

# UNIVERSIDAD PÚBLICA DE EL ALTO

## CARRERA INGENIERÍA DE SISTEMAS



### PROYECTO DE GRADO

## “SISTEMA WEB PARA ADMINISTRACIÓN EN SERVICIOS DE TRADUCCIÓN”

### CASO: CONSULTORIO LINGÜÍSTICO DE LA CARRERA LINGÜÍSTICA E IDIOMAS

Para optar al Título de Licenciatura en Ingeniería de Sistemas  
**MENCIÓN: GESTIÓN y PRODUCCIÓN**

**Postulante** : Univ. Jhenny Celia Cusicanqui Maldonado  
**Tutor Metodológico** : Lic. Ing. Dionicio Henry Pacheco Rios  
**Tutor Revisor** : M.Sc. Lic. Ing. Ramiro Kantuta Limachi  
**Tutor Especialista** : Lic. Maria Magdalena Aguilar Guanto

EL ALTO – BOLIVIA

2023



# DECLARACIÓN JURADA DE AUTENTICIDAD Y RESPONSABILIDAD

Yo **JHENNY CELIA CUSICANQUI MALDONADO** estudiante con C.I. **8365192LP** mediante la presente **declaro** de manera pública que la propuesta del **PROYECTO DE GRADO** titulada “**SISTEMA WEB PARA ADMINISTRACIÓN EN SERVICIOS DE TRADUCCIÓN CASO CONSULTORIO LINGÜÍSTICO DE LA CARRERA DE LINGÜÍSTICA E IDIOMAS**” es original, siendo resultado de mi trabajo personal y no constituye una copia o replica de trabajos similares elaborados.

Autorizo la publicación del resumen de mi propuesta en internet y me comprometo a responder a todos los cuestionamientos que se desprenden de su lectura.

Asimismo, me hago responsable ante la universidad o terceros, de cualquiera irregularidad o daño que pudiera ocasionar, por el incumplimiento de lo declarado.

De identificarse falsificación, plagio, fraude, o que el **PROYECTO DE GRADO** haya sido publicado anteriormente; asumo las consecuencias y sanciones que de mi acción se deriven, responsabilizándome por todas las cargas legales que se deriven de ello sometiéndome a las normas establecidas y vigentes de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

El Alto, Noviembre del 2023.



Jheny Celia Cusicanqui  
Maldonado

C.I. 8365192 LP  
**yujjhen@gmail.com**

## **DEDICATORIA**

Este trabajo de grado dedico a mi madre,  
por el legado que me inculcó y a mi hija,  
por ser la luz de mí existir.

## **AGRADECIMIENTOS**

Siento gratitud hacia:

Dios, por darme esta gran oportunidad para trascender en mi vida profesional y por haberme rodeado de personas tan valiosas y generosas que han ayudado en mi formación.

Todos mis docentes de la Carrera Ingeniería de Sistemas, ya que cada uno de ellos contribuyó en el crecimiento profesional y personal.

A mi esposo, Fredy Choque Ticona, por el apoyo incondicional.

A todos ellos, GRACIAS.

## ÍNDICE

MARCO PRELIMINAR.....	1
1.1. INTRODUCCIÓN .....	1
1.2. ANTECEDENTES INSTITUCIONALES .....	2
1.2.1. Principio de dependencia .....	2
1.2.1 Principio de responsabilidad .....	2
1.2.3 Principio de legitimidad y exclusividad .....	2
1.2.4 Principio de servicio a la sociedad .....	2
1.2.5 Principio de coordinación .....	3
1.2.6 Marco jurídico de la institución .....	3
1.2.7 Objeto y ámbito de aplicación .....	3
1.3. ANTECEDENTES INTERNACIONALES.....	3
1.4. ANTECEDENTES NACIONALES .....	5
1.4 PLANTEAMIENTO DEL PROBLEMA .....	6
1.5.1 Problema Principal .....	6
1.5.1. Problemas Secundarios .....	6
1.6. OBJETIVOS.....	7
1.6.1. Objetivo General .....	7
1.6.2. Objetivos Específicos .....	7
1.7. JUSTIFICACIÓN.....	8

1.7.1. Justificación Técnica .....	8
1.7.2. Justificación Económica .....	9
1.7.3. Justificación Social.....	9
1.8. METODOLOGÍA .....	10
1.8.1. Método de investigación .....	10
1.8.2. Técnicas de investigación .....	10
1.9. Método de costos COCOMO .....	11
1.9.1 Métricas de calidad .....	11
1.9.2 Pruebas de software .....	12
1.9.3 Pruebas de seguridad .....	12
1.10. HERRAMIENTAS .....	12
1.10.1. PHP .....	12
1.10.2. Gestor de Bases de Datos MySql .....	12
1.10.3. Laravel .....	13
1.10.4. Bootstrap .....	13
1.11. LÍMITES Y ALCANCES .....	13
1.11.1. Límites .....	13
1.11.2. Alcances .....	14
1.12. APORTES.....	14
MARCO TEÓRICO .....	15

2.1. Fundamento conceptual.....	15
2.1.1. Sistema web .....	15
2.1.2. Administración de servicios.....	15
2.1.3. Traducción .....	16
2.1.4. Interpretación .....	16
2.2. Fundamento teórico .....	17
2.2.1. Ingeniería de software.....	17
2.2.2. Metodologías de desarrollo de software.....	18
2.2.2.1 Metodologías tradicionales.....	18
2.2.2.2 Metodologías ágiles .....	21
2.2.2.2.1 Metodología ágil SCRUM.....	23
2.2.3. Sistema de información.....	28
2.2.4. Lenguaje Unificado de Modelado (UML) .....	28
2.2.4.1. Elementos en UML.....	29
2.2.4.1.1. Elementos estructurales.....	29
2.2.4.1.2. Elementos comportamiento.....	33
2.2.4.2. Relaciones en UML.....	34
2.2.4.3. Diagramas en UML .....	35
2.2.4.2.1. Diagramas de clase .....	36
2.2.4.2.2. Diagramas de actividades .....	37

