronograma actualizado	A.D.A	D.Pozzi	30/8	31/8	1 día	hecho
● Revisión de los puntos de la primera entrega	A.D.A	B.Matos, F. Rodriguez	31/8	1/9	1 día	hecho



FIG49. diagrama de Gantt cronograma 2(parte 1)

● metodología	1 día	jue 18/8/22	jue 18/8/22	B.Matos
● ciclo de vida elegido	2 días	vie 19/8/22	lun 22/8/22	B.Matos
● integrantes y roles	2 días	sáb 20/8/22	lun 22/8/22	D.Pozzi
● descripción de herramientas	2 días	dom 21/8/22	lun 22/8/22	B.Matos
● plan SQA	2 días	lun 22/8/22	mar 23/8/22	D.Rodriguez
● estándares definidos y convenciones	2 días	mar 23/8/22	mié 24/8/22	B.Matos
● plan de testing	2 días	mié 24/8/22	jue 25/8/22	B.Matos
● plan SCM e informe sobre GIT	2 días	jue 25/8/22	vie 26/8/22	D.Pozzi
● plan de capacitación	2 días	vie 26/8/22	lun 29/8/22	B.Matos
● plan de implementación	2 días	sáb 27/8/22	lun 29/8/22	D.Rodriguez
● Acta de reunion actualizada	2 días	dom 28/8/22	lun 29/8/22	B.Matos
● bitacora actualizada	2 días	lun 29/8/22	mar 30/8/22	B.Matos
● cronograma actualizado	2 días	mar 30/8/22	mié 31/8/22	D.Pozzi
● revision de los puntos de la primera entrega	2 días	mié 31/8/22	jue 1/9/22	B.Matos

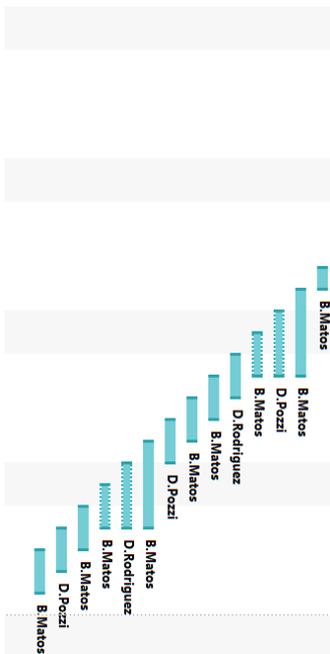


FIG50. diagrama de Gantt cronograma 2(parte 2)

Planificación tercera entrega:

Name	Materia	Responsable	Fecha inicio	Fecha entrega	Duración	Estado
• Plan de Marketing. Mezcla Comercial	Formación empresarial	F. Rodriguez	jue 29/9/22	jue 29/9/22	1 día	hecho
• Plan de inversiones (recursos necesarios para puesta en marcha: materiales y tecnológicos)	Formación empresarial	F. Rodriguez	vie 30/9/22	vie 30/9/22	1 día	hecho
• Plan de Recursos:	Formación empresarial	F. Rodriguez	sáb 1/10/22	sáb 1/10/22	1 día	hecho
• Casos de uso	Análisis y diseño de aplicaciones	D.Rodriguez	dom 2/10/22	dom 2/10/22	1 día	hecho
• Manual	Análisis y diseño de aplicaciones	F. Rodriguez	lun 3/10/22	lun 3/10/22	1 día	hecho
• Corrección de los manuales Técnico y de Usuarios.	Análisis y diseño de aplicaciones	B.Matos	mar 4/10/22	mar 4/10/22	1 día	hecho
• Pruebas	Análisis y diseño de aplicaciones	D.Pozzi	mié 5/10/22	mié 5/10/22	1 día	hecho
• Logros	Análisis y diseño de aplicaciones	F. Rodriguez	jue 6/10/22	jue 6/10/22	1 día	hecho
• Glosario	Análisis y diseño de aplicaciones	D.Rodriguez	vie 7/10/22	vie 7/10/22	1 día	hecho
• Bibliografía	Análisis y diseño de aplicaciones	B.Matos	sáb 8/10/22	sáb 8/10/22	1 día	hecho
• Revisión Final de la documentación: Formato, índices, nombre de imágenes, todo	Proyecto	F. Rodriguez	dom 9/10/22	dom 9/10/22	1 día	hecho
• Cronograma 3a Entrega	Proyecto	D.Pozzi	lun 10/10/22	lun 10/10/22	1 día	hecho
• Gestión de la iteraciones 1-2-3	Proyecto	B.Matos	mar 11/10/22	mar 11/10/22	1 día	hecho
• Cierre	Proyecto	D.Rodriguez	mié 12/10/22	mié 12/10/22	1 día	hecho

<ul style="list-style-type: none"> Control de acceso mejorado(implementación de seguridad con uso de contraseña encriptada en formularios y Base de Datos). 	Programación Web	B.Matos	jue 13/10/22	jue 13/10/22	1 día	hecho
<ul style="list-style-type: none"> GITHUB y acceso al sitio web mediante hosting contratado. (usuario y contraseña) 	Programación Web	F. Rodriguez	vie 14/10/22	vie 14/10/22	1 día	hecho
<ul style="list-style-type: none"> Análisis y recomendación de plan de hosting para el cliente en base a las características de su sitio. 	Programación Web	F. Rodriguez	sáb 15/10/22	sáb 15/10/22	1 día	hecho
<ul style="list-style-type: none"> Sitio web completo (enlace a Drive de los archivos). 	Programación Web	B.Matos	dom 16/10/22	dom 16/10/22	1 día	hecho
<ul style="list-style-type: none"> Implementación C.R.U.D. + MVC restantes. 	Programación Web	D.Rodriguez	lun 17/10/22	lun 17/10/22	1 día	hecho
<ul style="list-style-type: none"> Corrección y agregados de la entrega anterior 	Sistemas operativos	B.Matos, F. Rodriguez	mar 18/10/22	mar 18/10/22	1 día	hecho
<ul style="list-style-type: none"> Shell script para administrar el servidor 	Sistemas operativos	B.Matos	mié 19/10/22	mié 19/10/22	1 día	hecho
<ul style="list-style-type: none"> Informe de los procedimientos por servicio con breve manual descriptivo 	Sistemas operativos	D.Pozzi	jue 20/10/22	jue 20/10/22	1 día	hecho
<ul style="list-style-type: none"> Informe pruebas de funcionamiento y conectividad. 	Sistemas operativos	B.Matos	vie 21/10/22	vie 21/10/22	1 día	hecho
<ul style="list-style-type: none"> Prueba de servicio 	Sistemas operativos	D.Rodriguez	sáb 22/10/22	sáb 22/10/22	1 día	hecho
<ul style="list-style-type: none"> MER, MR y Diccionario de Datos definitivos 	Base de datos II	B.Matos	dom 23/10/22	dom 23/10/22	1 día	hecho
<ul style="list-style-type: none"> Implementación final de la base de datos (instalador, con datos precargados) 	Base de datos II	F. Rodriguez	lun 24/10/22	lun 24/10/22	1 día	hecho

• Listado de las consultas en función de los requerimientos especificados (ver con docentes)	Base de datos II	D.Pozzi	mar 25/10/22	mar 25/10/22	1 día	hecho
• Corrección de observaciones anteriores	Base de datos II	B.Matos, F. Rodriguez	mié 26/10/22	mié 26/10/22	1 día	hecho
• Datos de prueba en las tablas	Base de datos II	D.Rodriguez	jue 27/10/22	jue 27/10/22	1 día	hecho
• Correcciones respecto al diseño relacionado a la entrega anterior	Diseño web	B.Matos	vie 28/10/22	vie 28/10/22	1 día	hecho
• Implementación del HTML restante y aplicar diseño (CSS).	Diseño web	B.Matos	sáb 29/10/22	sáb 29/10/22	1 día	hecho
• Responsividad de la página web en todos los casos que sean necesarios screen) (@media	Diseño web	D.Pozzi, F. Rodriguez	dom 30/10/22	dom 30/10/22	1 día	hecho
• JS	Diseño web	B.Matos	dom 30/10/22	dom 30/10/22	1 día	hecho

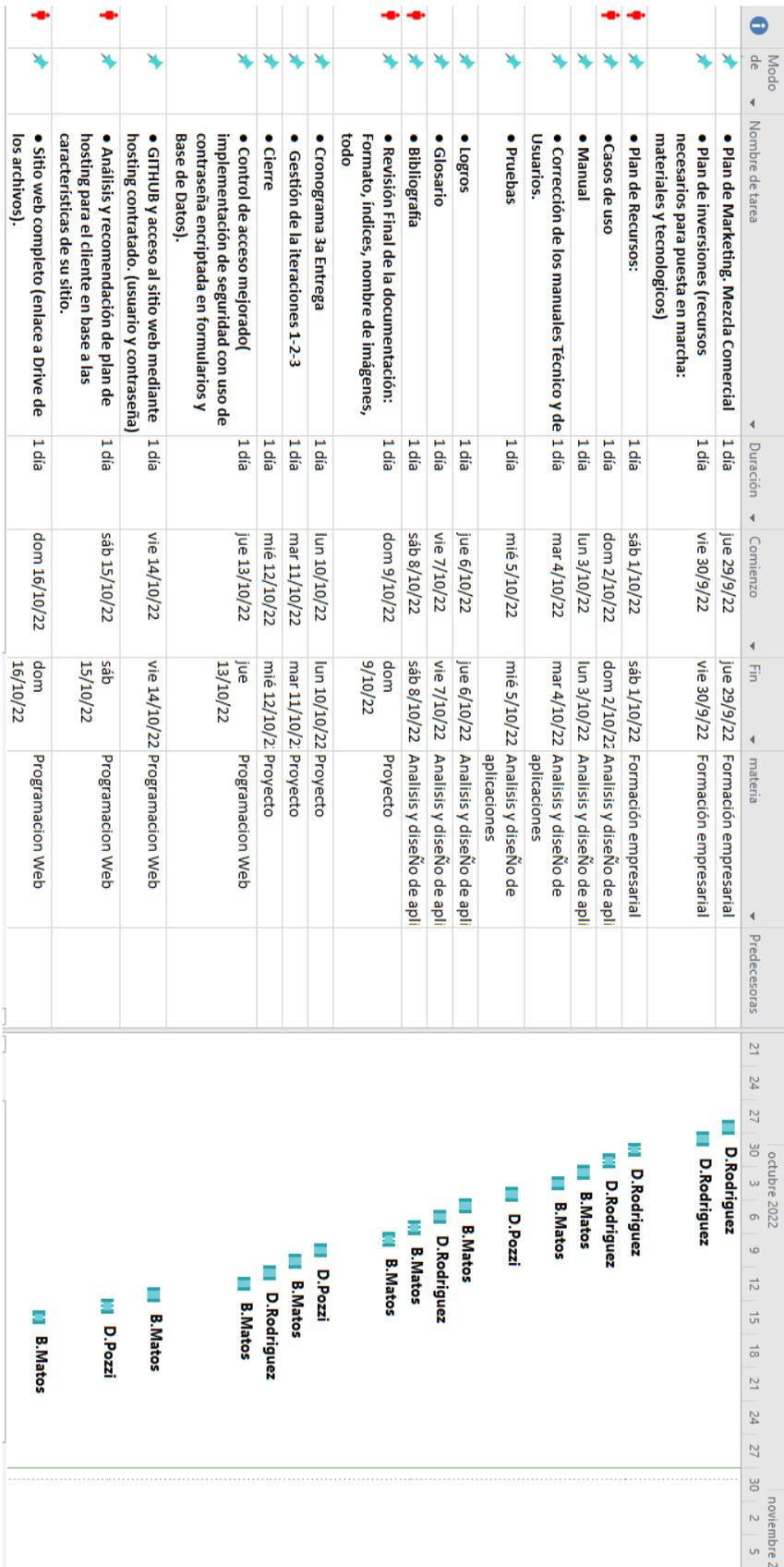


FIG51. diagrama de Gantt cronograma 3(parte 1)



FIG52. diagrama de Gantt cronograma 3(parte 2)

2.14 Informe sobre SGBD elegido

Sistemas de Gestión de Base de Datos. Es una aplicación que permite a los usuarios definir, crear y mantener la base de datos, y proporciona acceso controlado a la misma. El SGBD es la aplicación que interacciona con los usuarios de los programas de aplicación y la base de datos. Este proporciona los siguientes servicios:

- Permite la definición de la base de datos mediante el lenguaje de definición de datos. Permitiendo especificar la estructura y el tipo de los datos, así como las restricciones sobre los datos. Todo esto se almacenará en la base de datos.

- Permite la inserción, actualización, eliminación y consulta de datos mediante el lenguaje de manejo de datos. Al disponer de un lenguaje para realizar consultas reduce el problema de los sistemas de ficheros, en los que el usuario tiene que trabajar con un conjunto fijo de consultas, o bien, dispone de un gran número de programas de aplicación costosos de gestionar.

Hay dos tipos de lenguajes de manejo de datos: los procedurales y los no procedurales. Estos se diferencian por la forma en que acceden a los datos. Los lenguajes procedurales manipulan la base de datos registro a registro, mientras que los no procedurales operan sobre conjuntos de registros. En los procedurales se especifica qué operaciones se deben realizar para obtener los datos resultados, mientras que en los lenguajes no procedurales se especifica qué datos deben obtenerse sin decir cómo hacerlo. El lenguaje no procedural más utilizado es el SQL que, de hecho, es un estándar y es el lenguaje de los SGBD relacionales.

Diferentes Sistemas gestores de base de datos que existen:

- Relacional
- Jerárquica
- De red
- Orientada a objetos
- Orientada a documentos

Ventajas de los sistemas gestores de base de datos:

- Gestión fácil de grandes conjuntos de datos
- Acceso sencillo y eficaz a los datos almacenados
- Gran flexibilidad
- Integridad y consistencia de los datos
- Control de acceso del usuario (seguridad y protección de datos)
- Alta disponibilidad

Inconvenientes de los sistemas gestores de base de datos:

- Inversión inicial relativamente elevada (incluidos costes de hardware adicionales)
- Bastante menos eficaz para el software especial
- Se requieren empleados cualificados (administradores de bases de datos)
- Mayor vulnerabilidad por el hecho de centralizar los datos

Ejemplos de los SGBD más populares y utilizados en el mercado:

MySQL:

Es un sistema gestor de base de datos, y actualmente uno de los más usados y reconocidos del mercado. Especialmente en lo que se refiere a desarrollo web, está clasificada como la base de datos de código abierto más popular del mundo.

Características:

- Permite escoger múltiples motores de almacenamiento para cada tabla.
- Agrupación de transacciones, pudiendo reunir las de forma múltiple desde varias conexiones con el fin de incrementar el número de transacciones por segundo.
- Conectividad segura.
- Ejecución de transacciones y uso de claves foráneas.
- Presenta un amplio subconjunto del lenguaje SQL.
- Replicación
- Disponible en casi todas las plataformas o sistemas.
- Búsqueda e indexación de campos de texto.
- Utiliza varias herramientas para portabilidad.
- Tablas hash en memorias temporales
- Uso de tablas en disco b-tree para búsquedas rápidas con compresión de índice.
- Ofrece un sistema de contraseñas y privilegios seguros de verificación basada en el host y tráfico de contraseñas encriptado al conectarse a un servidor.
- Uso de multihilos mediante hilos de kernel.
- Soporta gran cantidad de datos, incluso con más de 50 millones de registros.
- En las últimas versiones, se permiten hasta 64 índices por tablas. Cada índice puede consistir desde 1 a 16 columnas o partes de columnas. El máximo ancho de límite es de 1000 bytes.

MariaDB:

MariaDB es un al igual que el anterior es u SGBD, se deriva de MySQL, una de las bases de datos más importantes que ha existido

en el mercado, utilizada para manejar grandes cantidades de información.

Características:

- **Mejoras en la velocidad**

MariaDB es muy rápida a la hora de realizar consultas complejas gracias al uso del motor aria, que utiliza el caché para almacenar las filas de datos, en lugar de escribir en disco. También se han eliminado conversiones innecesarias de juegos de caracteres que consiguió incrementar su velocidad entre el 1 y el 5 %. Con estas y otras características se puede decir que MariaDB es un sistema rápido.

- **Extensiones**

Se han añadido diversas extensiones a MariaDB que le proporcionan mejoras y nuevas funcionalidades, como la posibilidad de manejar hasta 32 segmentos clave por clave (duplicando la capacidad inicial), uso de columnas virtuales, posibilidad de incluir un sistema de autenticación, selección del motor de almacenamiento al crear una tabla, caché de claves segmentadas o incremento de la precisión en la lista de procesos (en microsegundos), entre otras mejoras.

- **Errores y alertas**

En cuanto a los errores, en MariaDB se han corregido la mayoría de posibles errores que se producen en la gestión de bases de datos, y de forma constante, se añaden mejoras al sistema en este sentido.

Las alertas del compilador se han reducido en su mayoría, dejando solo aquellas que realmente son necesarias para optimizar, mejorar o solventar una incidencia.

- **Documentación y ayuda**

El soporte disponible de MariaDB es muy extenso. Se puede acceder a toda la información que se necesita para utilizar este gestor de bases de datos desde su documentación oficial. La comunidad alrededor de este proyecto es muy amplia y activa, por lo que es fácil hallar soluciones a cualquier problema o incidencia, o encontrar cómo realizar procesos o configuraciones avanzadas.

- **Licencia GPL**

MariaDB se distribuye bajo la licencia GPL, por lo que se trata de un sistema de software libre que puede utilizarse de manera gratuita en cualquier proyecto.

CouchDB

CouchDB es una base de datos NoSQL de código abierto basada en estándares comunes para facilitar la accesibilidad y compatibilidad web con una diversidad de dispositivos.

Características:

- Replicación fácil entre servidores a través de instancias.
- Soporte para resolución de conflictos y configuración del máster.
- Actualizaciones de alimentación de datos por suscripción.
- Indexación y búsqueda y recuperación rápidas.
- Interfaz web RESTful.
- Los documentos son accesibles a través de navegadores y los índices pueden ser consultados a través de HTTP.
- Operaciones de indexación, combinación y transformación realizadas con JavaScript.
- Operaciones para crear, leer, actualizar, borrar documentos simples (CRUD).
- MapReduce avanzado.

IMS de IBM

IMS es un gestor de transacciones basado en mensajes y un gestor de base de datos jerárquica para los OS. Las aplicaciones externas pueden utilizar transacciones para interactuar con aplicaciones que se ejecutan dentro de IMS.

Características:

- Utilizan un lenguaje para la comunicación con las Bases de datos DL/1 (Data Language/Interface).
- DL/1 proporciona interfaces para la comunicación a su SGBD IMS desde diversos lenguajes (p.e. PL/1 de IBM y COBOL).
- IMS está diseñado para trabajar con bases de datos jerárquicas.
- Es un sistema robusto, rápido, eficiente, mantenido y soportado por una de las grandes del sector IBM.