2.2.4.2.3. Diagramas de secuencias .....	38
2.2.4.2.4. Diagramas de casos de uso .....	38
2.2.4.2.5. Diagramas de componentes.....	39
2.2.4.2.6. Diagramas de estado .....	40
2.2.4.2.7. Diagramas de despliegue.....	40
2.2.4.2.8. Diagramas de objetos .....	41
2.2.4.2.9. Diagramas de estructura compuesta .....	42
2.2.4.2.10. Diagramas de paquetes .....	43
2.2.4.2.11. Diagramas de comunicación .....	44
2.2.4.2.12. Diagramas de tiempos .....	44
2.2.5. Lenguaje de modelado UWE .....	45
2.2.5.1. Características de UWE .....	45
2.2.5.2. Fases de la metodología UWE.....	46
2.2.5.2.1. Análisis y especificación de requisitos.....	46
2.2.5.2.2. Diseño del sistema.....	47
2.2.5.2.3. Codificación del software .....	47
2.2.5.2.4. Fase de pruebas .....	47
2.2.5.2.5. Fase de implementación .....	48
2.2.5.2.6. Mantenimiento.....	49
2.2.6. Arquitectura software .....	49

2.2.6.1 Ciclo de desarrollo de la arquitectura .....	49
2.2.6.2. Beneficios de arquitectura.....	51
2.2.6.3. Rol del arquitecto .....	51
2.2.7. Calidad de software .....	51
2.2.7.1. Calidad del producto software .....	52
2.2.7.1.1. Calidad interna y externa .....	52
2.2.7.1.2. Calidad de uso .....	53
2.2.7.2. Calidad en la industria del software (ISO-9126) .....	54
2.2.7.2.1. Funcionabilidad .....	54
2.2.7.2.2. Confiabilidad .....	55
2.2.7.2.3. Usabilidad .....	55
2.2.7.2.4. Eficiencia.....	56
2.2.7.2.5. Mantenibilidad .....	56
2.2.7.2.6. Portabilidad .....	57
2.2.8. Modelo de costos de software.....	57
2.2.8.1. COCOMO II .....	57
MARCO APLICATIVO .....	59
3.1. Diagnóstico del sistema .....	59
3.2. Análisis del sistema .....	62
3.2.1. Caso de uso general .....	62

3.2.2. Caso de uso-administrador .....	63
3.2.3. Caso de uso_ Coordinador .....	65
3.2.4. Caso de uso_ iniciar sesión .....	66
3.2.4. Caso de uso_ solicitar traducción.....	67
3.3. Diagrama de secuencia .....	68
3.3.1. Ingresar al Sistema (Usuario).....	68
3.3.2. Registrarse en el Sistema (Usuario).....	69
3.3.3. Ingreso del Sistema (Administrador) .....	70
3.3.4. Registrarse en el Sistema (Usuario).....	71
3.3.5. Mostrar perfil .....	72
3.3.6. Cambiar Perfil y Clave de Usuario .....	72
3.3.7. Administrar usuario operador .....	73
3.3.8. Ingresar al Sistema .....	74
3.3.8. Ingresar al Sistema .....	74
3.3. Diagrama de clases .....	76
3.4. Modelo de navegación .....	78
3.5. Modelo de presentación .....	79
3.5.1. Modelo de presentación (Inicio) .....	79
3.5.2. Modelo de presentación (Administración de usuarios) .....	80
3.5.3. Modelo de presentación (Traducciones) .....	80

3.5.4. Modelo de presentación (Interpretación) .....	81
3.5.5. Modelo de presentación (Pagos).....	81
3.6. Codificación de software .....	82
3.6. Implementación del sistema.....	86
3.6.1. Iniciar sesión .....	88
3.6.2. Editar perfil _ administrador .....	88
3.6.4. Administración de usuarios .....	90
3.6.5. Traducciones .....	90
3.6.6. Interpretaciones .....	91
3.6.7. Pagos.....	91
3.7. Pruebas de software .....	92
3.7.1. Pruebas de Caja Blanca .....	92
3.7.1.1. Pruebas de Caja Blanca _Inicio sesión .....	93
3.7.2. Pruebas de Caja Negra.....	94
3.7.2.1. Prueba de caja negra - Inicio de Sesión.....	94
3.7.2. Pruebas de estrés .....	96
CALIDAD DE SOFTWARE .....	99
4.1. Funcionalidad.....	99
4.2. Confiabilidad .....	102
4.3. Mantenibilidad.....	103

4.4. Usabilidad .....	103
4. 5. Portabilidad .....	105
4. 7. Seguridad .....	105
4. 6. Estimación de costos del software- COCOMO II.....	106
CONCLUSIONES Y RECOMENDACIONES .....	111
4.1. Conclusiones .....	111
4.2. Recomendaciones .....	112
BIBLIOGRAFÍA.....	113

## ÍNDICE DE FIGURAS

Figura 2.1. ....	19
Figura 2.2. ....	20
Figura. 2.3 ....	20
Figura. 2.4 ....	21
Figura 2.5. ....	25
Figura 2.6 ....	30
Figura 2.7 ....	31
Figura 2.8 ....	31
Figura 2.9 ....	31
Figura 2.10 ....	32
Figura 2.11 ....	32
Figura 2.12 ....	33
Figura 2.13 ....	34
Figura 2.14 ....	34
Figura 2.15 ....	35
Figura 2.16 ....	35
Figura 2.17 ....	36
Figura 2.18 ....	37
Figura 2.19 ....	38