SGBD	Suscripción anual
MySQL estándar	\$USD 2000
MariaDB estándar	\$USD 30
CouchDB	Gratis
IMS de IMB	Varía

2.15 Modelo Entidad Relación (1a Versión)

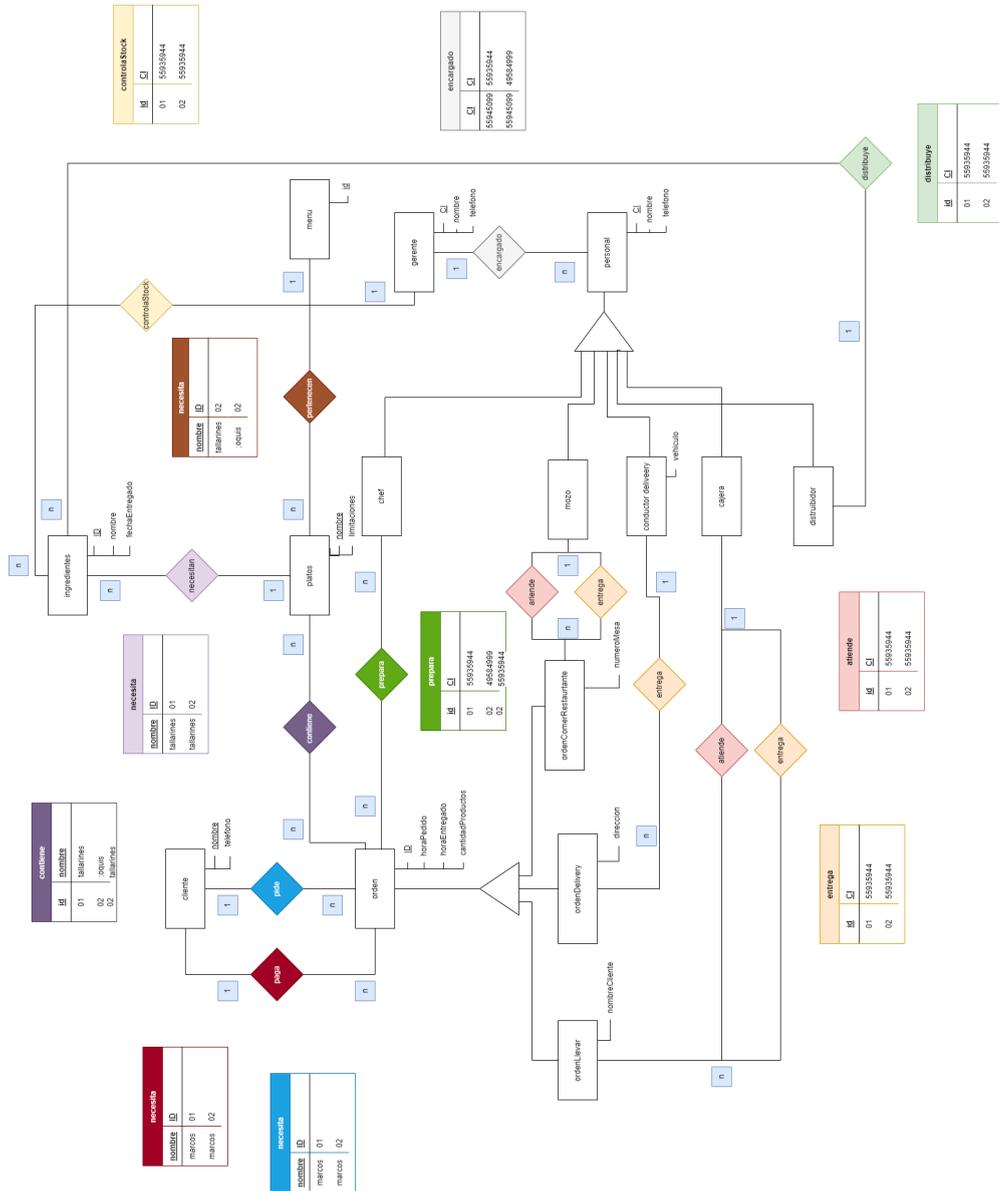


FIG53. MER

3 Proyecto

3.1 Análisis

Se va a implementar un software de comandas que va a integrarse por medio de una página web, para un restaurante.

3.2 Diseño

3.2.1 Diagrama de clases

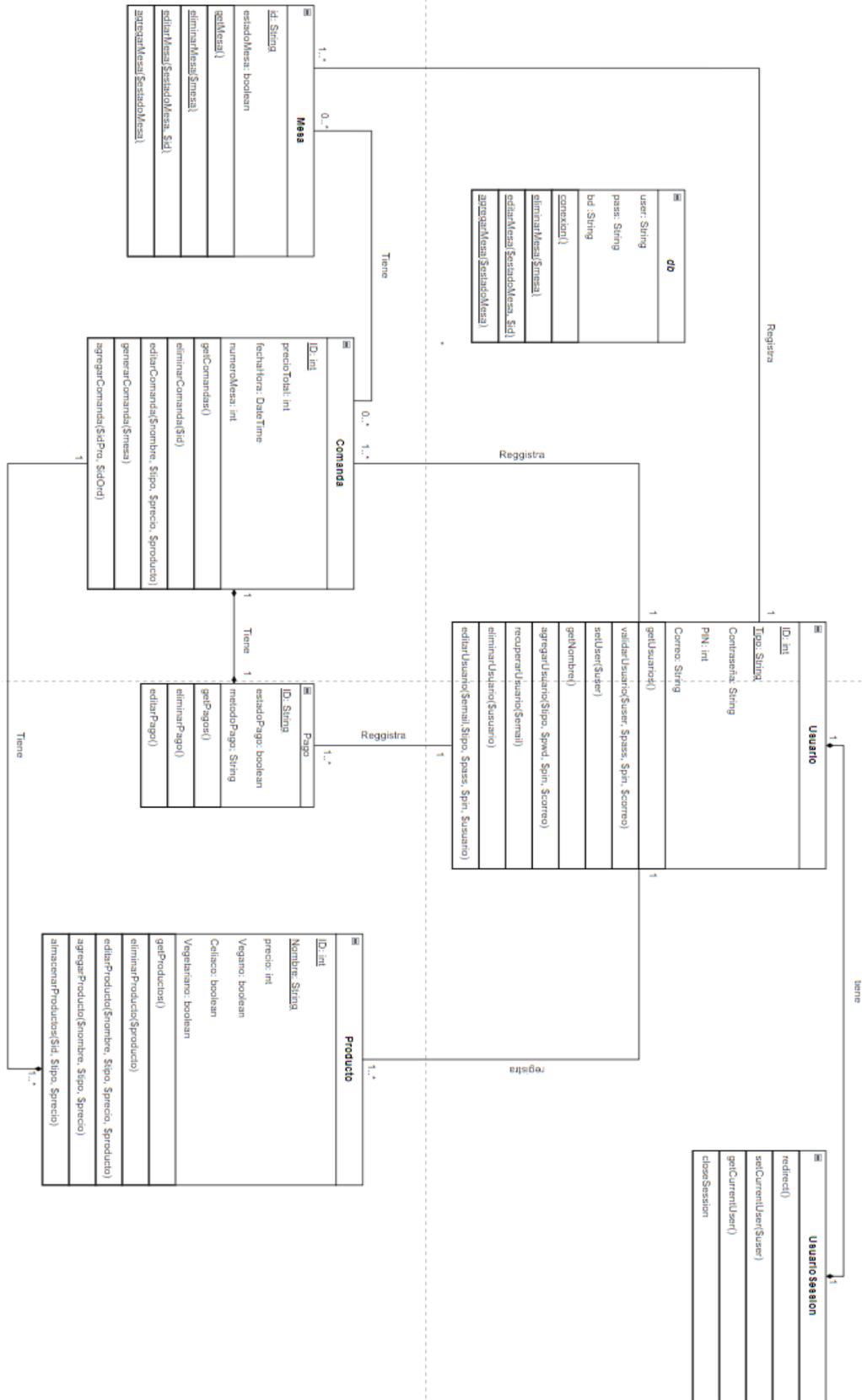


FIG54. Diagrama de clases

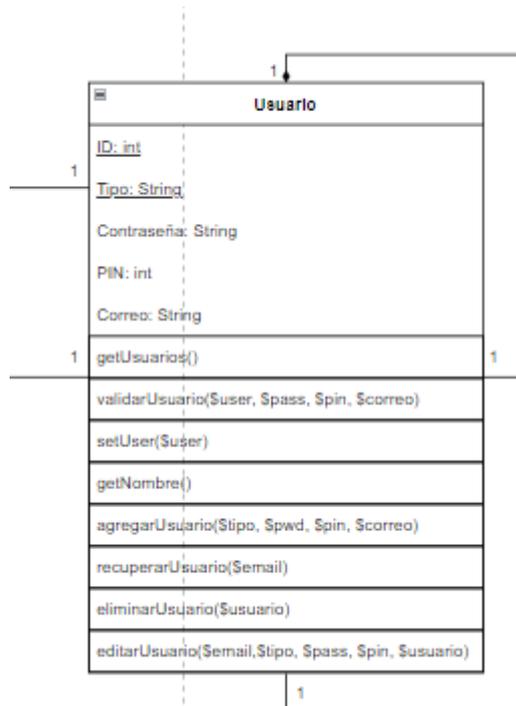


FIG55. Diagrama de clases 1

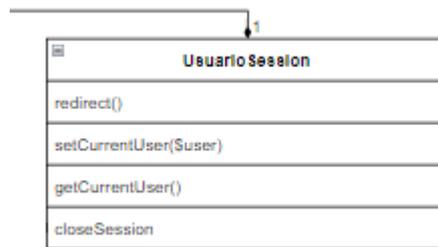


FIG56. Diagrama de clases 2

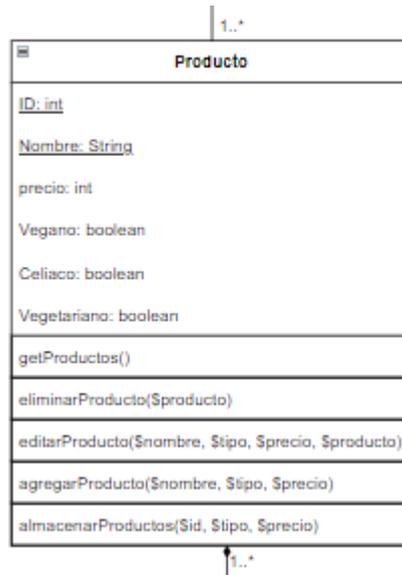


FIG57. Diagrama de clases 3

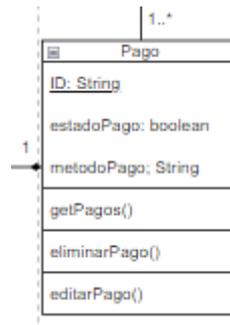


FIG58. Diagrama de clases 4

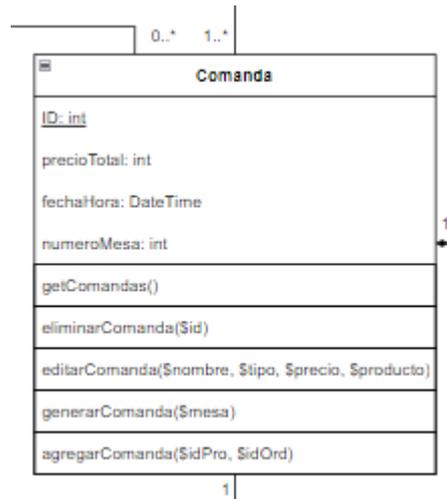


FIG59. Diagrama de clases 5



FIG60. Diagrama de clases 6

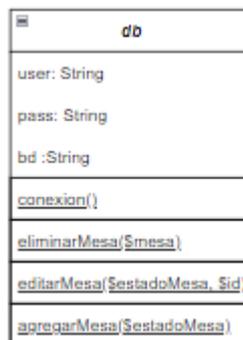


FIG61. Diagrama de clases 7

3.2.2 Casos de Uso

3.2.2.1 CU-01 Inicio de sesión

3.2.2.1.1 Diagrama de caso de uso

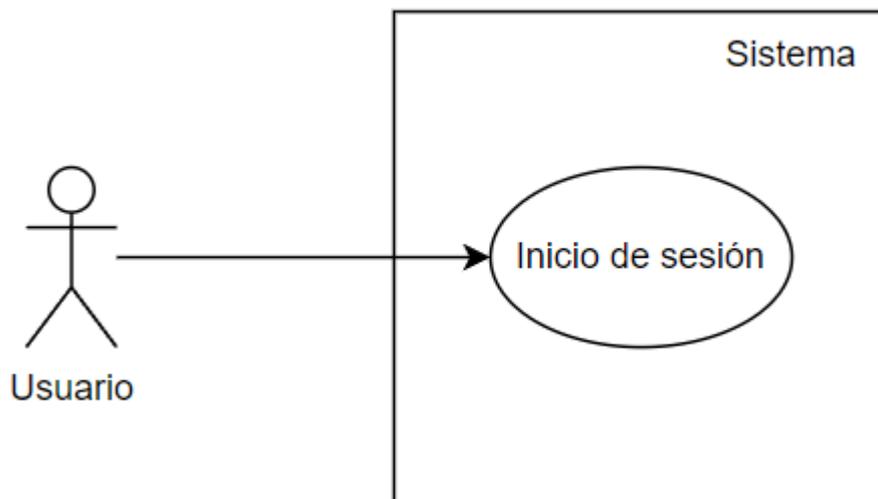


FIG62. Diagrama CU-1 Inicio de sesión

3.2.2.2 Formulario

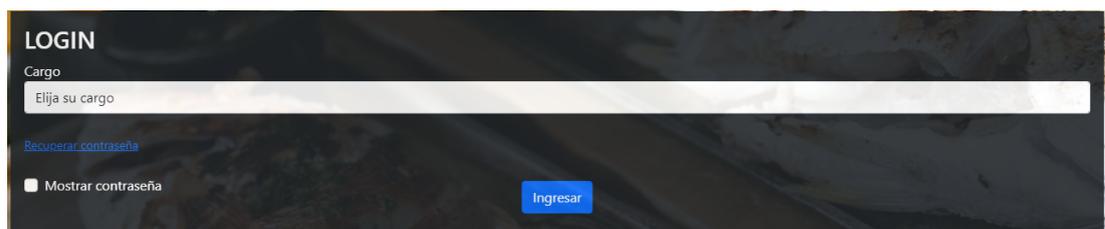


FIG63. Formulario inicio de sesión

3.2.2.2.1 Descripción del caso de uso

Descripción	
Nº	CU-01
Nombre	Inicio de sesión
Descripción	Permite a los usuarios ingresar al sistema
Tipo	Alto Nivel - Curso Normal y Alternativo

Prioridad	Alta
Actores participantes	Cualquier usuario
Pre-Condiciones (requerimientos)	El usuario a loguearse debe estar previamente ingresado en el sistema.
Curso normal	
Usuario	Sistema
	1. Despliega el formulario Login
2. Selecciona su tipo de usuario y rellena la contraseña o el pin dependiendo del usuario	
3. Hace click en el botón " Ingresar "	
	4. Verifica que no existan campos vacíos
	5. Verifica que la combinación de tipo de usuario y y Clave o PiN sean de un usuario registrado en la base de datos.
	6. Despliega la interfaz acorde al tipo de usuario
Curso alternativo	
	4.1 Existen campos vacíos
	4.2 Muestra un mensaje, "Por favor complete los campos"
	4.3 Vuelve al paso 3
	5.1 El usuario y contraseña o PIN no coinciden
	5.2 Muestra un mensaje "Usuario o contraseña incorrectas"
	5.3 Vuelve al paso 3
Referencias	
R. Funcional implementado	RF69- Inicio de sesión
Requerimiento no funcional	
Testing	
Pos-condición	El usuario luego de loguearse, accede al menú principal.

3.2.2.3 CU-02 Alta usuarios

3.2.2.3.1 Formulario

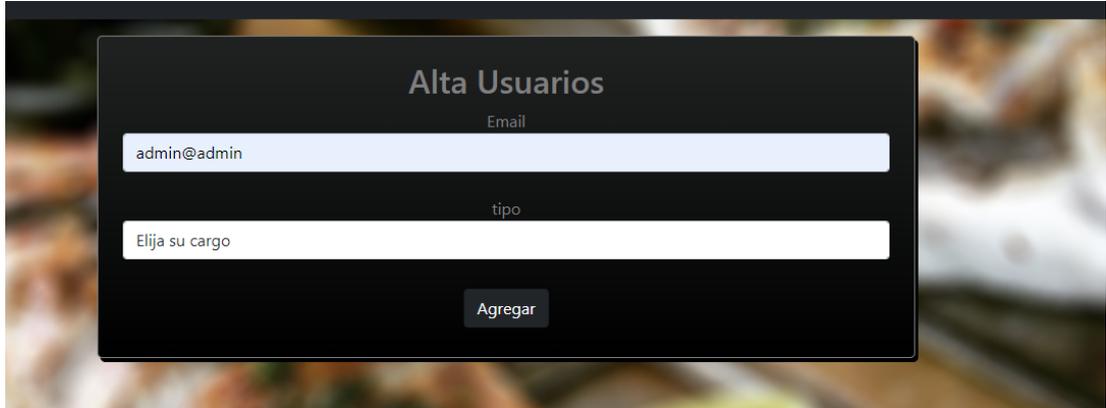


FIG64. Formulario Alta Usuarios

3.2.2.3.2 Descripción del caso de uso

Descripción	
Nº	CU-02
Nombre	Alta usuarios
Descripción	Permite al administrador dar de alta usuarios
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador.
Curso normal	
Usuario	Sistema
1.Presiona "Agregar" en el formulario de usuarios	
	2. Despliega formulario de registro
3. Rellena los campos	
4. Presiona el botón de "Registrar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Ingresa el usuario al sistema.
Curso alternativo	

2.1 Se presiona "Registrar" con algún campo vacío	
2.1 Se presiona "Registrar" con algún campo incorrecto	
5.1 Usuario presiona "Registrar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen "
7.1 Usuario vuelve a ingresar los datos	
	7.2 Sistema muestra un mensaje "Los datos ingresados ya existen"
Referencias	
R. Funcional implementado	RF63- Alta usuarios del sistema
Requerimiento no funcional	
Testing	
Pos-condición	Se crea un nuevo usuario

3.2.2.4 CU-03 Baja usuarios

3.2.2.4.1 Formulario

correo	Tipo	Contraseña	Pin	estadoRecuperar	
admin@admin	Administrador	1234		usuario OK	Editar Eliminar
mozo@mozo	Mozo		1234	usuario OK	Editar Eliminar
gerente@gerente	Gerente	1234		usuario OK	Editar Eliminar
cocina@cocina	Cocina		1234	usuario OK	Editar Eliminar
admin@admin	Administrador	332323		usuario OK	Editar Eliminar

FIG65. Formulario Baja Usuarios

3.2.2.4.2 Descripción del caso de uso

Descripción	
Nº	CU-03
Nombre	Baja usuarios
Descripción	Permite al administrador dar de baja usuarios
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador y el usuario a dar de baja debe estar registrado.
Curso normal	
Usuario	Sistema
1.Presiona "Eliminar" en el usuario a eliminar, dentro del listado de usuarios	
	2. Elimina el usuario
	3. Refresca la página
Curso alternativo	
Referencias	
R. Funcional implementado	RF64- Baja usuarios del sistema

Requerimiento no funcional	
Testing	
Pos-condición	El usuario seleccionado ya no existe más

3.2.2.5 CU-04 Modificar usuarios

3.2.2.5.1 Formulario

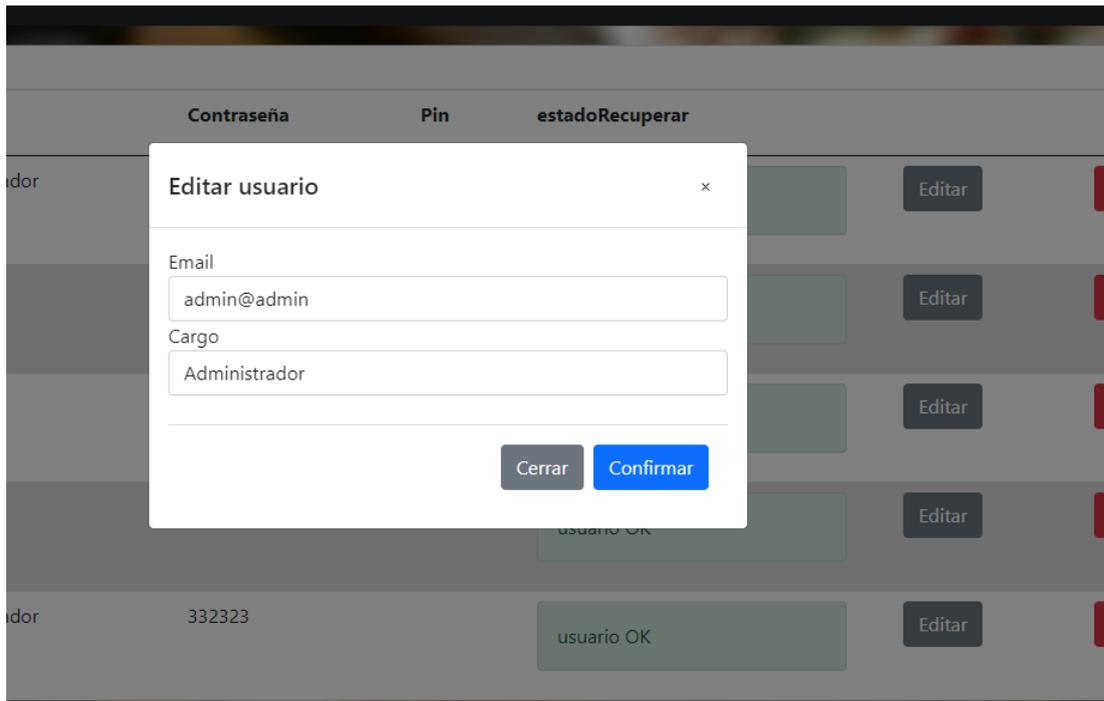


FIG66. Formulario Modificar Usuarios

3.2.2.5.2 Descripción del caso de uso

Descripción	
Nº	CU-04
Nombre	Modificar usuarios
Descripción	Permite al administrador modificar los usuarios registrados
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser administrador.
Curso normal	
Usuario	Sistema