Figura 2.20 .....	39
Figura 2.21 .....	39
Figura 2.24 .....	41
Figura 2.25 .....	42
Figura 2.26 .....	43
Figura 2.27 .....	43
Figura 2.28 .....	44
Figura 2.29 .....	50
Figura 2.30 .....	53
Figura 2.31 .....	54
Figura 2.32 .....	58
Figura 3.1. ....	63
Figura 3.2. ....	64
Figura 3.3. ....	65
Figura 3.4. ....	66
Figura 3.5. ....	67
Figura 3.6 .....	69
Figura. 3.7.....	70
Figura. 3.9.....	71
Figura. 3.10.....	72

Figura. 3.11.....	72
Figura. 3.12.....	73
Figura. 3.13.....	74
Figura. 3.14.....	74
Figura. 3.15.....	75
Figura. 3.16.....	76
Figura. 3.17.....	77
Figura. 3.18.....	78
Figura. 3.19.....	79
Figura. 3.20.....	80
Figura. 3.21.....	80
Figura. 3.22.....	81
Figura. 3.23.....	81
Figura. 3.24.....	82
Figura. 3.25.....	86
Figura. 3.26.....	86
Figura. 3.27.....	87
Figura. 3.28.....	87
Figura. 3.29.....	87
Figura. 3.30.....	88

Figura. 3.31.....	88
Figura. 3.32.....	89
Figura. 3.33.....	89
Figura. 3.34.....	90
Figura. 3.35.....	91
Figura. 3.36.....	91
Figura. 3.37.....	92
Figura. 3.38.....	93
Figura. 3.39.....	95
Figura. 3.40.....	97

## ÍNDICE DE TABLAS

Tabla 3.1.....	59
Tabla 3.2.....	61
Tabla 3.3.....	62
Tabla 3.4.....	64
Tabla 3.5.....	65
Tabla 3.6.....	66
Tabla 3.7.....	67
Tabla 3.8.....	94
Tabla 3.9.....	95
Tabla 3.10.....	95
Tabla 3.11.....	98
Tabla 4.1.....	99
Tabla 4.2.....	100
Tabla 4.3.....	100
Tabla 4.4.....	104
Tabla 4.5.....	104
Tabla 4.6.....	107
Tabla 4.7.....	108

## RESUMEN

En base a investigaciones científicas, se ha identificado la necesidad de cualidades clave en los sistemas web, como un diseño efectivo, fiabilidad, seguridad y atención al cliente. Un diseño web adecuado influye positivamente en la satisfacción del usuario, mientras que la confiabilidad y seguridad de los datos en línea son cruciales para una experiencia satisfactoria. En la actualidad, numerosas instituciones académicas, gubernamentales y organismos internacionales han adoptado sitios web para proporcionar información esencial y servicios ininterrumpidos.

Se presenta un proyecto de investigación destinado a mejorar la gestión de un consultorio lingüístico a través del Sistema Web de Servicios de Traducción. Este sistema requiere una administración competente del servicio de traducción, generación de usuarios y la obtención de datos de usuarios para enriquecer los servicios de traducción disponibles para la población local y nacional. El proyecto busca sistematizar registros de usuarios y traductores, implementando un modelo de atención en línea para agilizar la generación de informes y eliminar la burocracia en la entrega de documentos de traducción. El desarrollo del sistema web seguirá el modelo UWE, utilizando tecnologías como PHP, MariaDB y Laravel 9.2 para construir una plataforma que responda a las necesidades de un Centro de Traducción moderno y eficiente.

Las palabras clave son: sistema web, diseño de web, servicio al cliente, lingüística, traducción, institución académica, UWE, Model, PHP, MariaDB, Laravel 9.2.

## ABSTRAC

Based on scientific research, the need for key qualities in web systems has been identified, such as effective design, reliability, security, and customer support. Proper web design positively influences user satisfaction, while the reliability and security of online data are crucial for a satisfactory experience. Today, numerous academic, governmental, and international organizations have adopted websites to provide essential information and uninterrupted services.

A research project aimed at improving the management of a language practice through the Web System of Translation Services is presented. This system requires competent administration of the translation service, user generation and the collection of user data to enrich the translation services available to the local and national population. The project seeks to systematize user and translator records, implementing an online service model to streamline the generation of reports and eliminate bureaucracy in the delivery of translation documents. The development of the web system will follow the UWE model, using technologies such as PHP, MariaDB and Laravel 9.2 to build a platform that responds to the needs of a modern and efficient Translation Center.

**Keywords:** web system, web design, customer service, linguistics, translation, academic institution, UWE, Model, PHP, MariaDB, Laravel 9.2

# CAPÍTULO I

## MARCO PRELIMINAR

### 1.1 INTRODUCCIÓN

Basado en estudios científicos, los sistemas web necesitan ciertas cualidades importantes: diseño de la web, fiabilidad, seguridad y servicio al cliente. El diseño de la web tiene un efecto positivo y significativo en la aceptación y satisfacción del cliente. A su vez, la fiabilidad y la seguridad de los datos en el sitio web generan un efecto satisfactorio. Así en los últimos tiempos, muchas instituciones académicas, gubernamentales, y organismos internacionales han ido implantado un sitio web para proporcionar información básica, guardar información especializada y prestar servicio a los usuarios las 24 horas.

El sistema de consultorio lingüístico requerirá de una administración competente en el servicio de traducción, generar informes, conocer datos de los usuarios. Esto construirá en información muy útil para mejorar el servicio de traducción a la población Alteña y Boliviana. Desafortunadamente, los usuarios que acuden a la Carrera de lingüística por servicios de traducción no disponen de un sitio web que esté a disposición las 24 horas. Por otra parte, los traductores o profesionales en lingüística no disponen de una herramienta digital que les ofrezca servicios de traducción a toda la población.

Este proyecto de investigación tiene como objetivo la sistematización de todos los registros de los usuarios y traductores. Para lograr este objetivo, se llevará a cabo un estudio estadístico para recopilar y analizaron datos sobre el posible sistema utilizando encuestas. El concepto de atención online es un tema tendente que evitará demoras en la generación de informes para la institución, el usuario y los traductores. También se evitará la burocracia en el despacho de los documentos de traducción.

El método utilizado para desarrollar el sistema web será el Modelo UWE, flujo de datos, diagrama lógico y entidad relación; el lenguaje a utilizar será PHP, MySql y Laravel 9.2.

## 1.2. ANTECEDENTES INSTITUCIONALES

La Carrera de Lingüística presta servicios enseñanza de idiomas a la toda la Universidad Pública de El Alto y la población, a través del Departamento de Idiomas. Sin embargo, no cuenta oficialmente con un centro de traducción, aunque en su organigrama tiene planificado un Centro de traducciones. En las últimas gestiones, la Decanatura del área Ciencias Sociales de la Universidad Pública de El Alto (**UPEA**) ha gestionado un proyecto de construcción de Consultorios Comunitarios, donde se pretende implementar el Consultorio Lingüístico. Para ello, en el año 2020 se gestionó la licitación del equipamiento para el Centro de Traducción de la Carrera de lingüística.

### **1. 2.1. Principio de dependencia**

El Departamento de Traducción e Interpretación tiene dependencia administrativa y ejecutiva de la carrera de lingüística e idiomas de la Universidad Pública de El Alto.

### **1.2.1 Principio de responsabilidad**

El Departamento de Traducción e Interpretación de La Carrera de Lingüística e Idiomas, tiene la responsabilidad de dar servicio de traducción e interpretación a la sociedad alteña.

### **1.2.3 Principio de legitimidad y exclusividad**

El Departamento de Traducción e Interpretación de La Carrera de Lingüística e Idiomas, es la única institución exclusiva que se atribuye el servicio de Traducción e Interpretación en la Universidad Pública de El Alto.

### **1.2.4 Principio de servicio a la sociedad**

El Departamento de Traducción e Interpretación de la Carrera de Lingüística e Idiomas está al servicio de la comunidad, dado que brinda atención de Traducción e Interpretación a estudiantes, docentes y a la comunidad alteña y público en general de escasos recursos, además cumple su rol social al interior de la universidad y con la población en general.

### **1.2.5 Principio de coordinación**

El Departamento de Traducción e Interpretación de la Carrera de Lingüística e Idiomas, generará la aplicación práctica de la traducción e interpretación en coordinación con diferentes entidades para su ejecución.

### **1.2.6 Marco jurídico de la institución**

- Constitución Política del Estado.
- Políticas lingüísticas.
- Estatuto orgánico de la Universidad Pública de El Alto.

### **1.2.7 Objeto y ámbito de aplicación**

El servicio del departamento de Traducción e Interpretación de la Carrera de Lingüística e Idiomas brinda ayuda de traducción e interpretación de documentos que requieran los estudiantes, docentes administrativos y público en general de escasos recursos de la ciudad de El Alto, cumpliendo de esta manera el rol social al interior de la universidad, así mismo con la sociedad.

## **1.3. ANTECEDENTES INTERNACIONALES**

Silva (2018) presentó un proyecto denominado “Online Appointment System For Kotelawala Defence University Hospital” en la Universidad de Colombo School of Computing. El propósito del sistema es facilitar el agendado de las citas médicas en los horarios de los doctores. Además, el paciente puede ver las especialidades de los doctores, el horario de atención y volver a ver su cita. Tanto doctores y pacientes se podrán loguearse en el sistema. Las herramientas usadas en el sistema son PHP, HTML, MySQL, CSS y XAMP (Silva, 2018).

Zambrano (2020) realizó un proyecto “Implementación de un Sistema Web para mejorar el proceso de Reserva de Citas en el Policlínico Dr. Nixon. Bagua Grande, Utcubamba, Amazonas”, en la Universidad Politécnica Amazonica. El objetivo principal del sistema es mejorar significativamente el proceso de Reserva de Citas en el

Policlínico Dr. Nixon, Bagua Grande, Utcubamba, Amazonas. El software permite mejorar de manera eficiente y eficaz el manejo del proceso de reserva de citas en el establecimiento de salud. La metodología utilizada fue SCRUM, con 2 Sprints y cada Sprint con su historia de usuarios respectivas. Las herramientas utilizadas son HTML5, JQuery y Bootstrap (CSS); también se utilizó PHP (Framework CodeIgniter que utiliza arquitectura de Model View Controller - MVC), además como gestor de base de datos se usó Mysql administrado por Phpmysql ( Zambrano, 2020).

Rashid y otros (2021) desarrollaron un sistema denominado “PatDoc – Online Appointment Management System” en la Universidad de Jain. El sistema tiene el objetivo de conectar al paciente con el especialista de manera oportuna en un tiempo récord. Los pacientes podrán agendar su cita en cualquier horario, sin restricción alguna. La finalidad del sistema es principalmente incrementar las demandas de las gestiones realizando un testeado constante. La metodología utilizada es el Extreme Programming methodology (XP) que incluye cinco cualidades importantes: comunicación, simplicidad, feedback, respeto y coraje. Las herramientas empleadas en el proyecto son HTML, CSS, JavaScript, Bootstrap (para front end); PHP, MySQL (para back end) (Rashid et.al, 2021).