1. Presiona "Modificar" en el listado de usuarios	
	2. Despliega formulario de modificación
3. Rellena los campos	
4. Presiona "Modificar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Modifica el usuario del sistema.
Curso alternativo	
2.1 Se presiona "Modificar" con algún campo vacío	
2.1 Se presiona "Modificar" con algún campo incorrecto	
5.1 Usuario presiona "Modificar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen "
Referencias	
R. Funcional implementado	RF65- Modificar usuarios del sistema
Requerimiento no funcional	
Testing	
Pos-condición	El usuario seleccionado fue modificado

3.2.2.6 CU-05 Listado usuarios

3.2.2.6.1 Formulario

correo	Tipo	Contraseña	Pin	estadoRecuperar	
admin@admin	Administrador	1234		usuario OK	Editar Eliminar
mozo@mozo	Mozo		1234	usuario OK	Editar Eliminar
gerente@gerente	Gerente	1234		usuario OK	Editar Eliminar
cocina@cocina	Cocina		1234	usuario OK	Editar Eliminar
admin@admin	Administrador	332323		usuario OK	Editar Eliminar

FIG67. Formulario Listar Usuarios

3.2.2.6.2 Descripción del caso de uso

Descripción	
Nº	CU-05
Nombre	Listado usuarios
Descripción	Permite al administrador visualizar la lista de usuarios registrados y sus datos
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario a loguearse debe estar previamente ingresado en el sistema y debe ser Administrador.
Curso normal	
Usuario	Sistema
1.Presiona el botón “Usuarios” en el formulario principal	
	2. Despliega el listado de usuarios
Curso alternativo	

Referencias	
R. Funcional implementado	RF66- Listar usuarios del sistema
Requerimiento no funcional	
Testing	
Pos-condición	El usuario puede realizar cualquier acción dentro del listado

3.2.2.7 CU-06 Alta producto

3.2.2.7.1 Diagrama de caso de uso

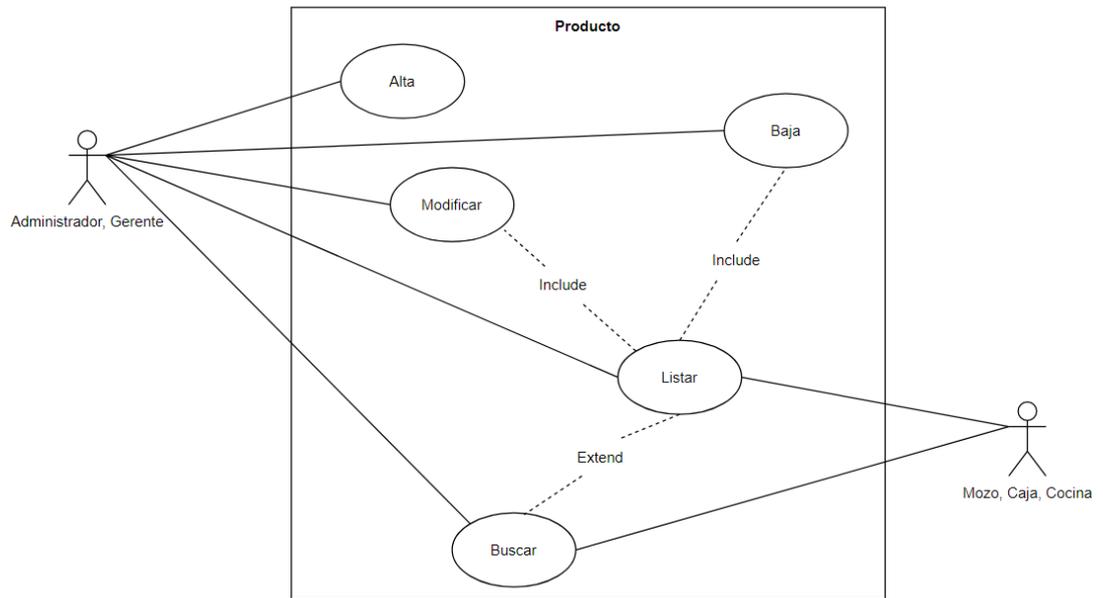


FIG68. Diagrama CU-02 Alta producto

3.2.2.7.2 Formulario

FIG69. Formulario Alta Producto

3.2.2.7.3 Descripción del caso de uso

Descripción

N°	CU-06
Nombre	Alta Producto
Descripción	Permite al administrador dar de alta productos
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador.
Curso normal	
Usuario	Sistema
1. Presiona "Agregar" en el listado de productos	
	2. Despliega formulario de registro
3. Rellena los campos	
4. Presiona "Registrar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Ingresa el producto al sistema.
Curso alternativo	
2.1 Se presiona "Registrar" con algún campo vacío	
2.1 Se presiona "Registrar" con algún campo incorrecto	
5.1 Usuario presiona "Registrar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen"
Referencias	
R. Funcional implementado	RF32- Alta productos
Requerimiento no funcional	
Testing	
Pos-condición	Se crea un nuevo producto

3.2.2.8 CU-07 Baja producto

3.2.2.8.1 Formulario

Imagen	ID	Nombre	Tipo	Precio		
	59	pizza	Plato	123	Editar	Eliminar
	60	flan	Postre	123	Editar	Eliminar
	74	coca-col	Bebida	2135	Editar	Eliminar
	75	helado	Postre	123	Editar	Eliminar

FIG70. Formulario Baja Producto

3.2.2.8.2 Descripción del caso de uso

Descripción	
Nº	CU-07
Nombre	Baja Producto
Descripción	Permite al administrador eliminar productos
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador y el producto a dar de baja debe estar registrado.
Curso normal	
Usuario	Sistema
1.Presiona "Eliminar" en el producto a eliminar, dentro del listado de productos	
	2. Elimina el producto
	3. Refresca la página 2 veces
Curso alternativo	
Referencias	
R. Funcional implementado	RF33- Baja producto
Requerimiento no funcional	
Testing	
Pos-condición	El producto es eliminado del sistema

3.2.2.9 CU-08 Modificar producto

3.2.2.9.1 Formulario

The image shows a modal window titled "Editar producto" with a close button (X) in the top right corner. The form contains three input fields: "Nombre:" with the value "pizza", "Tipo:" with the value "Bebida", and "Precio:" with the value "123". At the bottom right of the form are two buttons: "Cerrar" (grey) and "Confirmar" (blue).

FIG71. Formulario Modificar Producto

3.2.2.9.2 Descripción del caso de uso

Descripción	
Nº	CU-08
Nombre	Modificar Producto
Descripción	Permite al administrador modificar los productos registrados
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser administrador.
Curso normal	
Usuario	Sistema
1.Presiona "Modificar" en el listado de productos	
	2. Despliega formulario de modificación
3. Rellena los campos	
4. Presiona "Modificar"	

	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Modifica el producto del sistema.
Curso alternativo	
2.1 Se presiona "Modificar" con algún campo vacío	
2.1 Se presiona "Modificar" con algún campo incorrecto	
5.1 Usuario presiona "Modificar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen "
Referencias	
R. Funcional implementado	RF34- Modificar productos
Requerimiento no funcional	
Testing	
Pos-condición	El producto seleccionado fue modificado

3.2.2.10 CU-09 Listado producto

3.2.2.10.1 Formulario

Imagen	ID	Nombre	Tipo	Precio		
	59	pizza	Plato	123	Editar	Eliminar
	60	flan	Postre	123	Editar	Eliminar
	74	coca-col	Bebida	2135	Editar	Eliminar
	75	helado	Postre	123	Editar	Eliminar

FIG72. Formulario Listado Producto

3.2.2.10.2 Descripción del caso de uso

Descripción	
Nº	CU-09
Nombre	Listado Producto
Descripción	Permite al administrador visualizar la lista de productos registrados y sus datos
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe tener permiso para ver la lista
Curso normal	
Usuario	
1.Presiona "Producto" en el nav	
	2. Despliega el listado de productos
Curso alternativo	
1. Usuario no autorizado ingresa el link	
	1.1 No se le permite ingresar
	1.2 Se le redirecciona al login en caso de no estar logueado, a la página principal en caso de estar logueado en una cuenta sin permiso
Referencias	
R. Funcional implementado	RF35- Listar productos

Requerimiento no funcional	
Testing	
Pos-condición	El usuario puede realizar cualquier acción dentro del listado

3.2.2.11 CU-10 Alta Mesa

3.2.2.11.1 Formulario

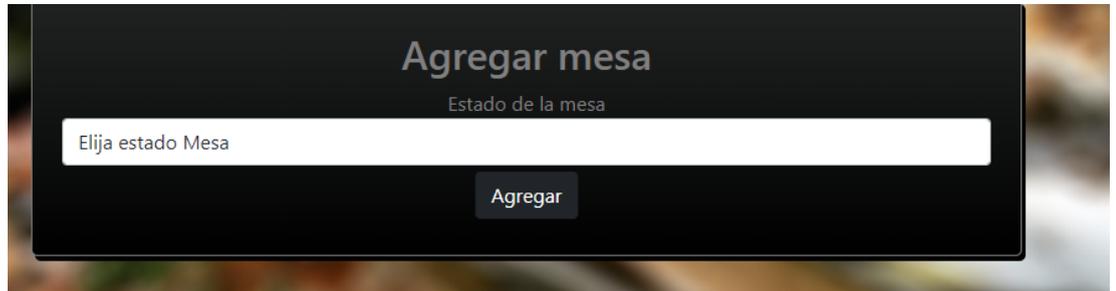


FIG73. Formulario Alta Mesa

3.2.2.11.2 Descripción del caso de uso

Descripción	
Nº	CU-10
Nombre	Alta Mesa
Descripción	Permite al administrador dar de alta mesas
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador.
Curso normal	
Usuario	Sistema
1. Presiona "Agregar" en el listado de mesa	
	2. Despliega formulario de registro
3. Rellena los campos	
4. Presiona "Registrar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Ingresar la mesa al sistema.
Curso alternativo	
2.1 Se presiona "Registrar" con algún campo vacío	
2.1 Se presiona "Registrar" con algún campo incorrecto	
5.1 Usuario presiona "Registrar" sin completar los campos	

	5.2 Sistema muestra un mensaje “Campos incompletos”
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje “Los datos ingresados no existen”
Referencias	
R. Funcional implementado	RF8- Alta mesa
Requerimiento no funcional	
Testing	
Pos-condición	Se crea una nueva mesa

3.2.2.12 CU-11 Baja Mesa

3.2.2.12.1 Formulario

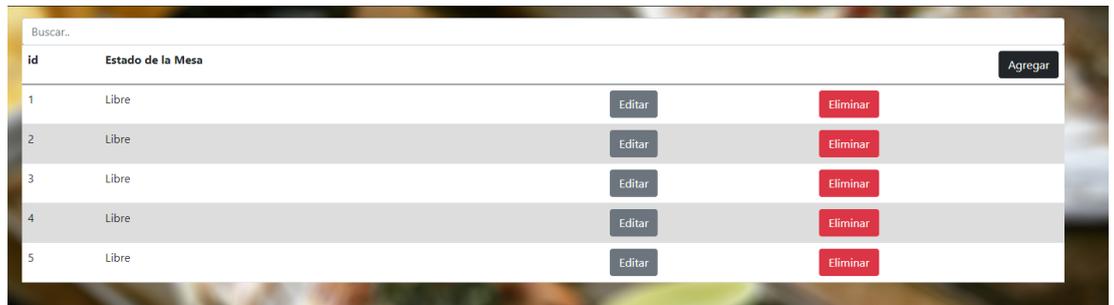


FIG74. Formulario Baja Mesa

3.2.2.12.2 Descripción del caso de uso

Descripción	
Nº	CU-11
Nombre	Baja Mesa
Descripción	Permite al administrador eliminar mesas
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador y la mesa a dar de baja debe estar registrada.
Curso normal	
Usuario	Sistema
1.Presiona “Eliminar” en la mesa a eliminar, dentro del listado de mesas	
	2. Elimina la mesa

	3. Refresca la página 2 veces
Curso alternativo	
Referencias	
R. Funcional implementado	RF9- Baja mesa
Requerimiento no funcional	
Testing	
Pos-condición	La mesa seleccionada ya no existe más

3.2.2.13 CU-12 Modificar Mesa

3.2.2.13.1 Formulario

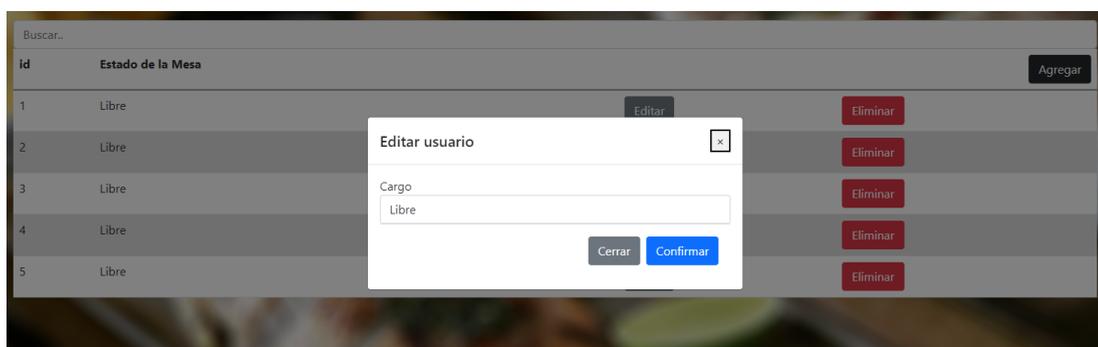


FIG75. Formulario Modificar Mesa

3.2.2.13.2 Descripción del caso de uso

Descripción	
Nº	CU-12
Nombre	Modificar Mesa
Descripción	Permite al administrador modificar las mesas registradas
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser administrador.
Curso normal	
Usuario	Sistema
1. Presiona "Modificar" en el listado de mesa	
	2. Despliega formulario de modificación
3. Rellena los campos	
4. Presiona "Modificar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Modifica la mesa del sistema.
Curso alternativo	
2.1 Se presiona "Modificar" con algún campo vacío	
2.1 Se presiona "Modificar" con algún campo incorrecto	

5.1 Usuario presiona "Modifica"r sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen "
Referencias	
R. Funcional implementado	RF10- Modificar mesa
Requerimiento no funcional	
Testing	
Pos-condición	La mesa seleccionada fue modificada

3.2.2.14 CU-13 Listado mesas

3.2.2.14.1 Formulario

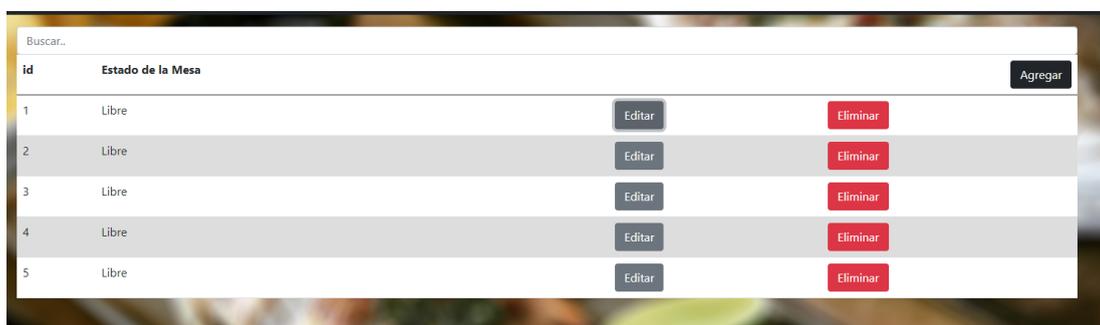


FIG76. Formulario Listado Mesa

3.2.2.14.2 Descripción del caso de uso

Descripción	
Nº	CU-13
Nombre	Listado Mesa
Descripción	Permite al administrador visualizar la lista de mesas registradas y sus datos
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe tener permiso para ver la lista
Curso normal	
Usuario	
1.Presiona "Mesa" en el nav	
	2. Despliega el listado de mesa
Curso alternativo	
2. Usuario no autorizado ingresa el link	
	1.1 No se le permite ingresar
	1.2 Se le redirecciona al login en caso de no estar logueado, a la página principal en caso de estar logueado en una cuenta sin permiso
Referencias	

R. Funcional implementado	RF11- Listar mesa
Requerimiento no funcional	
Testing	
Pos-condición	El usuario puede realizar cualquier acción dentro del listado

3.2.2.15 CU-14 Alta orden

3.2.2.15.1 Formulario

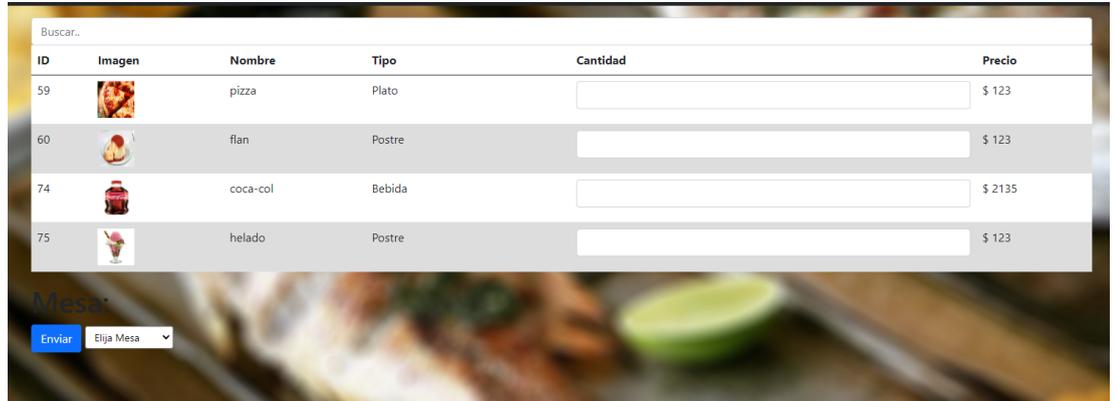


FIG77. Formulario Alta Orden

3.2.2.15.2 Descripción del caso de uso

Descripción	
Nº	CU-14
Nombre	Alta Orden
Descripción	Permite al administrador dar de alta ordenes
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador.
Curso normal	
Usuario	Sistema
1. Selecciona los productos de la orden indicando que cantidad requiere la comanda	
2. Selecciona la mesa a la que va la comanda	
3. Presiona enviar	
	2, Genera la orden
	3. Asigna los productos a la orden
	4. Envía la orden
Curso alternativo	

1.1 No selecciona ningún producto	
	1.2 Da aviso de que ningún producto fue seleccionado
2.1 No selecciona ninguna mesa	
	2.2 Da aviso de que no se seleccionó ninguna mesa
Referencias	
R. Funcional implementado	RF2- Alta orden
Requerimiento no funcional	
Testing	
Pos-condición	Se crea una nueva orden

3.2.2.16 CU-15 Baja orden

3.2.2.16.1 Formulario

Buscar...							
ID	Precio	Fecha-Hora	Mesa	Estado Orden		Filtrar	
243	0	2022-10-29	1	Entregado	Detalles	Eliminar	
244	0	2022-10-29	1	Entregado	Detalles	Eliminar	
245	0	2022-10-29	1	Entregado	Detalles	Eliminar	

FIG78. Formulario Baja Orden

3.2.2.16.2 Descripción del caso de uso

Descripción	
Nº	CU-15
Nombre	Baja Orden
Descripción	Permite al administrador eliminar órdenes
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador y la orden a dar de baja debe estar registrada.
Curso normal	
Usuario	Sistema
1. Presiona "Eliminar" en la orden a eliminar, dentro del listado de ordenes	
	2. Elimina la mesa
	3. Refresca la página 2 veces
Curso alternativo	
Referencias	
R. Funcional implementado	RF3- Baja orden
Requerimiento no funcional	
Testing	
Pos-condición	La orden seleccionada ya no existe más

3.2.2.17 CU-16 Modificar orden

3.2.2.17.1 Formulario

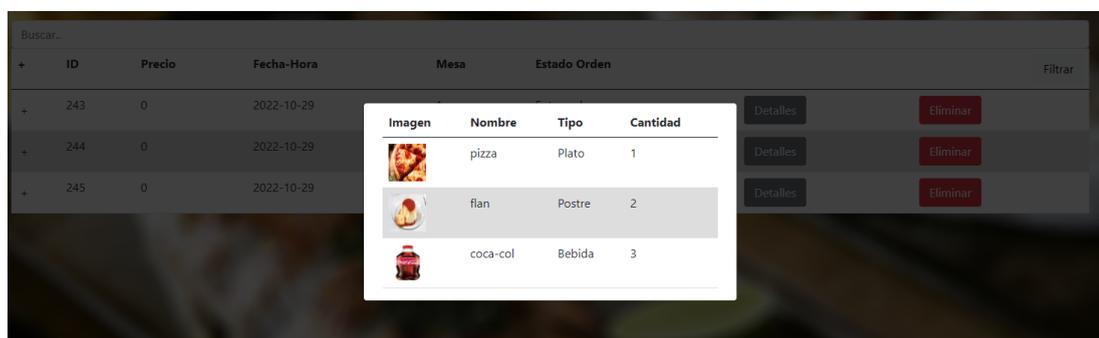


FIG79. Formulario Modificar Orden

3.2.2.17.2 Descripción del caso de uso

Descripción	
Nº	CU-16
Nombre	Modificar Orden
Descripción	Permite al administrador modificar las órdenes registradas
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser administrador.
Curso normal	
Usuario	Sistema
1. Presiona "Modificar" en el listado de orden	
	2. Despliega formulario de modificación
3. Rellena los campos	
4. Presiona "Modificar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Modifica la orden del sistema.
Curso alternativo	
2.1 Se presiona "Modificar" con algún campo vacío	
2.1 Se presiona "Modificar" con algún campo incorrecto	

5.1 Usuario presiona "Modificar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen"
Referencias	
R. Funcional implementado	RF4- Modificar orden
Requerimiento no funcional	
Testing	
Pos-condición	La orden seleccionada fue modificada

3.2.2.18 CU-17 Listado orden

3.2.2.18.1 Formulario

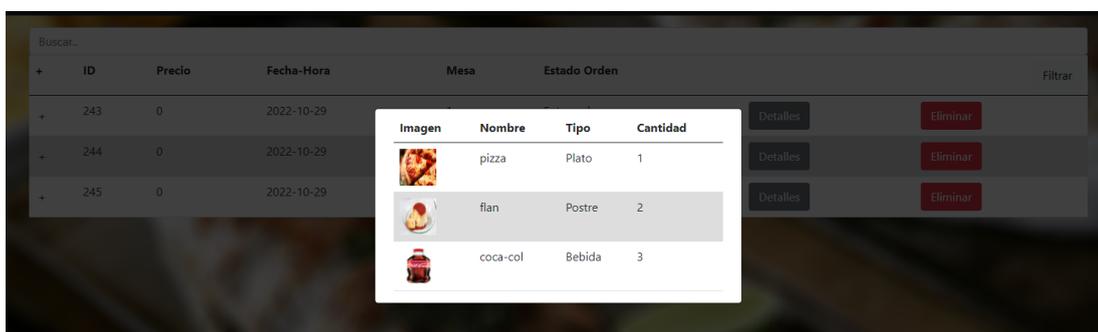


FIG80. Formulario Listado Orden

3.2.2.18.2 Descripción del caso de uso

Descripción	
Nº	CU-17
Nombre	Listado orden
Descripción	Permite al administrador visualizar la lista de órdenes registradas y sus datos
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe tener permiso para ver la lista
Curso normal	
Usuario	
1. Presiona "Orden" en el nav	
	2. Despliega el listado de orden
Curso alternativo	
3. Usuario no autorizado ingresa el link	
	1.1 No se le permite ingresar
	1.2 Se le redirecciona al login en caso de no estar logueado, a la página principal en caso de estar logueado en una cuenta sin permiso
Referencias	

R. Funcional implementado	RF5- Listar orden
Requerimiento no funcional	
Testing	
Pos-condición	El usuario puede realizar cualquier acción dentro del listado

3.2.2.19 CU-18 Alta cliente

3.2.2.19.1 Formulario

3.2.2.19.2 Descripción del caso de uso

Descripción	
Nº	CU-18
Nombre	Alta Cliente
Descripción	Permite al administrador dar de alta clientes
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador.
Curso normal	
Usuario	Sistema
1. Presiona "Agregar" en el listado de cliente	
	2. Despliega formulario de registro
3. Rellena los campos	
4. Presiona "Registrar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Ingresa al cliente al sistema.
Curso alternativo	
2.1 Se presiona "Registrar" con algún campo vacío	
2.1 Se presiona "Registrar" con algún campo incorrecto	
5.1 Usuario presiona "Registrar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen "
Referencias	
R. Funcional implementado	RF13- Alta cliente
Requerimiento no funcional	

Testing	
Pos-condición	Se crea un nuevo cliente

3.2.2.20 CU-19 Baja cliente

3.2.2.20.1 Formulario

3.2.2.20.2 Descripción del caso de uso

Descripción	
Nº	CU-19
Nombre	Baja Cliente
Descripción	Permite al administrador eliminar clientes
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador y el cliente a dar de baja debe estar registrado.
Curso normal	
Usuario	Sistema
1.Presiona "Eliminar" en el cliente a eliminar, dentro del listado de clientes	
	2. Elimina al cliente
	3. Refresca la página 2 veces
Curso alternativo	
Referencias	
R. Funcional implementado	RF14- Baja cliente
Requerimiento no funcional	
Testing	
Pos-condición	El cliente seleccionado ya no existe más

3.2.2.21 CU-20 Modificar cliente

3.2.2.21.1 Formulario

3.2.2.21.2 Descripción del caso de uso

Descripción	
Nº	CU-20
Nombre	Modificar Cliente
Descripción	Permite al administrador modificar los clientes registrados
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser administrador.
Curso normal	
Usuario	Sistema
1. Presiona "Modificar" en el listado de cliente	
	2. Despliega formulario de modificación
3. Rellena los campos	
4. Presiona "Modificar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Modifica el cliente del sistema.
Curso alternativo	
2.1 Se presiona "Modificar" con algún campo vacío	
2.1 Se presiona "Modificar" con algún campo incorrecto	
5.1 Usuario presiona "Modificar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen "
Referencias	
R. Funcional implementado	RF15- Modificar cliente
Requerimiento no funcional	

Testing	
Pos-condición	El cliente seleccionado fue modificado

3.2.2.22 CU-21 Listado cliente

3.2.2.22.1 Formulario

3.2.2.22.2 Descripción del caso de uso

Descripción	
Nº	CU-21
Nombre	Listado Cliente
Descripción	Permite al administrador visualizar la lista de cliente registradas y sus datos
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe tener permiso para ver la lista
Curso normal	
Usuario	Sistema
1.Presiona "Cliente" en el nav	
	2. Despliega el listado de cliente
Curso alternativo	
4. Usuario no autorizado ingresa el link	
	1.1 No se le permite ingresar
	1.2 Se le redirecciona al login en caso de no estar logueado, a la página principal en caso de estar logueado en una cuenta sin permiso
Referencias	
R. Funcional implementado	RF16- Listar cliente
Requerimiento no funcional	
Testing	
Pos-condición	El usuario puede realizar cualquier acción dentro del listado

3.2.2.23 CU-22 Alta Personal

3.2.2.23.1 Formulario

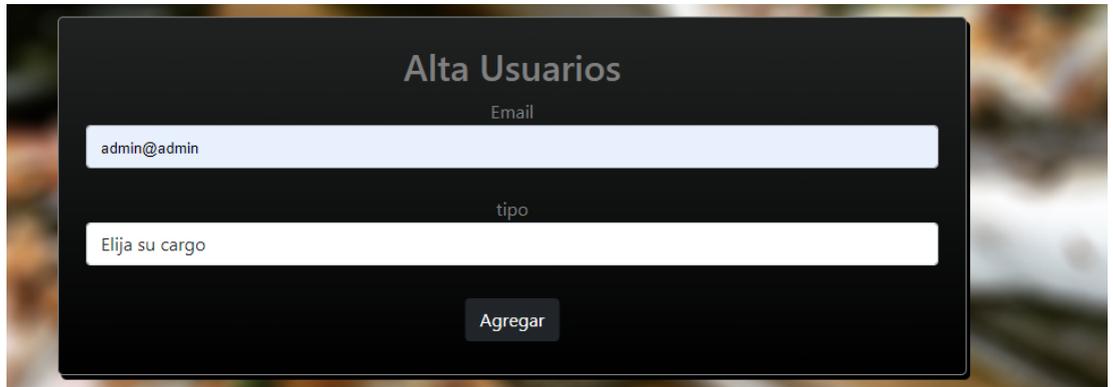


FIG81. Formulario Alta Personal

3.2.2.23.2 Descripción del caso de uso

Descripción	
Nº	CU-22
Nombre	Alta Personal
Descripción	Permite al administrador dar de alta personal
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador.
Curso normal	
Usuario	Sistema
1. Presiona "Agregar" en el listado de personal	
	2. Despliega formulario de registro
3. Rellena los campos	
4. Presiona "Registrar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Ingresa al personal al sistema.
Curso alternativo	
2.1 Se presiona "Registrar" con algún campo vacío	

2.1 Se presiona "Registrar" con algún campo incorrecto	
5.1 Usuario presiona "Registrar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen "

Referencias

R. Funcional implementado	RF19- Alta personal
Requerimiento no funcional	
Testing	
Pos-condición	Se crea un nuevo personal

3.2.2.24 CU-23 Baja Personal

3.2.2.24.1 Formulario

correo	Tipo	Contraseña	Pin	estadoRecuperar	
admin@admin	Administrador	1234		usuario OK	Editar Eliminar
mozo@mozo	Mozo		1234	usuario OK	Editar Eliminar
gerente@gerente	Gerente	1234		usuario OK	Editar Eliminar
cocina@cocina	Cocina		1234	usuario OK	Editar Eliminar
admin@admin	Administrador	332323		usuario OK	Editar Eliminar

Formulario Baja Personal

3.2.2.24.2 Descripción del caso de uso

Descripción	
Nº	CU-23
Nombre	Baja Personal
Descripción	Permite al administrador eliminar personal
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador y el personal a dar de baja debe estar registrado.
Curso normal	
Usuario	Sistema
1.Presiona "Eliminar" en el personal a eliminar, dentro del listado de personal	
	2. Elimina al personal
	3. Refresca la página 2 veces
Curso alternativo	
Referencias	
R. Funcional implementado	RF20- Baja personal
Requerimiento no funcional	
Testing	
Pos-condición	El personal seleccionado ya no existe más

3.2.2.25 CU-24 Modificar Personal

3.2.2.25.1 Formulario

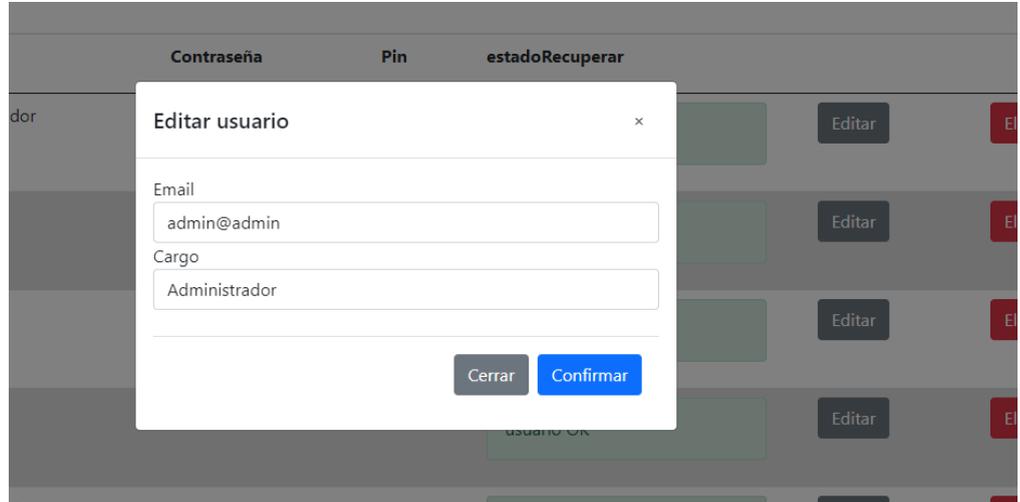


FIG82. Formulario Modificar Personal

3.2.2.25.2 Descripción del caso de uso

Descripción	
Nº	CU-24
Nombre	Modificar Personal
Descripción	Permite al administrador modificar el personal registrado
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser administrador.
Curso normal	
Usuario	Sistema
1. Presiona "Modificar" en el listado de personal	
	2. Despliega formulario de modificación
3. Rellena los campos	
4. Presiona "Modificar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados

	8. Modifica el personal del sistema.
Curso alternativo	
2.1 Se presiona "Modificar" con algún campo vacío	
2.1 Se presiona "Modificar" con algún campo incorrecto	
5.1 Usuario presiona "Modificar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen "
Referencias	
R. Funcional implementado	RF21- Modificar personal
Requerimiento no funcional	
Testing	
Pos-condición	El personal seleccionado fue modificado

3.2.2.26 CU-25 Listado Personal

3.2.2.26.1 Formulario

correo	Tipo	Contraseña	Pin	estado	Recuperar
admin@admin	Administrador	1234		usuario OK	Editar Eliminar
mozo@mozo	Mozo		1234	usuario OK	Editar Eliminar
gerente@gerente	Gerente	1234		usuario OK	Editar Eliminar
cocina@cocina	Cocina		1234	usuario OK	Editar Eliminar
admin@admin	Administrador	332323		usuario OK	Editar Eliminar

FIG83. Formulario Listado Personal

3.2.2.26.2 Descripción del caso de uso

Descripción	
Nº	CU-25
Nombre	Listado Personal
Descripción	Permite al administrador visualizar la lista del personal registrado y sus datos
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe tener permiso para ver la lista
Curso normal	
Usuario	Sistema
1. Presiona "Personal" en el nav	
	2. Despliega el listado de personal
Curso alternativo	
5. Usuario no autorizado ingresa el link	
	1.1 No se le permite ingresar

	1.2 Se le redirecciona al login en caso de no estar logueado, a la página principal en caso de estar logueado en una cuenta sin permiso
Referencias	
R. Funcional implementado	RF22- Listar personal
Requerimiento no funcional	
Testing	
Pos-condición	El usuario puede realizar cualquier acción dentro del listado

3.2.2.27 CU-26 Alta Pago

3.2.2.27.1 Formulario

3.2.2.27.2 Descripción del caso de uso

Descripción	
Nº	CU-26
Nombre	Alta Pago
Descripción	Permite al administrador dar de alta personal
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador.
Curso normal	
Usuario	Sistema
1. Presiona "Agregar" en el listado de pago	
	2. Despliega formulario de registro
3. Rellena los campos	
4. Presiona "Registrar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Ingresa el pago al sistema.
Curso alternativo	
2.1 Se presiona "Registrar" con algún campo vacío	
2.1 Se presiona "Registrar" con algún campo incorrecto	
5.1 Usuario presiona "Registrar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen"
Referencias	
R. Funcional implementado	RF26- Alta pago
Requerimiento no funcional	

Testing	
Pos-condición	Se crea un nuevo pago

3.2.2.28 CU-27 Baja Pago

3.2.2.28.1 Formulario

3.2.2.28.2 Descripción del caso de uso

Descripción	
Nº	CU-27
Nombre	Baja Pago
Descripción	Permite al administrador eliminar pagos
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador y el pago a dar de baja debe estar registrado.
Curso normal	
Usuario	Sistema
1. Presiona "Eliminar" en el pago a eliminar, dentro del listado de pagos	
	2. Elimina el pago
	3. Refresca la página 2 veces
Curso alternativo	
Referencias	
R. Funcional implementado	RF28- Baja pago
Requerimiento no funcional	
Testing	
Pos-condición	El pago seleccionado ya no existe más

3.2.2.29 CU-28 Modificar Pago

3.2.2.29.1 Formulario

3.2.2.29.2 Descripción del caso de uso

Descripción	
Nº	CU-28
Nombre	Modificar Pago
Descripción	Permite al administrador modificar el pago registrado
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser administrador.
Curso normal	
Usuario	Sistema
1.Presiona "Modificar" en el listado de pago	
	2. Despliega formulario de modificación
3. Rellena los campos	
4. Presiona "Modificar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Modifica el pago del sistema.
Curso alternativo	
2.1 Se presiona "Modificar" con algún campo vacío	
2.1 Se presiona "Modificar" con algún campo incorrecto	
5.1 Usuario presiona "Modificar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen "
Referencias	
R. Funcional implementado	RF27- Modificar pago
Requerimiento no funcional	

Testing	
Pos-condición	El pago seleccionado fue modificado

3.2.2.30 CU-29 Listado Pago

3.2.2.30.1 Formulario

3.2.2.30.2 Descripción del caso de uso

Descripción	
Nº	CU-29
Nombre	Listado Pago
Descripción	Permite al administrador visualizar la lista del pago registrado y sus datos
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe tener permiso para ver la lista
Curso normal	
Usuario	Sistema
1.Presiona "Pago" en el nav	
	2. Despliega el listado de pago
Curso alternativo	
6. Usuario no autorizado ingresa el link	
	1.1 No se le permite ingresar
	1.2 Se le redirecciona al login en caso de no estar logueado, a la página principal en caso de estar logueado en una cuenta sin permiso
Referencias	
R. Funcional implementado	RF29- Listar Pago
Requerimiento no funcional	
Testing	
Pos-condición	El usuario puede realizar cualquier acción dentro del listado

3.2.2.31 CU-30 Alta Menú

3.2.2.31.1 Formulario

3.2.2.31.2 Descripción del caso de uso

Descripción	
Nº	CU-30
Nombre	Alta Menú
Descripción	Permite al administrador dar de alta menú
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador.
Curso normal	
Usuario	Sistema
1.Presiona "Agregar" en el listado de menú	
	2. Despliega formulario de registro
3. Rellena los campos	
4. Presiona "Registrar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Ingresa el menú al sistema.
Curso alternativo	
2.1 Se presiona "Registrar" con algún campo vacío	
2.1 Se presiona "Registrar" con algún campo incorrecto	
5.1 Usuario presiona "Registrar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen "
Referencias	
R. Funcional implementado	RF37- Alta menú
Requerimiento no funcional	

Testing	
Pos-condición	Se crea un nuevo menú

3.2.2.32 CU-31 Baja Menú

3.2.2.32.1 Formulario

3.2.2.32.2 Descripción del caso de uso

Descripción	
Nº	CU-31
Nombre	Baja Menú
Descripción	Permite al administrador eliminar menú
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador y el menú a dar de baja debe estar registrado.
Curso normal	
Usuario	Sistema
1.Presiona "Eliminar" en el menú a eliminar, dentro del listado de menú	
	2. Elimina el menú
	3. Refresca la página 2 veces
Curso alternativo	
Referencias	
R. Funcional implementado	RF38- Baja menú
Requerimiento no funcional	
Testing	
Pos-condición	El menú seleccionado ya no existe más

3.2.2.33 CU-32 Modificar Menú

3.2.2.33.1 Formulario

3.2.2.33.2 Descripción del caso de uso

Descripción	
Nº	CU-32
Nombre	Modificar Menú
Descripción	Permite al administrador modificar el menú registrado
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser administrador.
Curso normal	
Usuario	Sistema
1. Presiona "Modificar" en el listado de menú	
	2. Despliega formulario de modificación
3. Rellena los campos	
4. Presiona "Modificar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Modifica el menú del sistema.
Curso alternativo	
2.1 Se presiona "Modificar" con algún campo vacío	
2.1 Se presiona "Modificar" con algún campo incorrecto	
5.1 Usuario presiona "Modificar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen "
Referencias	
R. Funcional implementado	RF40- Modificar menú
Requerimiento no funcional	

Testing	
Pos-condición	El menú seleccionado fue modificado

3.2.2.34 CU-33 Listado Menú

3.2.2.34.1 Formulario

3.2.2.34.2 Descripción del caso de uso

Descripción	
Nº	CU-33
Nombre	Listado Menú
Descripción	Permite al administrador visualizar la lista del menú registrado y sus datos
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe tener permiso para ver la lista
Curso normal	
Usuario	Sistema
1.Presiona "Menú" en el nav	
	2. Despliega el listado de menú
Curso alternativo	
7. Usuario no autorizado ingresa el link	
	1.1 No se le permite ingresar
	1.2 Se le redirecciona al login en caso de no estar logueado, a la página principal en caso de estar logueado en una cuenta sin permiso
Referencias	
R. Funcional implementado	RF39- Listar menú
Requerimiento no funcional	
Testing	
Pos-condición	El usuario puede realizar cualquier acción dentro del listado

3.2.2.35 CU-34 Alta Reserva

3.2.2.35.1 Formulario

FIG84. Formulario Alta Reserva

3.2.2.35.2 Descripción del caso de uso

Descripción	
Nº	CU-34
Nombre	Alta Reserva
Descripción	Permite al administrador dar de alta reserva
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador.
Curso normal	
Usuario	Sistema
1.Presiona "Agregar" en el listado de reserva	
	2. Despliega formulario de registro
3. Rellena los campos	
4. Presiona "Registrar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Ingresa la reserva al sistema.