Valenzuela (2021) ejecutó el proyecto denominado “Implementación de un Sistema Web para la Gestión de Consultas Médicas en una Entidad de Salud Privada en la ciudad de Lima - 2021” en la Universidad Tecnológica del Perú. El objetivo general fue implementar un Sistema Web de Gestión de Consultas Médicas para una Entidad de Salud Privada en la ciudad de Lima - 2021. La metodología utilizada es AUP que consta de cuatro fases: inicio, elaboración, construcción y transición (Valenzuela, 2021).

Buitrón y otros (2021) desarrollaron un proyecto de aplicación Web Para La Gestión De Citas E Historia Clínica De Pacientes, Utilizando Metodologías De Desarrollo Ágil. Caso De Estudio “Consultorio Médico Medicina Integral” Del Dr. Cando Herrera Johnny Stalin en la universidad de las Fuerzas Armadas. El objetivo principal del proyecto es desarrollar una aplicación web para la gestión de citas médicas e historia clínica,

utilizando metodología de desarrollo ágil Kanban junto con el Framework Laravel para el “Consultorio Médico Medicina Integral” del Dr. Cando Herrera Johnny Stalin. Las herramientas usadas son lenguaje PHP, framework Laravel. La metodología empleada fue UML (Buitrón & y Otros , 2021).

#### 1.4. ANTECEDENTES NACIONALES

Coaquira (2017) elaboró un Sistema web de reserva de hoteles caso: agencia de viajes “Definitely Bolivia” en la Universidad Mayor de San Andrés. El proyecto tiene como objetivo principal diseñar e implementar el Sistema de Reserva de Hoteles para la Agencia de Viajes Definitivamente Bolivia, que permita a la agencia y al hotelero. La metodología empleada es SCRUM y con el objetivo de modelar y cubrir todo el ciclo de vida del desarrollo de la aplicación web se recurre al UWE. Para ello, se utilizan las siguientes herramientas: PHP Versión 5.6, Servidor Web Apache, Codeigniter (framework) y MySql Versión 5.7 (gestor de base de datos) (Coaquira, sistema web de reserva de hoteles caso: Agencia de Viajes Definitely Bolivia, 2017).

Chuquimia (2021) desarrolló el proyecto “Sistema web de consulta médica virtual para el servicio de ginecología del Hospital de la Mujer” en la Universidad Mayor de San Andrés. La finalidad del proyecto fue optimizar la administración de citas médicas, y de esta forma acelerar los procesos. Para ello el objetivo general fue desarrollar un sistema web para el hospital de la mujer que permita realizar una consulta médica virtual. Para el diseño e implementación del proyecto se utilizó la metodología SCRUM combinándolo con la UWE. Para el desarrollo del Software, se empleó el lenguaje de programación PHP y un gestor de base de datos MySQL (Chuquimia, 2021).

Gonzales (2020) realizó el proyecto Sistema web de administración de historias clínicas caso: (Hospital de la Mujer) en la Universidad Pública de El Alto. El objetivo principal de la investigación fue desarrollar un sistema web de administración de Historias Clínicas que permitan mejorar las tareas de registro, búsqueda y elaboración de reportes de manera rápida y eficiente, facilitando la entrega oportuna y precisa del registro de datos. Las herramientas utilizadas son María DB (gestor de base de datos),

el lenguaje de programación PHP (con tecnologías Ajax, java script, servidor XAMPP) (Gonzales C. , 2020).

## **1.4 PLANTEAMIENTO DEL PROBLEMA**

### **1.5.1 Problema Principal**

La problemática central que aborda el proyecto "Sistema Web para Administración en Servicios de Traducción" reside en la falta de facilidades para que los clientes soliciten servicios de traducción e interpretación a través de internet. A pesar de la importancia de la traducción en diversos ámbitos sociales y profesionales, el Centro de Traducciones carece de un sistema web eficiente que permita a los usuarios realizar solicitudes de servicios de manera online. Esta carencia no solo limita la accesibilidad a los servicios de traducción e interpretación, sino que también representa una barrera para la eficiencia y agilidad en la gestión de solicitudes, afectando así la capacidad del Centro para satisfacer las necesidades de la sociedad en términos de servicios lingüísticos.

#### **1.5.1. Problemas Secundarios**

- La falta de un sistema web dificulta la comunicación fluida entre el Centro de Traducciones y los usuarios, generando incertidumbre y posibles malentendidos en cuanto al estado y progreso de las traducciones solicitadas. Esto no solo afecta la satisfacción del cliente, sino que también incide en la capacidad del Centro para brindar un servicio transparente y de calidad.
- Junto con la carencia de un sitio web especializado, se identifica que la Carrera de Lingüística e Idiomas carece de un canal digital para la promoción y difusión de sus servicios de traducción. La falta de presencia online limita la visibilidad de la Carrera y la disponibilidad de información detallada sobre los servicios ofrecidos, lo que podría resultar en una pérdida de oportunidades para atraer nuevos clientes. Esta limitación no solo afecta la eficacia en la captación de usuarios, sino que también reduce la posibilidad de establecer una relación

continúa con la comunidad, debilitando así el impacto social y educativo de la institución.

- Los traductores enfrentan el desafío de administrar manualmente documentos y archivos, lo que conlleva a demoras en el proceso de traducción. Esta práctica no solo resulta en una gestión menos eficiente de los recursos, sino que también puede generar errores humanos y afectar la calidad final de las traducciones.
- Al acudir a la Carrera de Lingüística, los usuarios se enfrentan a la carencia de información oportuna sobre los servicios de traducción. La ausencia de material informativo actualizado dificulta la comprensión de los procedimientos, tarifas y tiempos de respuesta, generando incertidumbre y afectando la experiencia del usuario.

## **1.6. OBJETIVOS**

### **1.6.1. Objetivo General**

- Desarrollar un sistema web para la administración en los servicios de traducción en el Centro de Traducción de la Carrera Lingüística e Idiomas, de la Universidad Pública de El Alto, el cual contribuirá a la eficiencia de los servicios de traducción.