Curso alternativo	
2.1 Se presiona "Registrar" con algún campo vacío	
2.1 Se presiona "Registrar" con algún campo incorrecto	
5.1 Usuario presiona "Registrar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen"
Referencias	
R. Funcional implementado	RF43- Alta reserva
Requerimiento no funcional	
Testing	
Pos-condición	Se crea una nueva reserva

3.2.2.36 CU-35 Baja Reserva

3.2.2.36.1 Formulario



FIG85. Formulario Baja Reserva

3.2.2.36.2 Descripción del caso de uso

Descripción	
Nº	CU-35
Nombre	Baja Reserva
Descripción	Permite al administrador eliminar reservas
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador y la reserva a dar de baja debe estar registrada.
Curso normal	
Usuario	Sistema
1.Presiona "Eliminar" en la reserva a eliminar, dentro del listado de reservas	
	2. Elimina la reserva
	3. Refresca la página 2 veces
Curso alternativo	
Referencias	
R. Funcional implementado	RF44- Baja reserva
Requerimiento no funcional	
Testing	
Pos-condición	La reserva seleccionada ya no existe más

3.2.2.37 CU-36 Modificar Reserva

3.2.2.37.1 Formulario

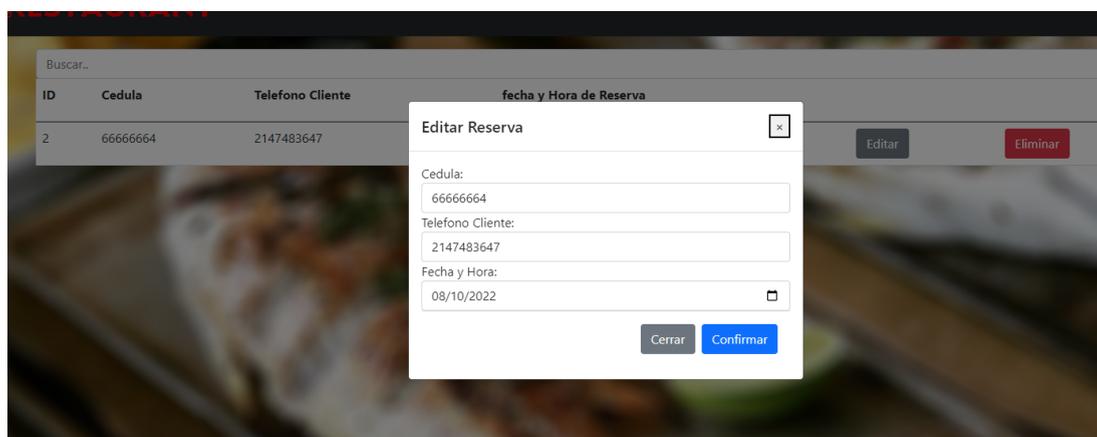


FIG86. Formulario modificar Reserva

3.2.2.37.2 Descripción del caso de uso

Descripción	
Nº	CU-36
Nombre	Modificar Reserva
Descripción	Permite al administrador modificar la reserva registrada
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser administrador.
Curso normal	
Usuario	Sistema
1. Presiona "Modificar" en el listado de reserva	
	2. Despliega formulario de modificación
3. Rellena los campos	
4. Presiona "Modificar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Modifica la reserva del sistema.
Curso alternativo	

2.1 Se presiona "Modificar" con algún campo vacío	
2.1 Se presiona "Modificar" con algún campo incorrecto	
5.1 Usuario presiona "Modificar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen"

Referencias

R. Funcional implementado	RF45- Modificar reserva
Requerimiento no funcional	
Testing	
Pos-condición	La reserva seleccionada fue modificada

3.2.2.38 CU-37 Listado Reserva

3.2.2.38.1 Formulario



FIG87. Formulario listar Reserva

3.2.2.38.2 Descripción del caso de uso

Descripción	
Nº	CU-37
Nombre	Listado Reserva
Descripción	Permite al administrador visualizar la lista de reserva registrada y sus datos
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe tener permiso para ver la lista
Curso normal	
Usuario	Sistema
1.Presiona "Reserva" en el nav	
	2. Despliega el listado de reserva
Curso alternativo	
8. Usuario no autorizado ingresa el link	
	1.1 No se le permite ingresar
	1.2 Se le redirecciona al login en caso de no estar logueado, a la página principal en caso de estar logueado en una cuenta sin permiso
Referencias	
R. Funcional implementado	RF46- Listar reserva

Requerimiento no funcional	
Testing	
Pos-condición	El usuario puede realizar cualquier acción dentro del listado

3.2.2.39 CU-38 Alta Estación

3.2.2.39.1 Formulario

3.2.2.39.2 Descripción del caso de uso

Descripción	
Nº	CU-38
Nombre	Alta Estación
Descripción	Permite al administrador dar de alta estaciones
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador.
Curso normal	
Usuario	Sistema
1. Presiona "Agregar" en el listado de estación	
	2. Despliega formulario de registro
3. Rellena los campos	
4. Presiona "Registrar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Ingresa la estación al sistema.
Curso alternativo	
2.1 Se presiona "Registrar" con algún campo vacío	
2.1 Se presiona "Registrar" con algún campo incorrecto	
5.1 Usuario presiona "Registrar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen "
Referencias	
R. Funcional implementado	RF51- Alta estación
Requerimiento no funcional	

Testing	
Pos-condición	Se crea una nueva estación

3.2.2.40 CU-39 Baja Estación

3.2.2.40.1 Formulario

3.2.2.40.2 Descripción del caso de uso

Descripción	
Nº	CU-39
Nombre	Baja Estación
Descripción	Permite al administrador eliminar estaciones
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador y la estación a dar de baja debe estar registrada.
Curso normal	
Usuario	Sistema
1.Presiona "Eliminar" en la estación a eliminar, dentro del listado de estaciones.	
	2. Elimina la estación
	3. Refresca la página 2 veces
Curso alternativo	
Referencias	
R. Funcional implementado	RF52- Baja estación
Requerimiento no funcional	
Testing	
Pos-condición	La estación seleccionada ya no existe más

3.2.2.41 CU-40 Modificar Estación

3.2.2.41.1 Formulario

3.2.2.41.2 Descripción del caso de uso

Descripción	
Nº	CU-40
Nombre	Modificar Estación
Descripción	Permite al administrador modificar las estaciones registradas
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser administrador.
Curso normal	
Usuario	Sistema
1.Presiona "Modificar" en el listado de estación	
	2. Despliega formulario de modificación
3. Rellena los campos	
4. Presiona "Modificar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Modifica la estación del sistema.
Curso alternativo	
2.1 Se presiona "Modificar" con algún campo vacío	
2.1 Se presiona "Modificar" con algún campo incorrecto	
5.1 Usuario presiona "Modificar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen "
Referencias	
R. Funcional implementado	RF53- Modificar estación
Requerimiento no funcional	

Testing	
Pos-condición	La estación seleccionada fue modificada

3.2.2.42 CU-41 Listado Estación

3.2.2.42.1 Formulario

3.2.2.42.2 Descripción del caso de uso

Descripción	
Nº	CU-41
Nombre	Listado Estación
Descripción	Permite al administrador visualizar la lista de las estaciones registradas y sus datos
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe tener permiso para ver la lista
Curso normal	
Usuario	
1.Presiona "Estación" en el nav	
	2. Despliega el listado de estación
Curso alternativo	
9. Usuario no autorizado ingresa el link	
	1.1 No se le permite ingresar
	1.2 Se le redirecciona al login en caso de no estar logueado, a la página principal en caso de estar logueado en una cuenta sin permiso
Referencias	
R. Funcional implementado	RF54- Listar estación
Requerimiento no funcional	
Testing	
Pos-condición	El usuario puede realizar cualquier acción dentro del listado

3.2.2.43 CU-42 Alta método Pago

3.2.2.43.1 Formulario

3.2.2.43.2 Descripción del caso de uso

Descripción	
Nº	CU-42
Nombre	Alta método Pago
Descripción	Permite al administrador dar de alta métodos de pago
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador.
Curso normal	
Usuario	Sistema
1. Presiona "Agregar" en el listado de método Pago	
	2. Despliega formulario de registro
3. Rellena los campos	
4. Presiona "Registrar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Ingresar el método Pago al sistema.
Curso alternativo	
2.1 Se presiona "Registrar" con algún campo vacío	
2.1 Se presiona "Registrar" con algún campo incorrecto	
5.1 Usuario presiona "Registrar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen"
Referencias	
R. Funcional implementado	RF57- Alta método Pago
Requerimiento no funcional	

Testing	
Pos-condición	Se crea un nuevo método de pago

3.2.2.44 CU-43 Baja método Pago

3.2.2.44.1 Formulario

3.2.2.44.2 Descripción del caso de uso

Descripción	
Nº	CU-43
Nombre	Baja método Pago
Descripción	Permite al administrador eliminar métodos de pago
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser Administrador y el método Pago a dar de baja debe estar registrado..
Curso normal	
Usuario	Sistema
1.Presiona "Eliminar" en el método Pago a eliminar, dentro del listado de métodos de Pago	
	2. Elimina el método Pago
	3. Refresca la página 2 veces
Curso alternativo	
Referencias	
R. Funcional implementado	RF58- Baja método Pago
Requerimiento no funcional	
Testing	
Pos-condición	El método de pago seleccionado ya no existe más

3.2.2.45 CU-44 Modificar método Pago

3.2.2.45.1 Formulario

3.2.2.45.2 Descripción del caso de uso

Descripción	
Nº	CU-44
Nombre	Modificar método Pago
Descripción	Permite al administrador modificar los métodos de pago registrados
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe ser administrador.
Curso normal	
Usuario	Sistema
1. Presiona "Modificar" en el listado de método Pago	
	2. Despliega formulario de modificación
3. Rellena los campos	
4. Presiona "Modificar"	
	5. Verifica que los campos estén completos
	6. Verifica que los datos sean correctos
	7. Verifica que los datos ingresados no estén duplicados
	8. Modifica el método Pago del sistema.
Curso alternativo	
2.1 Se presiona "Modificar" con algún campo vacío	
2.1 Se presiona "Modificar" con algún campo incorrecto	
5.1 Usuario presiona "Modificar" sin completar los campos	
	5.2 Sistema muestra un mensaje "Campos incompletos"
6.1 Usuario ingresa datos no registrados	
	6.2 Sistema muestra un mensaje "Los datos ingresados no existen"
Referencias	
R. Funcional implementado	RF59- Modificar método Pago
Requerimiento no funcional	

Testing	
Pos-condición	El método Pago seleccionado fue modificado

3.2.2.46 CU-45 Listado método Pago

3.2.2.46.1 Formulario

3.2.2.46.2 Descripción del caso de uso

Descripción	
Nº	CU-45
Nombre	Listado método Pago
Descripción	Permite al administrador visualizar la lista de los métodos de pago registrados y sus datos
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Administrador
Pre-Condiciones (requerimientos)	El usuario logueado debe tener permiso para ver la lista
Curso normal	
Usuario	
1.Presiona "método Pago" en el nav	
	2. Despliega el listado de método Pago
Curso alternativo	
10. Usuario no autorizado ingresa el link	
	1.1 No se le permite ingresar
	1.2 Se le redirecciona al login en caso de no estar logueado, a la página principal en caso de estar logueado en una cuenta sin permiso
Referencias	
R. Funcional implementado	RF60- Listar método Pago
Requerimiento no funcional	
Testing	
Pos-condición	El usuario puede realizar cualquier acción dentro del listado

3.2.2.47 CU-46 Cierre sesión

3.2.2.47.1 Formulario

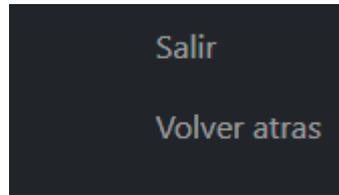


FIG88. Formulario cerrar sesión

3.2.2.47.2 Descripción del caso de uso

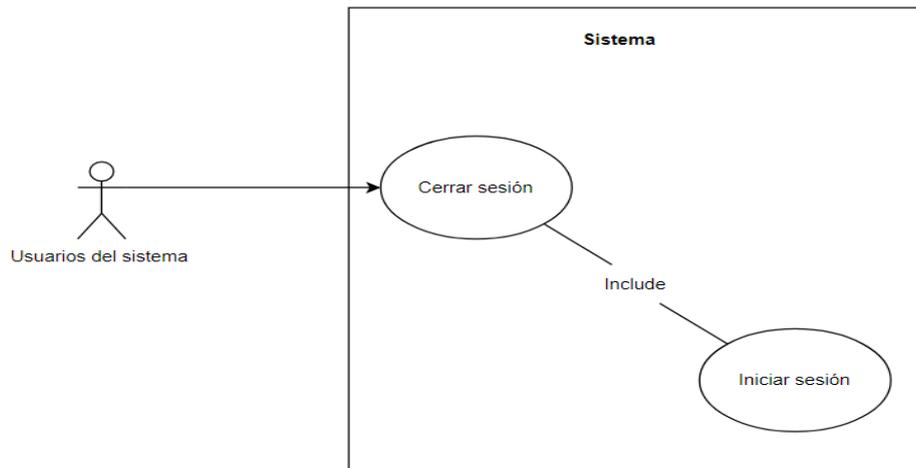


FIG89. Diagrama CU-46 Cierre de sesión

Descripción	
Nº	CU-46
Nombre	Cierre de sesión
Descripción	Permite cerrar sesión a todos los usuarios registrados en el sistemas
Tipo	Alto Nivel - Curso Normal y Alternativo
Prioridad	Alta
Actores participantes	Cualquier usuario
Pre-Condiciones (requerimientos)	El usuario a cerrar sesión debe estar previamente ingresado en el sistema.
Curso normal	
Usuario	Sistema
1.Presiona el botón “Cerrar sesión” en el NAV	

	2. Cierra la sesión del sistema
Curso alternativo	
Referencias	
R. Funcional implementado	RF75- Cerrar sesión
Requerimiento no funcional	
Testing	
Pos-condición	Se cierra la sesión

3.2.3 Base de datos

3.2.3.1 MER

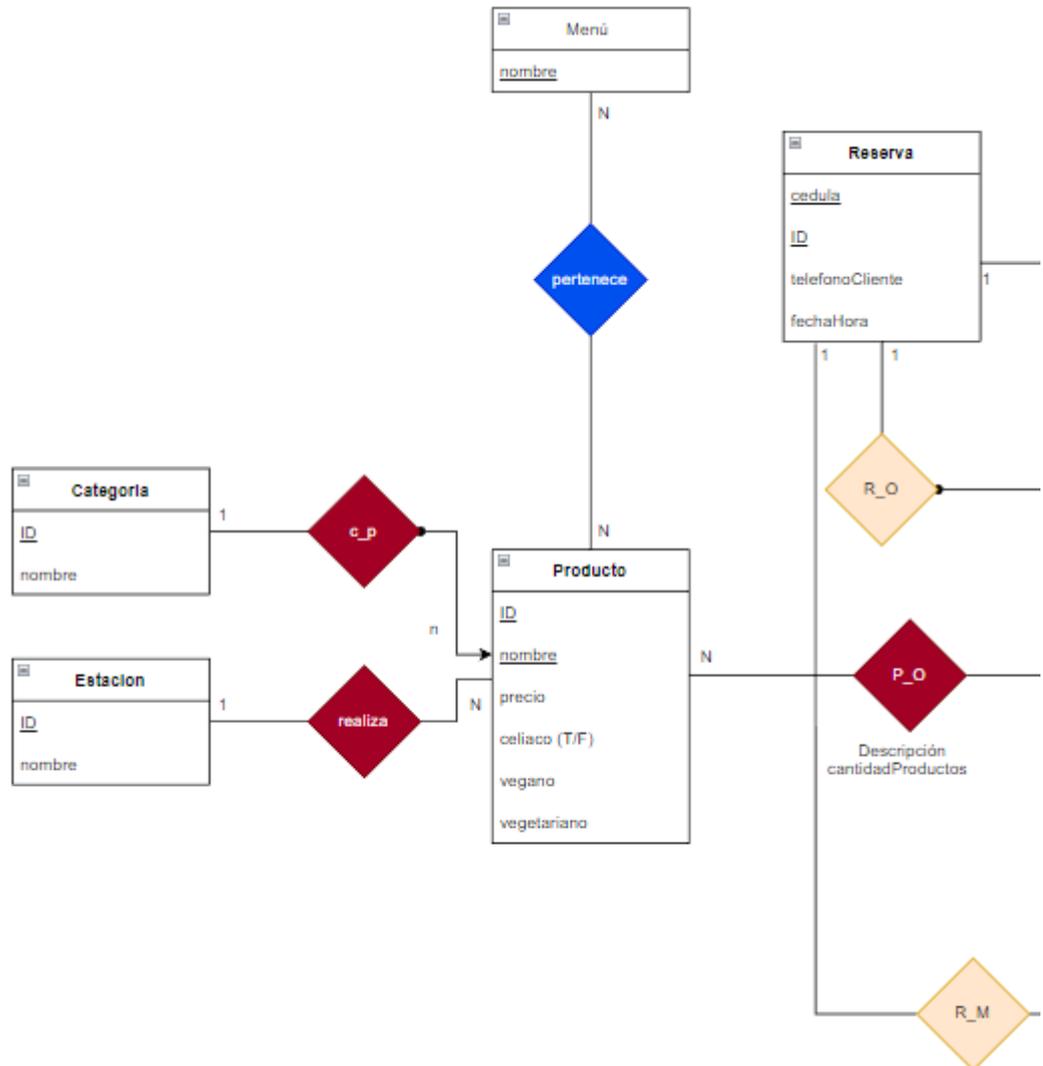


FIG90. MER

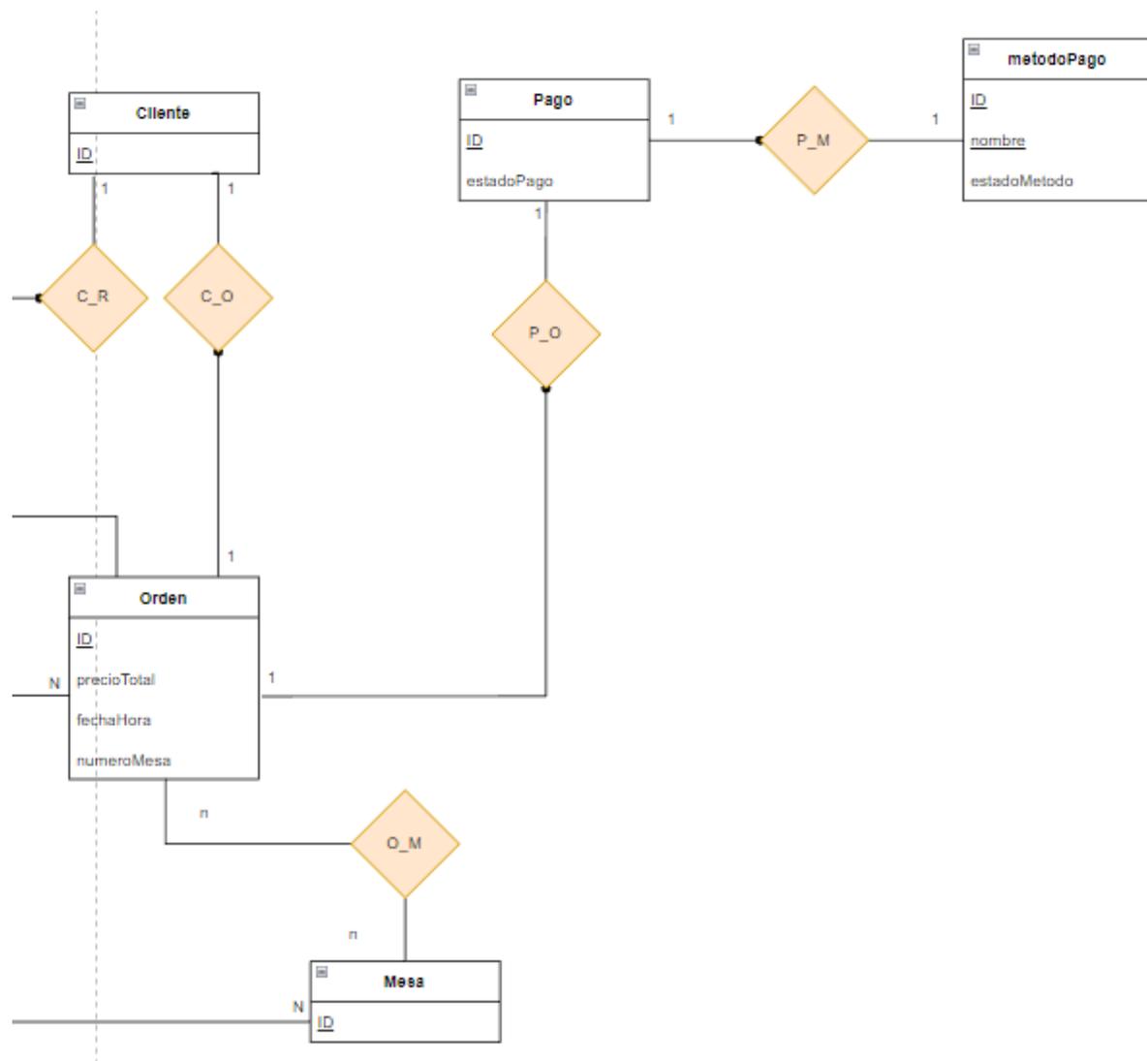


FIG91. MER 1

3.2.3.2 Tablas

Orden(ID, precioTotal, fechaHora, numeroMesa)