### **1.6.2. Objetivos Específicos**

- Analizar los requerimientos del sistema web y desarrollar la arquitectura del sistema web para la administración en los servicios de traducción en el Centro de Traducción.
- Desarrollar los módulos esenciales del sistema web conforme a los requerimientos específicos del Centro de Traducción a través de una metodología UWE.
- Ejecutar pruebas exhaustivas para validar la funcionalidad, seguridad y eficiencia del sistema web desarrollado.

## **1.7. JUSTIFICACIÓN**

### **1.7.1. Justificación Técnica**

El Sistema web para la atención de servicios de traducción es desarrollado para solucionar los problemas existentes en el servicio de traducción, sobre todo en los idiomas nativos. La web permite la centralización de toda la información relacionada con los servicios de traducción. Esto incluye la gestión de proyectos, la asignación de traductores, el seguimiento de los trabajos en curso, y el almacenamiento seguro de documentos y archivos relacionados con las traducciones.

La automatización de los procesos administrativos del Centro de Traducción agiliza la gestión de proyectos de traducción, a través de la asignación automática de tareas, el seguimiento del progreso y la generación de informes automatizados para contribuir a la eficiencia operativa. El propósito del este sistema web es automatizar el tradicional servicio de traducción, en la cual el usuario que desea un servicio de traducción acude a las instalaciones de la Carrera de Lingüística y busca a un traductor. Así, reduce al máximo la burocracia en los servicios de traducción.

Además, la plataforma permite una gestión más efectiva de los recursos humanos y materiales necesarios para llevar a cabo los servicios de traducción e interpretación. Esto incluye la asignación adecuada de traductores según su especialidad y disponibilidad, así como la gestión eficiente de los documentos y archivos relacionados. El sistema web está diseñado para coadyuvar en la planificación estratégica y garantizar la gestión organizada de archivos de traducción para los administradores.

La implementación de un sistema web no solo optimizará los procesos internos, sino que también contribuye la mejora de la calidad de los servicios de traducción ofrecidos. La supervisión y el seguimiento más efectivos permiten corregir posibles problemas de manera oportuna, garantizando así la satisfacción de los usuarios.

### **1.7.2. Justificación Económica**

Este sistema será compatible con cualquier del dispositivo del usuario, como PC, computadora portátil, tableta y teléfono inteligente. Para que el usuario pueda acceder fácilmente al sistema en cualquier momento y en cualquier lugar. Entonces económicamente evitará costos de transporte e impresión de la documentación del usuario y del traductor. Asimismo, este sistema será muy simple y fácil de usar, por lo que cualquier usuario podrá acceder.

### **1.7.3. Justificación Social**

El proyecto busca fortalecer los lazos entre la institución y la comunidad académica, así como ampliar el acceso a servicios lingüísticos en la sociedad en general. Al mejorar la eficiencia y accesibilidad de los servicios de traducción, se contribuirá a una mayor inclusividad dentro de la universidad, facilitando la comunicación entre estudiantes, profesionales y sociedad. Además, este proyecto tiene el potencial de fomentar un entorno académico más diverso y global al reducir las barreras lingüísticas, promoviendo así la participación activa de la universidad en contextos internacionales. Así, la carrera podrá cumplir el servicio a la sociedad.

Los beneficiarios directos de este proyecto incluyen el personal del Centro de Traducción, específicamente los traductores, coordinadores y personal administrativo involucrado en la prestación de servicios de traducción. Estos profesionales experimentarán mejoras significativas en la eficiencia de sus tareas diarias mediante la automatización de procesos administrativos, la asignación más efectiva de recursos y la optimización de flujos de trabajo. El sistema permite a los traductores administrar su trabajo en línea.

La comunidad académica de la Universidad constituye un grupo de beneficiarios indirectos: estudiantes, docentes y personal administrativo que requieran servicios de traducción para documentos académicos, investigaciones u otros fines se verán favorecidos por la mayor accesibilidad y eficiencia en la obtención de servicios lingüísticos. Además, la sociedad en general (extranjeros, estudiantes normalitas del

último año académico, estudiantes de universidades indígenas e historiadores) se beneficia de este proyecto porque los usuarios podrán seleccionar el idioma en que desean traducido su documento. El sistema gestionará los documentos de los usuarios de acuerdo a fecha y hora de solicitud. El usuario tendrá la capacidad de iniciar sesión y podrá disponer y/o descargar los archivos traducidos posteriormente.

## **1.8. METODOLOGÍA**

### **1.8.1. Método de investigación**

El método que a emplear será el analítico porque “este método se distinguen los elementos de un fenómeno y se precede a revisar ordenadamente cada uno de ellos por separado” (Rodríguez, 2005, p. 30). Se realizará el análisis de los requerimientos, análisis de las diferentes etapas del servicio de traducción.

### **1.8.2. Técnicas de investigación**

Con el fin de conocer los requerimientos para el sitio web se utilizará la entrevista a informantes clave (autoridades de la carrera). Esta técnica consistirá “en formular preguntas en forma verbal con el objetivo de obtener respuestas o información (...) valida o fiable” (Ñaupas et. al 2014, p. 219). La técnica permitirá obtener la información necesaria para poder definir los módulos para el servicio de traducción online. Además, servirá para definir las cualidades que contará el sistema web para servicios de traducción.

## **1.7. MÉTODO DE INGENIERÍA**

UWE (UML-Based Web Engineering) es un método interactivo que incrementa, el cual está diseñado para el desarrollo de aplicaciones web. Este método utiliza la notación estándar UML para describir formalmente las expresiones de los modelos en el análisis, diseño y documentación. Además, el método procesa de manera unificado el desarrollo de los proyectos.

UWE hace uso de los siguientes modelos de UML:

- a) Modelo de Caso de Uso.
- b) Modelo de Clases.
- c) Modelo de Navegación.
- d) Modelo de Presentación.

UWE presenta las siguientes fases:

- a) Análisis de requisitos.
- b) Diseño del sistema.
- c) Codificación del software.
- d) Pruebas.
- e) Implementación.
- f) Mantenimiento.

## **1.9. Método de costos COCOMO**

Los procedimientos de evaluación de gastos en proyectos de desarrollo de software desempeñan un papel fundamental al señalar las justificaciones presupuestarias, orientar la planificación y supervisión del proyecto, y mejorar el análisis de inversión en los proyectos de software. Estos métodos permiten establecer una relación cuantitativa entre la inversión de recursos y el lapso de desarrollo.

### **1.9.1 Métricas de calidad**

En el marco del desarrollo de nuestro proyecto de Sistema Web de Servicios de Traducción, se implementarán las métricas de calidad de software establecidas por la norma ISO/IEC 25000. Estas métricas, centradas en la calidad del producto de software, proporcionarán un marco estructurado para evaluar aspectos clave como la funcionalidad, la confiabilidad, la usabilidad, la eficiencia y la mantenibilidad de nuestro sistema. La adopción de estas métricas permitirá una evaluación integral y

estandarizada de la calidad del software, asegurando así la entrega de un producto confiable, eficiente y que cumple con los estándares internacionales de calidad.

### **1.9.2 Pruebas de software**

En el desarrollo del Sistema Web de Servicios de Traducción, se realizarán pruebas de software conforme a las pautas de la norma ISO/IEC 27000, centradas en la seguridad de la información. Estas pruebas buscan garantizar la resistencia del sistema ante posibles amenazas y vulnerabilidades, asegurando así la implementación de estándares de seguridad robustos para ofrecer un servicio de traducción en línea confiable y seguro.

### **1.9.3 Pruebas de seguridad**

En el desarrollo de proyecto, Sistema Web de Servicios de Traducción, se lleva a cabo exhaustivas pruebas de software. Estas pruebas abarcan diversos aspectos, como funcionalidad, usabilidad, rendimiento y seguridad, con el objetivo de asegurar un producto final confiable, eficiente y que cumpla con los estándares de calidad necesarios. La aplicación de pruebas de software contribuye a identificar y corregir posibles fallos, garantizando así una experiencia fluida y segura para los usuarios que utilicen nuestros servicios de traducción en línea.

## **1.10. HERRAMIENTAS**

### **1.10.1. PHP**

PHP “es un lenguaje interpretado del lado del servidor que surge dentro de la corriente denominada código abierto” (Cobo et.al, 2005, p. 23). Este lenguaje se caracteriza por su versatilidad, potencialidad, robustez y modularidad. Una de sus ventajas de este lenguaje es su carácter multiplataforma.

### **1.10.2. Gestor de Bases de Datos MySql**

MySQL es un sistema de gestión de bases de datos relacional de código abierto conocido por su confiabilidad y rendimiento. Utiliza SQL para consultar y manipular

datos, siendo ampliamente utilizado en aplicaciones web y empresariales debido a su facilidad de uso y comunidad activa de desarrolladores.

### 1.10.3. Laravel

“Laravel es el nombre de un framework creado para trabajar con PHP creado en el año 2011 por Taylor Otwell (...) para trabajar con PHP como Symfony o Zend Framework” (Laravel, 2022). Como la mayoría de la arquitectura de los framework se compone de Modelo-Vista-Controlador (MVC) que permite la división de cualquier proyecto en tres grandes partes:

- **Modelo:** Hace referencia a los datos de la aplicación y su reglamentación
- **Vista:** Es la forma que utilizamos para presentar los datos
- **Controlador:** Es la parte del programa encargada de procesar las peticiones de los usuarios y controlar el flujo de la ejecución del sistema

### 1.10.4. Bootstrap

Bootstrap es un marco front-end gratuito para un desarrollo web más rápido y sencillo que incluye plantillas de diseño basadas en HTML y CSS para tipografía, formularios, botones, tablas, navegación, entre otros, también, así como complementos de JavaScript opcionales. Bootstrap también le brinda la capacidad de crear fácilmente diseños receptivos.

## 1.11. LÍMITES Y ALCANCES

### 1.11.1. Límites

- El sistema web estará desarrollado específicamente solo para apoyar la gestión de la información administración en servicios de traducción.
- El Sistema Web no emitirá facturación electrónica, sino un comprobante de pago.
- El sistema no realiza traducción automatizada.

### **1.11.2. Alcances**

Los módulos del sistema a desarrollarse son:

- Módulo gestión de traductores e intérpretes. Se utiliza para administrar información de los profesionales
- Módulo gestión usuarios: Se utiliza para administrar información de los clientes
- Módulo gestión de servicios: Se utiliza para administrar información de los servicios en traducción
- Módulo iniciar sesión/ registrarse: Se utiliza para administrar información de inicio de sesión y registro de usuarios-traductores.
- Módulo gestión de pagos: Se utiliza para administrar información del pago por los servicios.
- Módulo reportes: Se utiliza para administrar datos históricos de los servicios realizados.
- Módulo Publicaciones: Se utiliza para administrar publicidad de eventos que realice la institución.

### **1.12. APORTES**

El desarrollo de este proyecto será un aporte significativo para la Carrera Lingüística e Idiomas, el cual beneficiará con un sistema web para los servicios de traducción en el Consultorio Lingüístico. Este brindará un sistema de calidad, acceso confiable y seguro logrando la satisfacción de los usuarios en el sitio web por sus características de escalabilidad y trazabilidad.