Producto(ID, Nombre, tipo, precio, celiaco, vegano, vegetariano)

Reserva(id, cedula, teléfonoCliente, fechaHora)

Mesa(ID)

Estación(ID, nombre)

Cliente(ID)

Pago(ID, estadoPago)

métodoPago(ID, Nombre, estadoMetodo)

Menú(nombre)

fk(Categoria) fk(Producto) fk(Producto)
C_P(ID, ID, nombre)

fk(Reserva) fk(Reserva)fk(Orden)
R_O(Cedula, IDReserva, IDOrden)

fk(Pago) fk(Orden)
P_O(IDPago, IDOrden, Descripcion, CantidadProductos)

fk(Orden) fk(Mesa)
O_M(IDOrden, IDMesa)

fk(Orden) fk(Cliente)
C_O(IDOrden, IDCliente)

fk(Cliente) fk(Reserva) fk(Reserva)
C_R(IDCliente, IDReserva, Cedula)

fk(Mesa) fk(Reserva) fk(Reserva)
R_M(IDMesa, IDReserva, Cedula)

fk(Pago) fk(MetodoPago) fk(MetodoPago)
P_M(IDPago, IDMetodoP, NombreMetodoP)

fk(Estacion) fk(Producto) fk(Producto)
Realiza(IDEstacion, IDProducto, NombreProducto)

fk(Menu) fk(Producto) fk(Producto)
Pertenece(IDMenu, IDProducto, NombreProducto)

3.2.3.3 Diccionario de Datos

Categoría

Nombre	Tipo	Tamaño	Descripción
ID	int	16	Identificador
FechaHora	date	no tiene	Controla la fecha

Estación

Nombre	Tipo	Tamaño	Descripción
ID	int	16	Identificador
Nombre	varchar	16	Identificador

Producto

Nombre	Tipo	Tamaño	Descripción
ID	int	16	Identificador
Nombre	varchar	16	Identificador
Precio	int	16	Registra precio
Celiaco	boolean	1	
Vegano	boolean	1	
Vegetariano	boolean	1	

Menú

Nombre	Tipo	Tamaño	Descripción
Nombre	varchar	16	Identificador

Reserva

Nombre	Tipo	Tamaño	Descripción
Cédula	int	16	Identificador
idReserva	int	16	Identificador
teléfonoCliente	int	16	Telefono del cliente
fechaHora	date	no tiene	Controla la fecha

Cliente

Nombre	Tipo	Tamaño	Descripción
ID	int	16	Identificador

Orden

Nombre	Tipo	Tamaño	Descripción
ID	int	16	Identificador
precioTotal	int	16	Registra precio
fechaHora	date	no tiene	Controla la fecha
númeroMesa	int	16	Identificador

Pago

Nombre	Tipo	Tamaño	Descripción
ID	int	16	Identificador
estadoPago	boolean	1	Registra estadoPago

métodoPago

Nombre	Tipo	Tamaño	Descripción
ID	int	16	Identificador
Nombre	varchar	16	Identificador
estadoMetodo	boolean	1	

Mesa

Nombre	Tipo	Tamaño	Descripción
ID	int	16	Identificador

3.3 Implementación

3.4 Gestión del proyecto

3.4.1 Control de avances de iteración 1(agosto)

3.4.1.1 Estado de situación

Pudimos implementar casi todos los requerimientos funcionales que teníamos planificados para esta iteración menos el ABML de usuarios. Básicamente el sistema de login quedó funcional y el ABML de productos también.

Nos quisimos organizar con un cronograma en formato excel donde podíamos elegir qué tareas hacer y cuándo hacerlas, pero al final no lo respetamos mucho por cuestión de no nos organizaremos para poner en común todo y por eso teníamos que hacer cambios frecuentes además de demorar las otras tareas.

Perdimos una poca parte del código por trabajar en las computadoras de la UTU y no hacer commit en el GITHUB.

En esta interacción pasamos haciendo cambios más que nada de diseño de interfaces, ideas y ponernos en común en como hacer el proyecto.

3.4.1.2 Conclusiones

Deberíamos empezar a organizarnos más y respetar el cronograma, así no tenemos que hacer tantos cambios y tambien asi nos movemos más rápido al hacer nuestras tareas, También tenemos que empezar a tener más cuidado de guardar siempre el código en GITHUB

3.4.1.3 Riesgos ocurridos

n°	riesgos	plan riesgos	de medidas tomadas
R1	Pérdida de datos	si	no se pudo recuperar la información
R2	Inclumplimiento de los tiempos planificados	si	se intentó aplicar el plan con no mucho éxito.
r9	errores en el código	si	se siguió el plan.

3.4.1.4 Modificaciones

Se modificó la cantidad de requerimientos, de 90 que teníamos nos quedamos con solo 60. Se quitó la posibilidad de delivery y de takeaway por las entrevistas que hicimos, donde el Restaurant lo de Oscar nos afirmó que no tenían delivery ni takeaway

3.4.2 Control de avances de iteración 2(septiembre)

3.4.2.1 Estado de situación

Pudimos implementar el ABML de Usuarios, el ABML de mesas, el ABML de Comandas y el ABML de reservas no nos dio el tiempo para el ABML de menú.

Nos pudimos organizar un poco mejor con el cronograma pero igualmente no podemos ajustarnos a las fechas del cronograma, esta vez porque las tareas eran muy pesadas y precisaban interacción entre en equipo.

Nos empezamos a motivar con el proyecto, ya que todos estamos haciendo lo que nos corresponde en mejor tiempo que en la primera iteración y además empezamos a entender el rumbo del proyecto.

Nos quedamos un poco atrás en la documentación en respecto al programa ya que a ninguno del equipo le gusta la documentación.

En este mes no perdimos datos, solo que siento que programamos algunos requerimientos sin análisis previo.

3.4.2.2 Conclusiones

Aunque vamos mejor que en la anterior entrega, tendríamos que meterle un poco más para llegar a los tiempos del proyecto, y ponerle las mismas ganas que le metemos en la programación a la documentación, lo bueno es que pudimos tener más cuidado con la pérdida de datos

3.4.2.3 Riesgos ocurridos

n°	riesgos	plan de riesgos	de medidas tomadas
r7	Programar un requeriment o sin una investigación previa	si	no se siguió ningún plan por falta de tiempo y por no querer tocar mucho más los requerimientos

R2	Inclumplimiento de los tiempos planificados	si	se aplicó el plan con resultados
r9	errores en el código	si	se siguió el plan.
r13	Problemas con la documentación	si	se intentó aplicar el plan con no mucho éxito

3.4.2.4 Modificaciones

Modificamos las interfaces del programa, incluimos correo como un identificador de usuario , agregamos imagen al producto, estado a mesa y a la orden, decidimos que el login distinga entre si el usuario es admin o gerente y si lo es les pide correo y contraseña, si es mozo, cocina o caja solo les pida pin.

3.4.3 Control de avances de iteración 3

3.4.3.1 Estado de situación

Pudimos implementar el ABML de Menú y algunos de los requerimientos especiales aunque nos faltaron algunos que esperamos terminar para la defensa.

En esta iteración pudimos organizar los tiempos bien, por lo tanto pudimos meterle mucho contenido a la documentación además de actualizarla, nos esforzamos en terminar la documentación antes de la entrega y meternos poco en el programa porque se nos acercaba la fecha de entrega de la documentación.

Dejamos de usar GITHUB frecuentemente porque nos corrompe los archivos y pasamos a hacernos copias de seguridad por Whatsapp

Perdimos todo el ABML del menú pero lo pudimos recuperar por una copia de seguridad.

Tuvimos problemas al pasar el programa al hosting ya que nuestro hosting no detecta bien los headers, y por eso tenemos que adaptar bastante el código.

3.4.3.2 Conclusiones

No avanzamos tanto en el Software ya que decidimos completar la documentación y actualizarla, pero pudimos organizarnos mejor con el cronograma cambiando las tareas asignadas entre nosotros, nos pareció más práctico usar Whatsapp porque git nos corrompe el código, esperamos tener el hosting para antes la defensa.

3.4.3.3 Riesgos ocurridos

n°	riesgos	plan de riesgos	medidas tomadas
r9	errores en el código	si	se siguió el plan.
R1	Pérdida de datos	si	se siguió con el plan

3.4.3.4 Modificaciones

Cambiamos de un hosting gratuito a un pago, algunos cambios al los requerimientos.

3.5 Logros

3.5.1 Requerimientos implementados

Se logró implementar el 81% de los requerimientos acordados.

- RF1. Comprobar orden
- RF2. Alta orden
- RF3. Baja orden
- RF4. Modificar orden
- RF5. Listar orden
- RF6. Buscar orden
- RF7. Comprobar mesa
- RF8. Alta mesa
- RF9. Baja mesa
- RF10. Modificar mesa
- RF11. Listar mesa
- RF12. Buscar mesa
- RF19. Alta personal
- RF20. Baja personal
- RF21. Modificar personal
- RF22. Listar personal

- RF23. Buscar personal
- RF24. Comprobar personal
- RF31. Comprobar Producto
- RF32. Alta Producto
- RF33. Baja Producto
- RF34. Modificar Producto
- RF35. Listar Producto
- RF36. Buscar Producto
- RF43. Alta reserva
- RF44. Baja reserva
- RF45. Modificar reserva
- RF46. Listar reserva
- RF47. Comprobar reserva
- RF48. Buscar reserva
- RF49. Comprobar Usuarios
- RF63. Alta usuario del sistema
- RF64. Baja usuario del sistema
- RF65. Modificar usuario del sistema
- RF66. Listar usuario del sistema
- RF67. Comprobar usuario del sistema
- RF68. Buscar usuario del sistema
- RF69. Inicio de sesión
- RF71. Categorizar productos por menú
- RF72. Categorizar productos por tipo
- RF73. Categorizar productos por aptitud (celíaco, vegano, vegetariano)
- RF74. Gestionar stock
- RF75. Cerrar sesión
- RF76. Ver menú
- RF25. Comprobar pago
- RF26. Alta pago
- RF27. Baja pago
- RF28. Modificar pago
- RF29. Listar pago
- RF30. Buscar pago
- RF37. Alta menú
- RF38. Baja menú
- RF39. Listar menú
- RF40. Modificar menú
- RF41. Comprobar menú
- RF42. Buscar menú
- RF50. Organizar Órdenes
- RF57. Alta metodo Pago
- RF58. Baja metodo Pago
- RF59. Modifica método Pago
- RF60. Listar método Pago
- RF61. Comprobar método Pago
- RF62. Buscar metodo Pago

3.5.2 Requerimientos no implementados

- RF13. Alta cliente - no aplica
- RF14. Baja cliente - no aplica
- RF15. Modificar cliente - no aplica
- RF16. Listar cliente - no aplica
- RF17. Buscar cliente - no aplica
- RF18. Comprobar cliente - no aplica
- RF51. Alta estación - no aplica
- RF52. Baja estación - no aplica
- RF53. Modificar estación - no aplica
- RF54. Listar estación - no aplica
- RF55. Comprobar estación - no aplica
- RF56. Buscar estación - no aplica
- RF77. Generar estadísticas en base a las ventas - tiempo
- RF70. Ordenar productos por precio - tiempo

3.6 Conclusiones finales

3.6.1 Facundo Rodriguez

Desde mi perspectiva, planear, programar y ejecutar el proyecto fue una experiencia muy nutritiva y en la que he aprendido más que en todos los anteriores años juntos. Fue una buena experiencia aprender a trabajar en equipo, organizar y estructurar la forma de trabajo y combinar distintas formas de programación.

Las dificultades, errores y fallas que surgieron supimos resolverlas de forma óptima y eso dió mucho de lo que aprender. Además, aprendí a optimizar el tiempo y encontrar lugares en el día para programar.

3.6.2 Diego Pozzi

La experiencia en general fue muy buena para darnos una idea del desarrollo de un proyecto de mayor importancia y aunque experimentamos dificultades al organizarnos y en general, esas cosas siempre pasan normalmente y lo importante fue que pudimos seguir el proyecto con normalidad.

Personalmente creo que este proyecto nos enseñó demasiadas cosas en todos los ámbitos, aunque me hubiera gustado que fuera más lento.

Me voy con una buena idea de este año donde los profesores dieron todo lo posible para enseñarnos y que entendiéramos temas que en realidad son bastante difíciles para aprenderlos solos.

3.6.3 Dylan Rodriguez

Esta instancia fue la mejor oportunidad para poner a prueba lo aprendido durante estos años, dando así un cierre ideal para lo que fué el bachillerato, aprendí a trabajar en equipo y todo lo que conlleva y puedo decir que me tocó trabajar con grandes personas lo cual hizo todo más llevadero.

3.6.4 Brian Matos

Esta fue una gran oportunidad para fomentar el trabajo en equipo, la realización de todo lo aprendido desde el primer año y a pesar de todos los errores y dificultades que hubieron, pudimos superarlos sin ningún problema con la ayuda de nuestros profesores y la unión de nuestro equipo.

3.6.5 Del equipo

A pesar del lento acostumbramiento al trabajo en equipo, supimos adaptarnos y consideramos que esta instancia fue muy rica para nuestro desarrollo como programadores. Aprendimos como nunca antes sobre programación, empresas, compañerismo y organización del trabajo, pero más allá de lo aprendido, valoramos enormemente la experiencia que el proyecto nos brindó.

3.7 Glosario

Software: Es un conjunto de programas y sistemas lógicos que permiten a la computadora realizar determinadas tareas.

UML: Unified Modeling Language o Lenguaje Unificado de Modelado fue creado con el fin de forjar un lenguaje de modelado visual y semántica común y para la arquitectura , el diseño y la implementación de sistemas de software complejos, tanto en estructura como en comportamiento.

FODA:FODA significa fortalezas, oportunidades, debilidades y amenazas. Es una herramienta de estudio de la situación de una empresa, institución, proyecto o persona, analizando sus características internas y su situación externa en una matriz cuadrada.

SGBD: Sistema Gestor de Base de Datos es un conjunto de programas que nos permiten gestionar bases de datos.

Requerimiento no funcional: Se trata de requisitos que no se refieren directamente a las funciones específicas suministradas por el sistema (características de usuario), sino a las propiedades del sistema: rendimiento, seguridad, disponibilidad.

MER:(Modelo-Entidad-Relación):Un modelo entidad-relación es una herramienta para el modelo de datos, la cual facilita la representación de entidades de una base de datos.

Actores del sistema:Se le llama actor a toda entidad externa al sistema que guarda una relación con este y que le demanda una funcionalidad.

Requerimiento funcional: Los requerimientos funcionales de un sistema, son aquellos que describen cualquier actividad que esta deba realizar, en otras palabras, el comportamiento o función particular de un sistema o software cuando se cumplen ciertas condiciones.

SO:(Sistema Operativo) Un sistema operativo es un conjunto de programas que permite manejar la memoria, disco, medios de almacenamiento de información y los diferentes periféricos o recursos de nuestra computadora.

Hardware:Representa los componentes físicos y tangibles de una computadora, es decir, los componentes que pueden ser vistos y tocados.

3.8 Bibliografía

Plan SQA:

<https://www.fing.edu.uy/inco/cursos/ingsoft/pis/memoria/dvd02/experiencia2009/material/grupo5/sqa/inicial/iter1/SQAPLAG5v1.1.pdf>

Cálculo de costo por puntos de función:

<https://www.fing.edu.uy/tecnoinf/mvd/cursos/ingsoft/material/teorico/is05-PuntosFuncionComp.pdf>

Informe git:

<https://www.atlassian.com/es/git/tutorials/what-is-version-control>

Anexos

3.9 Equipo

3.9.1 Inscripciones al equipo

Dylan Rodriguez - 30/4
Diego Pozzi - 30/4
Brian Matos - 30/4
Facundo Rodriguez - 30/4

3.9.2 Reglamento del equipo

- Todos los integrantes deberán estar presentes en una reunión, virtual o presencial semanal, la cual tendrá su fecha y horario a definir todas las semanas.
- Cada avance en el proyecto tendrá que ser documentado en la bitácora y en un archivo de texto compartido en forma de bosquejo.
- Avances a nivel de software serán publicados en Git.
- Las inasistencias a la UTU deberán ser notificadas en el grupo de WhatsApp.
- Procurar avanzar individualmente a la par del equipo.
- Poner como prioridad el proyecto en conjunto.
- Se asignará un delegado en caso de falta.
- Cumplir en tiempo y forma con lo solicitado por el docente, en caso de no ser cumplido, hacérselo saber a el docente y a los otros integrantes.
- Cualquier idea deberá ser discutida en grupo antes de ser aplicada.
- Cumplir con los requisitos del proyecto.
- Cualquier cambio a nivel de software debe ser documentado y explicado.
- Cada avance debe ser reflejado en un UML común.
- Cualquier cambio en el equipo deberá ser notificado a un docente.

INFRACCIONES

El cumplimiento de los siguientes ítems de esta lista derivará en una infracción:

- Faltar a una reunión o clase, sin justificar.
- No cumplir con el cronograma de la documentación.
- Emplear indebidamente los espacios en común, (WhatsApp, Discord, Drive, Git, etc).

- Realizar cambios importantes en el software sin consultar al grupo.

Sanciones

- **1 infracción:** Llamada de atención.
- **2 infracciones:** Tarea extra y explicación al equipo(en caso de no darlo deriva en otra infracción).
- **3 infracciones:** Reunión con docente encargado del proyecto.

Historial de infracciones

Fecha	Integrante	Razón
-------	------------	-------

3.9.3 Desarrollo de las actividades

<u>Nro Entrada</u>	<u>Fecha</u>	<u>Integrantes</u>	<u>Actividad</u>
1	23/04	Dylan, Brian, Diego	Se formó el grupo
2	23/04	Dylan	Se creó grupo de WhatsApp
3	30/04	Dylan, Brian, Diego	Se definió en grupo y se creó el Reglamento
4	30/04	Dylan	Se definió el estilo de la documentación
5	30/04	Dylan, Brian, Diego, Facundo	Se agregó a un miembro
6	30/04	Todos	Se realizó la inscripción del equipo

Mayo

7	5/05	Brian, Diego, Facundo, Dylan	Definición de: -nombre de la
---	------	---------------------------------	---------------------------------

			empresa -razón social -misión -visión -valores -objetivos
8	5/05	Facundo	Se eligió un logo
10	7/05	Diego, Dylan, Brian, Facundo	Se definieron las preguntas para el formulario de entrevista
11	8/05	Todos	Se definió la forma jurídica de la empresa
12	17/05	Dylan	Se eligió aplicación para la gestión y el cronograma
13	17/05	Dylan	Se creó base del cronograma
14	17/05	Facundo	Se pasaron en limpio los datos recogidos en la entrevista con el cliente.
15	19/05	Diego, Dylan, Brian, Facundo	Creación de inv.previa
16	21/05	Diego, Dylan, Brian, Facundo	Finalización de inv.previa
17	26/05	Diego, Dylan, Brian, Facundo	Realizamos el contrato de formación de la empresa.
18	26/06	Facundo	Se registró el plano del local

Junio

19	02/06	Dylan, Brian	Se levantó papeleo necesario en la Intendencia Departamental
20	03/06	Diego, Dylan, Brian, Facundo	Terminamos los formularios legales (Bps, dgi, etc).
21	04/06	Diego, Dylan, Brian, Facundo	Realizamos esquema y diagrama del funcionamiento del software.
22	04/06	Dylan	Entrega primer avance ADA
23	07/06	Diego, Brian, Dylan	Se relleno formulario de la intendencia
24	07/06	Dylan	Se definió el emplazamiento y ubicación geográfica
25	07/06	Todos	Se definió el ramo de la empresa
26	10/06	Diego, Brian	Creacion de gestión de riesgos
27	11/06	Todos	Creación del plan de proyecto
28	12/06	Todos	Se terminó el plan de proyecto
29	13/06	Facundo	Se documentó la descripción del cliente
30	14/06	Dylan	Se definieron los procesos

31	15/06	Dylan	Se crearon los diagramas de los procesos
32	15/06	Facundo	Creación de organigrama
33	16/06	Facundo, Brian	Se definieron los actores involucrados
34	17/06	Todos	Se creó la descripción de entorno
35	17/06	Dylan, Brian	Realización de mockup de página web
36	17/06	Diego	Realización de avance de S.O
37	17/06	Dylan, Brian	Se empezó a hacer 2do avance
38	18/06	Facundo	Creación de Gestión de Riesgos
39	18/06	Dylan	Finalización de gestión de riesgo
40	19/06	Todos	Creación de la lista de necesidades
41	20/06	Dylan, Brian	Creación lista de requerimientos funcionales e infuncionales
42	21/06	Diego	Se realizó el análisis de Sistemas Operativos
43	22/06	Diego, Brian	Justificación del S.O elegido
44	23/06	Facundo	Creación de carátula

45	23/06	Dylan, Diego, Brian	Reunión de mantenimiento con docente de ADA
46	23/06	Dylan, Diego, Brian	Rectificación de procesos
47	23/06	Dylan	Rectificación de diagrama UML
48	24/06	Diego	Estudio de alternativas de S.O
49	25/06	Dylan, Diego	Maquetar las interfaces acerca de los requerimientos dados
50	26/06	Diego, Facundo, Dylan	Implementación del formulario de Login y Menú Principal
51	28/06	Dylan, Brian	Trámites necesarios en los organismos estatales para su formación
52	29/06	Dylan, Brian, Facundo, Diego	Trámites específicos según actividad
53	30/06	Dylan, Brian, Facundo, Diego	Identificación y descripción del entorno de la empresa

Julio

54	02/07	Dylan, Facundo, Diego, Brian	Envío de primera entrega
55	26/07	Dylan	Rectificación cronograma
56	26/07	Facundo	Correcciones

			varias
57	26/07	Facundo	ADA actualizaciones
58	26/07	Dylan, Brian, Facundo, Diego	Objetivos
59	27/07	Facundo	FORM.EMP actualizaciones
60	27/07	Facundo, Dylan	Comienzo base de datos
61	27/07	Dylan, Brian, Facundo, Diego	Matrices necesidades-objetivos y objetivos requerimientos
62	30/07	Diego, Dylan, Brian, Facundo	Puesta a punto de últimos detalles antes de la entrega

Agosto

63	06/08	Dylan	Index y login con sesiones.
64	06/08	Facundo	CRUD proyecto e inicio de BD.
65	10/08	Facundo	Pruebas y conexiones a BD.
66	10/08	Facundo	Actualización Documentación.
67	10/08	Dylan	Inicio de conversión a MVC
68	11/08	Facundo	Creación prueba GIT del equipo.
69	13/08	Diego, Facundo, Brian	Mejoras varias y gráficas documentación
70	13/08	Facundo, Dylan	Análisis de

			Mercado, variables económicas y crecimiento.
71	15/08	Facundo, Dylan, Diego, Brian	Aspectos regulatorios
72	15/08	Dylan	Actualización BD.
73	16/08	Dylan	Finalización conversión a MVC
74	18/08	Todos	Rectificación MER
75	18/08	Todos	Realización MR tercera forma normal
76	20/08	Facundo	Objetivos y necesidades tabla.
77	20/08	Todos	Mejoras DOC y BD.
78	26/08	Todos	Realización Avance 2
79	01/09	Facundo	Modificación análisis y correcciones varias
80	01/09	Dylan	Avanza en BD y programación
81	02/09	Dylan, Diego, Facundo	Puesta a punto de entrega, programación y base de datos.
82	03/09	Facundo	Últimos detalles Formación, Bitácora
83	03/09	Todos	Últimos detalles entrega 2
84	10/09	Diego, Facundo	Mockup comandas

85	14/09	Dylan, Diego	Iteración 2.1
86	15/09	Dylan	Iteración 2.2
87	17/09	Facundo	Avance Mockup
88	19/09	Facundo	Terminaciones Mockup general
89	24/09	Todos	Iteración 2.3
90	01/10	Facundo, Diego	Avance y terminación FE
91	05/10	Facundo	Hosting, dominios
92	06/10	Facundo	BD en Host y conexiones. Formación Empresarial (cuentas y presupuesto)
93	10/10	Facundo	Adaptación de código a BD en HOST
94	10/10	Dylan	Iteración 2.4
95	17/10	Diego	Iteración 2.5
96	19/10	Diego	Iteración 2.6
98	20/10	Todos	Pasaje a webhost

99	21/10	Dylan	Pasaje en limpio documentación
100	22/10	Diego	Pasaje en limpio documentación
101	23/10	Facundo	Pasaje en limpio documentación
102	24/10	Brian	Pasaje en limpio documentación
103	28/10	Dylan	Pasaje en limpio documentación
104	28/10	Facundo	Optimización webhost
105	28/10	Diego	Implementación de últimos requerimientos
106	28/10	Dylan, Diego	Rectificación y adaptación entre códigos

3.10 Documentación complementaria

3.10.1 Entrevistas

Entrevista número 1: Presencial, 11 de mayo de 2022.
Participantes: Todos.

Entrevista número 2: Virtual, 5 de julio de 2022.
Participantes: Todos.

Entrevista número 3: Virtual, 28 de septiembre de 2022.
Participantes: Todos.

Detallado entrevista 1:

1) ¿En el restaurante utilizan algún software de registro de comandas?

R) No, el mozo escribe los pedidos en un papel de doble copia y los pasa a la cocina, la otra copia queda guardada para hacer las facturas.

2) ¿Qué datos tiene su comanda en papel?

R) En el pedido se detallan los platos y la mesa. En la comanda hay otros datos pero para agilizar no se rellenan.

3) ¿Le gustaría tener un software de gestión de comandas?

R) Si, sí facilita y no me la complica estaría bueno tenerlo. Ya intentamos con uno pero era caro, nos cobraban una comisión.

4) ¿Cómo fue su experiencia con ese software que probaron?

R) Si no hubiese sido tan caro lo habiéramos dejado,

5) ¿Nos permite usar su marca para desarrollar un software de gestión?

R) Sin ningún problema, les pido que me muestren los avances que tengan.

6) En su software ideal, ¿qué información se registra?

R) Nosotros tenemos que registrar la comanda para la elaboración, la mesa si se reserva, como se hace el pago y quien reserva mesas. Ya conocemos a la mayoría de clientes que reservan pero estaría bueno poder guardar sus datos completos.

7) ¿Quién quiere que tenga acceso al software?

R) En realidad todos deberíamos tener acceso, pero se puede hacer que por ejemplo, ¿solo pueda hacer reservas la caja?

RT) Si, podemos hacer diferentes accesos para distintos usuarios con el objetivo de dividir los poderes que tiene cada uno.

8) ¿Tiene algo más que nos quiera indicar para el funcionamiento del software?

R) Por el momento no, si se me ocurre los llamo.

Detallado entrevista 2:

- 1) Desde nuestra primera entrevista, ¿surgió alguna duda o indicación sobre el software?
R) Si, sobre lo que hablamos por teléfono, me gustaría tener una lista de las reservas y los clientes que registramos.
- 2) Entendido, ¿qué empleados quieres que tengan acceso a esa información?
R) Quiero tenerla yo (administrador), y la cajera.
- 3) ¿Quieres limitar otra información a otros usuarios?
R) No me parece necesario
- 4) Por ejemplo, ¿quieres permitir que los mozos puedan agregar o eliminar a otros empleados?
R) No, eso solo debería poder hacerlo yo
- 5) ¿Quieres indicar algo más para el desarrollo?
R) Por el momento no se me ocurre, cualquier cosa te llamo.

Detallado entrevista 3:

- 1) ¿Cómo ves el avance de la muestra que te hicimos?
R) Lo veo bien y me gusta como funciona, noto que le falta un poco de estética para que sea más intuitivo nada más.
- 2) Según lo que tenemos a la fecha, ¿te parece que será funcional a tu restaurante? R) Calculo que si, va a funcionar
- 3) ¿Tienes alguna otra sugerencia? R) Confío en ustedes, trabajen tranquilos.

3.11 Manual de Usuarios

El sistema está diseñado tal que cada usuario pueda acceder mediante la barra de navegación a los apartados pertinentes a su sector.

3.11.1 Mozo

El usuario tipo “Mozo” podrá acceder a los siguientes apartados:



FIG92. Mozo nav

En Comandas tendrá la siguiente vista:



FIG93. Comandas

En la cual podrá elegir las dos opciones relacionadas a comandas. **En generar comandas** el mozo podrá enviar comandas a la cocina con los productos que el cliente solicite.

ID	Imagen	Nombre	Tipo	Cantidad	Precio
78		Ensalada mediter	Entradas y ensal	<input type="text"/>	\$ 850
79		Ensalada de salm	Entradas y ensal	<input type="text"/>	\$ 850
80		Rabas	Entradas y ensal	<input type="text"/>	\$ 750
81		Chipirones a la	Entradas y ensal	<input type="text"/>	\$ 750
82		Camarones al aji	Entradas y ensal	<input type="text"/>	\$ 950
83		Jamon crudo con	Entradas y ensal	<input type="text"/>	\$ 750
84		Brotola grille	Principales	<input type="text"/>	\$ 850

FIG94. Generar Comandas

Se podrá ver un listado de los productos del restaurante, platos, bebidas, postres, etc.
El mozo deberá indicar la cantidad solicitada por el cliente de cada producto.

Rabas	Entradas y ensal	<input type="text" value="4"/>
Chipirones a la	Entradas y ensal	<input type="text" value="1"/>

Fig 43. Generar Comandas view 2

Luego elegir la mesa la cual va a recibir la comanda

FIG95. Generar Comandas view 3

Y darle a enviar para que la comanda sea enviada a la cocina.

En ver comandas el mozo podrá ver las comandas ya preparadas listas para entregar a la mesa

ID	Precio	Fecha-Hora	Mesa	Estado Orden	
250	0	2022-10-30 9:01	1	Hecho	<input type="button" value="Entregado"/>
251	0	2022-10-30 9:01	5	Hecho	<input type="button" value="Entregado"/>

FIG96. Ver Comandas view

Al darle “Entregado” se le desplegará el formulario de pago en el cual especificará el método de pago, el valor de la orden se registrará automáticamente.

En Mesas, el mozo podrá ver el estado de las mesas.



FIG97. Mesas view

3.11.2 Gerente

El tipo de usuario “Gerente” podrá acceder a los siguientes apartados:



FIG98. Gerente nav

En “Productos” podrá ver un listado de los productos y su información

Buscar...								
Imagen	ID	Nombre	Tipo	Precio	Celiaco	Vegano	Vegetariano	Agregar
	78	Ensalada mediter	Entradas y ensal	850	1	1	1	Editar Eliminar
	79	Ensalada de salm	Entradas y ensal	850	1	0	0	Editar Eliminar
	80	Rabas	Entradas y ensal	750	0	0	0	Editar Eliminar
	81	Chipirones a la	Entradas y ensal	750	1	0	0	Editar Eliminar
	82	Camarones al aji	Entradas y ensal	950	1	0	0	Editar Eliminar
	83	Jamon crudo con	Entradas y ensal	750	1	0	0	Editar Eliminar
	84	Brotola grille	Principales	850	1	0	0	Editar Eliminar

FIG99. Productos view

Al presionar eliminar el gerente podrá eliminar los productos, los cuales van a dejar de estar disponibles.

Al darle editar se desplegará el formulario de edición

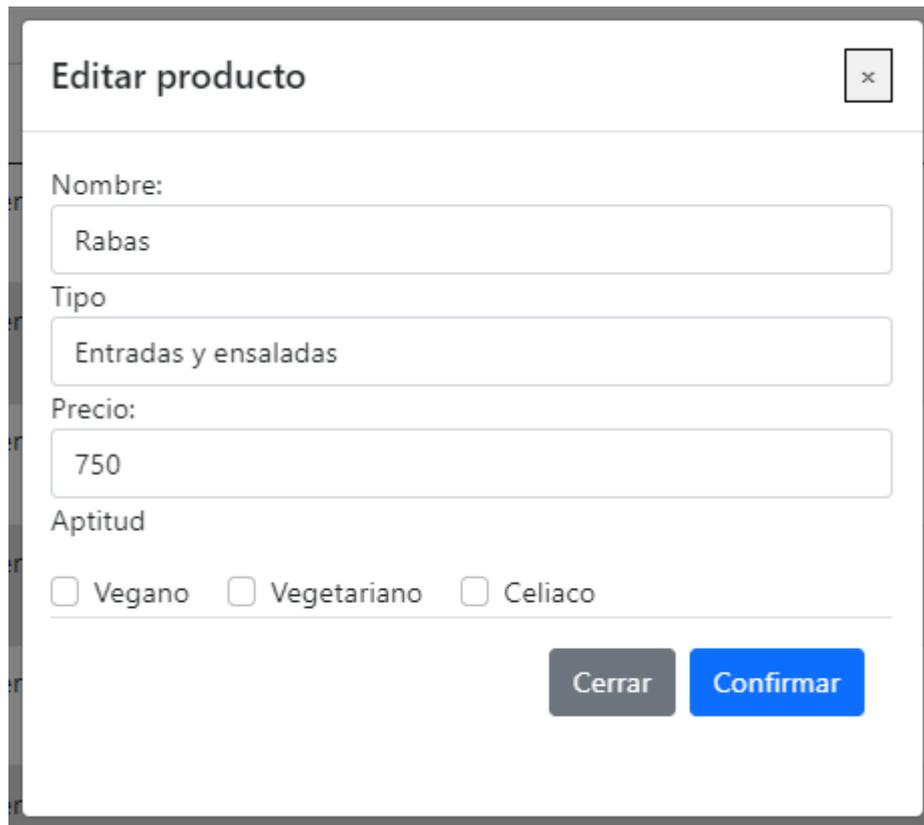


FIG100. Editar Productos

En el cual podrá cambiar la información del producto, el nombre, el tipo que es un listado

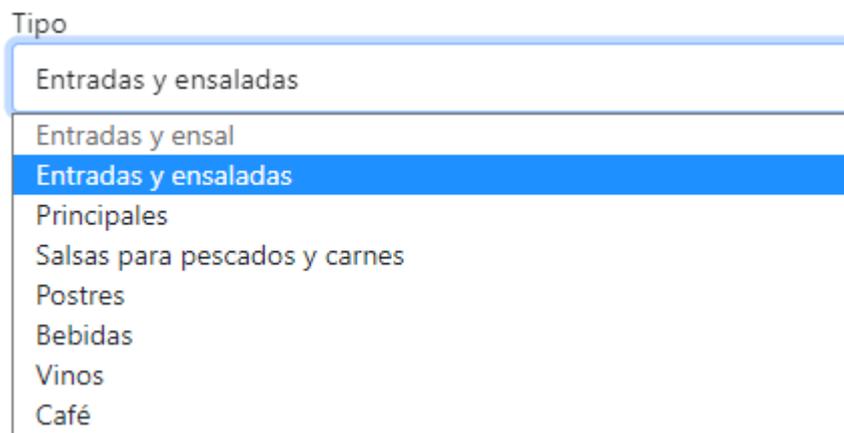


FIG101. Editar Productos 1

el precio en el cual podrá digitar solo números y las aptitudes del producto.

Al presionar agregar se desplegará el formulario de alta de productos

Alta productos

Imagen **Elegir archivos** Ninguno archivo selec.

Nombre
Pancho

Tipo
Principales

Precio
120

Aptitud
 Vegano Vegetariano Celiaco

Agregar

FIG102. Alta productos

El gerente podrá agregarle una imagen al producto la cual deberá estar descargada en el dispositivo en uso

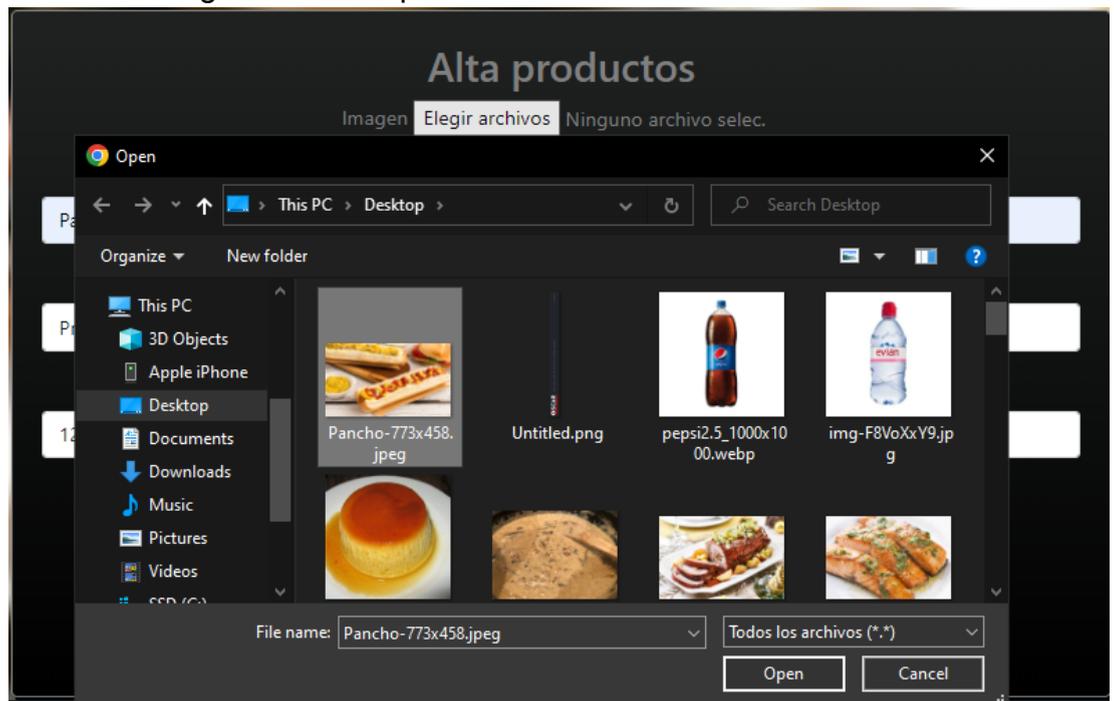


FIG103. Alta productos 1

Luego rellenará los campos como en el apartado de edición antes mencionado, le dará a agregar y el producto se añadirá a la lista.

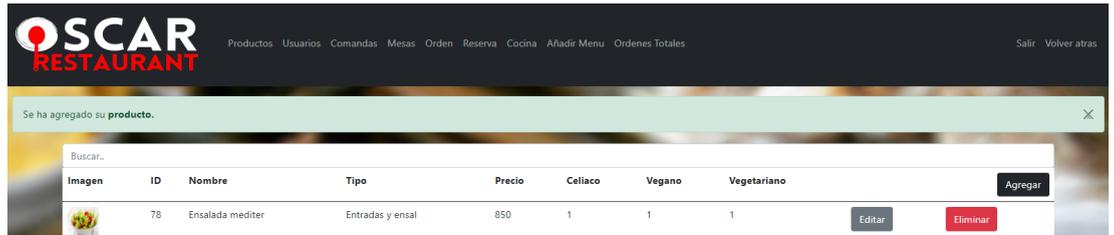


FIG104. Alta productos 2

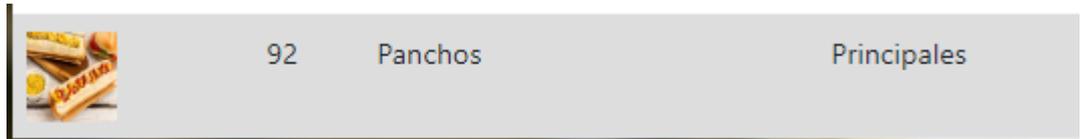


FIG105. Alta productos 3

3.11.3 Cocina

El tipo de usuario “Gerente” podrá acceder a los siguientes apartados:



FIG106. Cocina nav

Dentro de “Cocina” el cocinero podrá ver las órdenes enviadas por el mozo

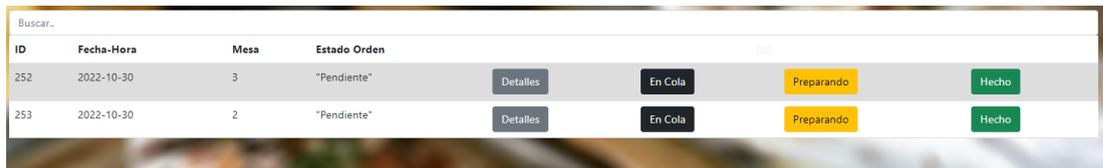


FIG107. Cocina

Al presionar en una comanda, accederá al listado de los productos de la orden, con su cantidad y detalles en caso de tenerlos

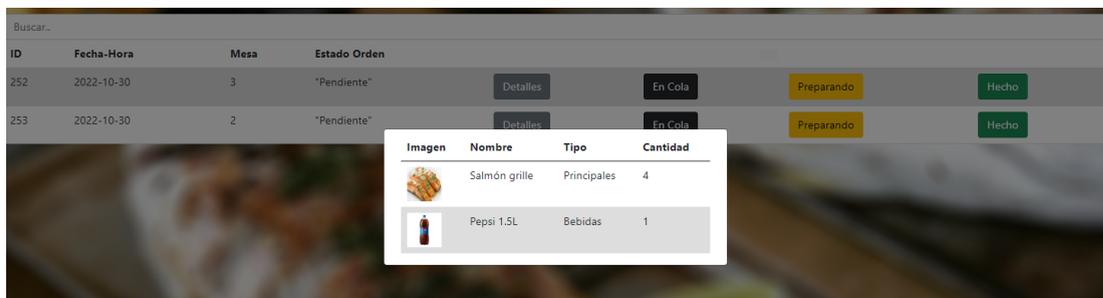


FIG108. Cocina 1

En los botones



FIG109. Cocina 2

El cocinero deberá presionar Preparando cuando empiece a preparar la orden y hecho cuando ya se haya realizado, para que el mozo vaya a retirarla.

3.11.4 Caja

El tipo de usuario “Caja” podrá acceder a los siguientes apartados:



FIG110. Caja nav

En Comandas tendrá la siguiente vista:

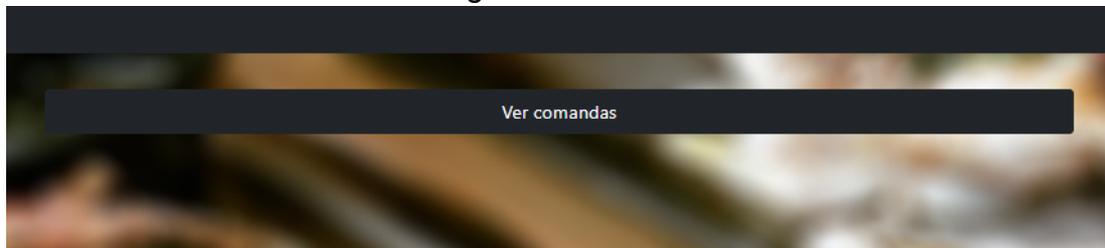


FIG111. Comandas

En ver comandas el el cajero podrá ver las comandas ya preparadas listas para entregar a la mesa

ID	Precio	Fecha-Hora	Mesa	Estado Orden	
250	0	2022-10-30 9:01	1	Hecho	Entregado
251	0	2022-10-30 9:01	5	Hecho	Entregado

FIG112. Ver Comandas

Al darle “Entregado” se le desplegará el formulario de pago en el cual especificará el método de pago, el valor de la orden se registrará automáticamente.

3.11.5 Administrador

El tipo de usuario “Administrador” podrá acceder a los siguientes apartados:

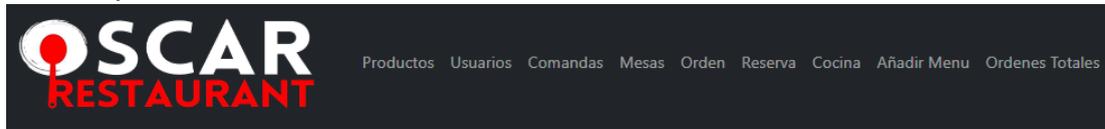


FIG113. Administrador nav

En “Productos” podrá ver un listado de los productos y su información

Imagen	ID	Nombre	Tipo	Precio	Celiaco	Vegano	Vegetariano	
	78	Ensalada mediter	Entradas y ensal	850	1	1	1	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
	79	Ensalada de salm	Entradas y ensal	850	1	0	0	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
	80	Rabas	Entradas y ensal	750	0	0	0	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
	81	Chipirones a la	Entradas y ensal	750	1	0	0	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
	82	Camarones al aji	Entradas y ensal	950	1	0	0	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
	83	Jamon crudo con	Entradas y ensal	750	1	0	0	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
	84	Brotola grille	Principales	850	1	0	0	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>

FIG114. Productos

Al presionar eliminar el administrador podrá eliminar los productos, los cuales van a dejar de estar disponibles.

Al darle editar se desplegará el formulario de edición

Editar producto ✕

Nombre:

Tipo

Precio:

Aptitud

Vegano
 Vegetariano
 Celiaco

FIG115. Editar Productos

En el cual podrá cambiar la información del producto, el nombre, el tipo que es un listado

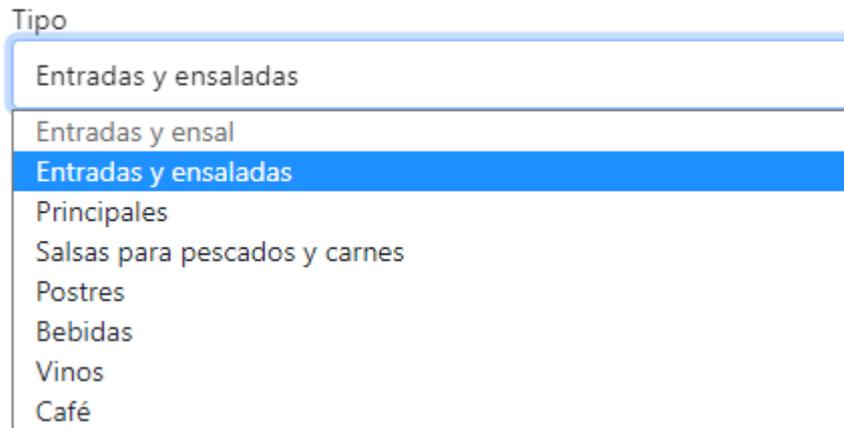


FIG116. Editar Productos 1

el precio en el cual podrá digitar solo números y las aptitudes del producto.

Al presionar agregar se desplegará el formulario de alta de productos

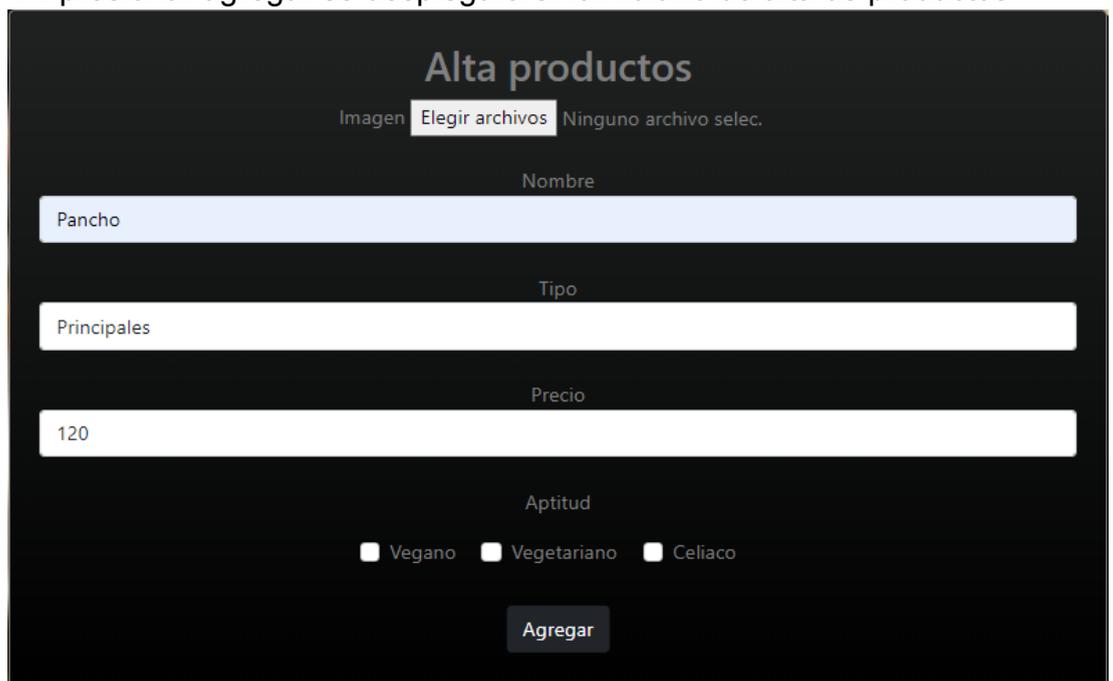
A screenshot of a form titled "Alta productos" on a dark background. The form contains several input fields and checkboxes. At the top, there is a label "Imagen" followed by a button "Elegir archivos" and the text "Ninguno archivo selec.". Below this is a "Nombre" field with the value "Pancho". The "Tipo" field has the value "Principales". The "Precio" field has the value "120". At the bottom, there are three checkboxes for "Aptitud": "Vegano", "Vegetariano", and "Celiaco", all of which are currently unchecked. A button labeled "Agregar" is positioned at the bottom center of the form.

FIG117. Alta productos

El administrador podrá agregar una imagen al producto la cual deberá estar descargada en el dispositivo en uso

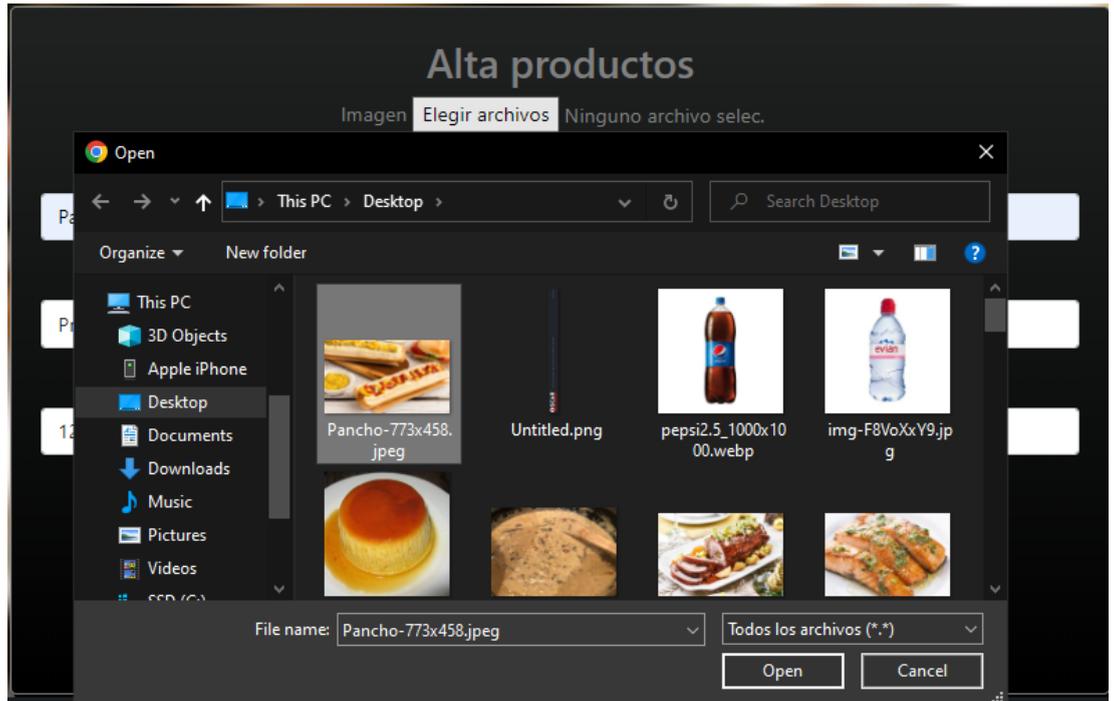


FIG118. Alta productos 1

Luego rellenará los campos como en el apartado de edición antes mencionado, le dará a agregar y el producto se añadirá a la lista.

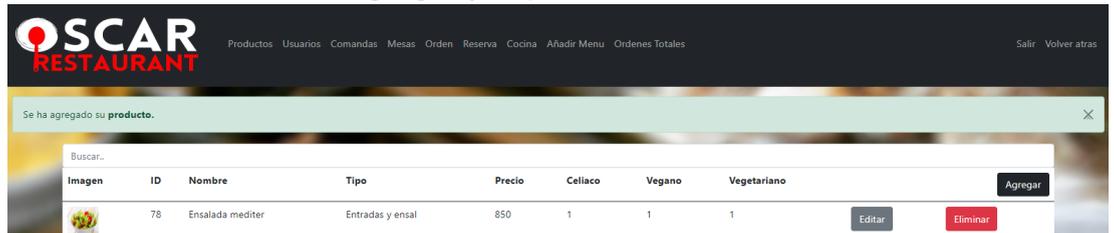


FIG119. Alta productos 2

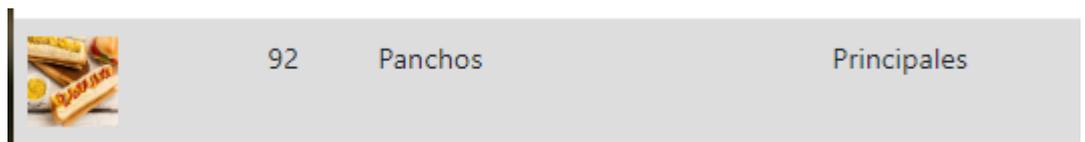


FIG120. Alta productos 3

En “**Usuarios**” podrá ver un listado de los usuarios del sistema y su información

correo	Tipo	Contraseña	Pin	estadoRecuperar	
brianmatos12@gmail.com	Administrador	asd		usuario OK	Editar Eliminar
dylanrodriguez@gmail.com	Mozo		123	usuario OK	Editar Eliminar
facundor@outlook.com	Gerente	chocolate123		usuario OK	Editar Eliminar
diegopozzi@hotmail.com	Cocina		1234554321	usuario OK	Editar Eliminar
admin@gmail.com	Administrador	admin		usuario OK	Editar Eliminar
jorge@gmail.com	Caja		123123	usuario OK	Editar Eliminar

FIG121. Usuarios

Al presionar editar se desplegará el formulario de edición de usuario

Editar usuario ×

Email

Cargo

Cerrar
Confirmar

FIG122. Editar Usuarios

Que dependiendo del cargo elegido, se desplegará un campo de PIN o de Contraseña

The screenshot shows a modal window titled "Editar usuario" with a close button (x) in the top right corner. The form contains three input fields: "Email" with the value "brianmatos12@gmail.com", "Cargo" with the value "Cocina", and "PIN" with the placeholder text "Ingrese PIN". At the bottom right, there are two buttons: "Cerrar" (grey) and "Confirmar" (blue). The background shows a blurred interface with labels "Contraseña", "Pin", and "estadoRe". The text "admin" is visible at the bottom left of the modal.

FIG123. Editar Usuarios 1

The screenshot shows a modal window titled "Editar usuario" with a close button (x) in the top right corner. The form contains three input fields: "Email" with the value "brianmatos12@gmail.com", "Cargo" with the value "Administrador", and a field labeled "pass:" which is currently empty. At the bottom right, there are two buttons: "Cerrar" (grey) and "Confirmar" (blue). The background shows a blurred interface with labels "Contraseña", "Pin", and "estadoRe". The text "admin" is visible at the bottom left of the modal.

FIG124. Editar Usuarios 2

En Comandas tendrá la siguiente vista:

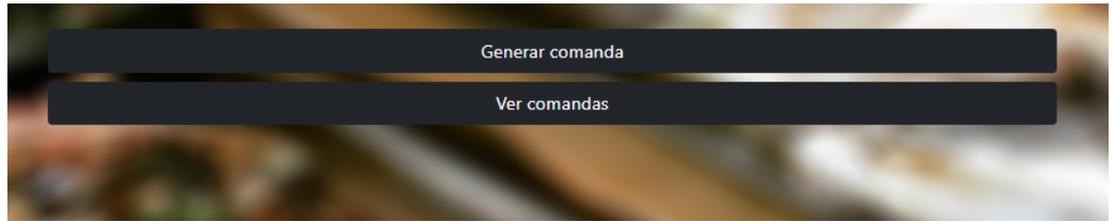


FIG125. Comandas

En la cual podrá elegir las dos opciones relacionadas a comandas. **En generar comandas** el administrador podrá enviar comandas a la cocina con los productos que el cliente solicite.

ID	Imagen	Nombre	Tipo	Cantidad	Precio
78		Ensalada mediter	Entradas y ensal	<input type="text"/>	\$ 850
79		Ensalada de salm	Entradas y ensal	<input type="text"/>	\$ 850
80		Rabas	Entradas y ensal	<input type="text"/>	\$ 750
81		Chipirones a la	Entradas y ensal	<input type="text"/>	\$ 750
82		Camarones al aji	Entradas y ensal	<input type="text"/>	\$ 950
83		Jamon crudo con	Entradas y ensal	<input type="text"/>	\$ 750
84		Brotola grille	Principales	<input type="text"/>	\$ 850

FIG126. Generar Comandas

Se podrá ver un listado de los productos del restaurante, platos, bebidas, postres, etc. El mozo deberá indicar la cantidad solicitada por el cliente de cada producto.

Rabas	Entradas y ensal	<input type="text" value="4"/>
Chipirones a la	Entradas y ensal	<input type="text" value="1"/>

FIG127. Generar Comandas 2

Luego elegir la mesa la cual va a recibir la comanda



FIG128. Generar Comandas 3

Y darle a enviar para que la comanda sea enviada a la cocina. **En ver comandas** el administrador podrá ver las comandas ya preparadas listas para entregar a la mesa

ID	Precio	Fecha-Hora	Mesa	Estado Orden	
250	0	2022-10-30 9:01	1	Hecho	Entregado
251	0	2022-10-30 9:01	5	Hecho	Entregado

FIG129. Ver Comandas

Al darle “Entregado” se le desplegará el formulario de pago en el cual especificará el método de pago, el valor de la orden se registrará automáticamente.

En “Mesas” el administrador verá un listado de las mesas y el estado

Buscar..				Agregar
id	Estado de la Mesa			
1	Ocupada	Editar	Eliminar	
2	Ocupada	Editar	Eliminar	
3	Ocupada	Editar	Eliminar	
4	Libre	Editar	Eliminar	
5	Ocupada	Editar	Eliminar	

FIG130. Ver Mesas

Al presionar Editar podrá editar el estado de la mesa

Editar mesa

×

Cargo

Ocupada

Cerrar
Confirmar

FIG131. Editar Mesas

Al presionar agregar se desplegará el formulario de agregar mesa

Agregar mesa

Estado de la mesa

Elija estado Mesa

Agregar

FIG132. Agregar Mesas

En el cual deberá elegir el estado de la mesa y esta se agregará con el último número

En Mesas, el administrador podrá ver el estado de las mesas.



FIG133. Mesas

En “Reserva” el administrador podrá ver un listado de las reservas con su información

ID	Cedula	Telefono Cliente	fecha y Hora de Reserva	
2	66666664	2147483647	2022-10-08	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>

FIG134. Reservas

Al presionar editar se desplegará el formulario de edición el cual rellenará acorde a lo que necesite

Editar Reserva ✕

Cedula:

Telefono Cliente:

Fecha y Hora:
 📅

FIG135. Editar Reservas

Al presionar agregar se desplegará el formulario de alta de reserva

FIG136. Alta Reservas

el cual deberá rellenar con la información especificada.

Dentro de “Cocina” el administrador podrá ver las órdenes enviadas por el mozo

ID	Fecha-Hora	Mesa	Estado Orden
252	2022-10-30	3	"Pendiente"
253	2022-10-30	2	"Pendiente"

FIG137. Cocina

Al presionar en una comanda, accederá al listado de los productos de la orden, con su cantidad y detalles en caso de tenerlos

ID	Fecha-Hora	Mesa	Estado Orden
252	2022-10-30	3	"Pendiente"
253	2022-10-30	2	"Pendiente"

Imagen	Nombre	Tipo	Cantidad
	Salmón grille	Principales	4
	Pepsi 1.5L	Bebidas	1

FIG138. Cocina 1

En los botones



FIG139. Cocina 2

El administrador deberá presionar Preparando cuando empiece a preparar la orden y hecho cuando ya se haya realizado, para que el mozo vaya a retirarla.



Buscar...						Filtrar	
+	ID	Precio	Fecha-Hora	Mesa	Estado Orden		
+	250	0	2022-10-30	1	Hecho	Detalles	Eliminar
+	251	0	2022-10-30	5	Hecho	Detalles	Eliminar
+	252	0	2022-10-30	3	"Pendiente"	Detalles	Eliminar
+	253	0	2022-10-30	2	"Pendiente"	Detalles	Eliminar

FIG140. Cocina 3