# **CAPÍTULO II**

## **MARCO TEÓRICO**

### **2.1. Fundamento conceptual**

#### **2.1.1. Sistema web**

De acuerdo a Maldonado “un sistema web es a aquellas aplicaciones a los usuarios pueden acceder mediante un servidor web a través del internet o de una intranet (2016, p. 12). El mismo autor nos indica que “una Página Web puede contener elementos que permiten una comunicación activa entre el usuario y la información. Esto permite que el usuario acceda a los datos de modo interactivo, gracias a que la página responderá a cada una de sus acciones” (Maldonado, 2016, p. 12). Algunos como por ejemplo rellenar y enviar formularios, participar en juegos diversos y acceder a gestores de base de datos de todo tipo.

“Se denomina sistema web a aquellas aplicaciones de software que puede utilizarse accediendo a un servidor web a través de Internet o de una intranet mediante un navegador (San Juan, 2016). Un sistema web se refiere a aplicaciones a las que los usuarios pueden acceder a través de un servidor web en internet o una intranet, lo que significa que son aplicaciones alojadas en servidores remotos en lugar de instalarse localmente en nuestras computadoras. Es esencial destacar que una página web puede incluir elementos que facilitan la interacción activa entre el usuario y la información. Esto posibilita una experiencia interactiva en la que el usuario puede acceder a datos y realizar acciones como completar formularios, participar en juegos y gestionar diversas bases de datos, ya que la página web responde a sus acciones de manera dinámica.

#### **2.1.2. Administración de servicios**

La administración de servicios se refiere a la planificación, organización, coordinación, supervisión y mejora de la prestación de servicios con el objetivo de garantizar la eficiencia, la satisfacción del cliente y la maximización del valor ofrecido por una organización. Implica la gestión de todas las actividades y recursos relacionados con la entrega de servicios, incluyendo la interacción con los clientes, la optimización de

procesos, la toma de decisiones estratégicas y la adaptación a las cambiantes demandas del mercado. La administración de servicios es esencial en una amplia variedad de sectores, como la atención médica, la tecnología, la hostelería, la educación y más, ya que contribuye a crear experiencias positivas para los usuarios, aumenta la competitividad de las empresas y fomenta la lealtad de los clientes.

“Un administrador de servicios es un estratega de negocios, orientado a asegurar que se cumpla el buen servicio como valor diferencial de la empresa. Pero hay mucho más detrás de una profesión que hoy puede hacer la diferencia en cualquier empresa” (Universidad de Piura, 2023).

### **2.1.3. Traducción**

“La traducción sería el proceso y el resultado de la transformación de mensajes mediante la interpretación de sus contenidos de sentido a partir de un código 1 y su posterior recreación en un código 2. Todo ello enmarcado en las coordenadas particulares de un acto comunicativo, un hecho lingüístico-textual del habla y una situación externa, extralingüística” (Vukovic, 2012, p 35-36).

La traducción se refiere al procedimiento y al producto de modificar mensajes mediante la comprensión de su significado original en un código 1, seguido de su posterior reestructuración en un código 2. Todo esto se realiza en el contexto específico de un acto de comunicación, un evento lingüístico-textual hablado y una situación externa, no relacionada con el lenguaje, que es singular y particular.

### **2.1.4. Interpretación**

“La interpretación es una actividad humana tan antigua como la traducción, (...) existe desde que el género humano tuvo uso de palabra puesto que siempre fueron necesarios los intermediarios entre pueblos de culturas e idiomas distintos para facilitar la comunicación a todos los niveles” (Valdivia, 2022, p. 175). Mientras que, para Valdivia la interpretación es una traducción oral instantánea por oposición a la traducción. Se puede definir como una operación sobre el discurso mediante la cual el

intérprete efectúa la transmisión del sentido del discurso de la lengua original formulándolo en la lengua terminal (2022, p 175).

La actividad de interpretación, al igual que la traducción, posee una larga historia en la humanidad y se origina desde el momento en que las personas comenzaron a utilizar el lenguaje. Siempre se han requerido intermediarios para facilitar la comunicación entre diferentes pueblos que hablan distintos idiomas y pertenecen a culturas diversas en todos los niveles. La interpretación se distingue de la traducción en que es una traducción oral instantánea. En esencia, se trata de un proceso en el que el intérprete transmite el significado del discurso original formulándolo en la lengua de destino. De esta manera, la interpretación desempeña un papel esencial en la mediación intercultural y la comunicación efectiva entre personas que hablan diferentes idiomas y provienen de diversas culturas.

## **2.2. Fundamento teórico**

### **2.2.1. Ingeniería de software**

Según Pradel y Rara (2013), un software es todo aquello intangible que hay en un ordenador (todo el conjunto de programas informáticos que indican la secuencia de instrucciones que un ordenador debe ejecutar durante su funcionamiento). Entonces, el software incluye los programas que gobiernan el funcionamiento del sistema, además contiene otros elementos como documentos, base de dato; estos se constituyen en aspectos inmateriales.

La ingeniería de software es “la aplicación de un enfoque sistemático, disciplinado y cuantificable en el desarrollo, la operación y el mantenimiento del software” (Pradel y Raya, 2013, p. 9). De tal manera, es una disciplina que implica el uso de herramientas, técnicas y estructuras para construir programas informáticos. Además, esta rama de la ingeniería se encarga del análisis previo de la situación, creación del software, pruebas necesarias y la redacción del proyecto.

## **2.2.2. Metodologías de desarrollo de software**

Las metodologías son conjuntos de procedimientos racionales que se utilizan para alcanzar un objetivo requiriendo habilidades y conocimientos específicos. Esto conlleva la elección del método, técnicas e instrumentos adecuados para cumplir con los objetivos. Por esta razón, la metodología del desarrollo de software se constituye en una etapa específica de un proyecto que parte de una posición teórica.

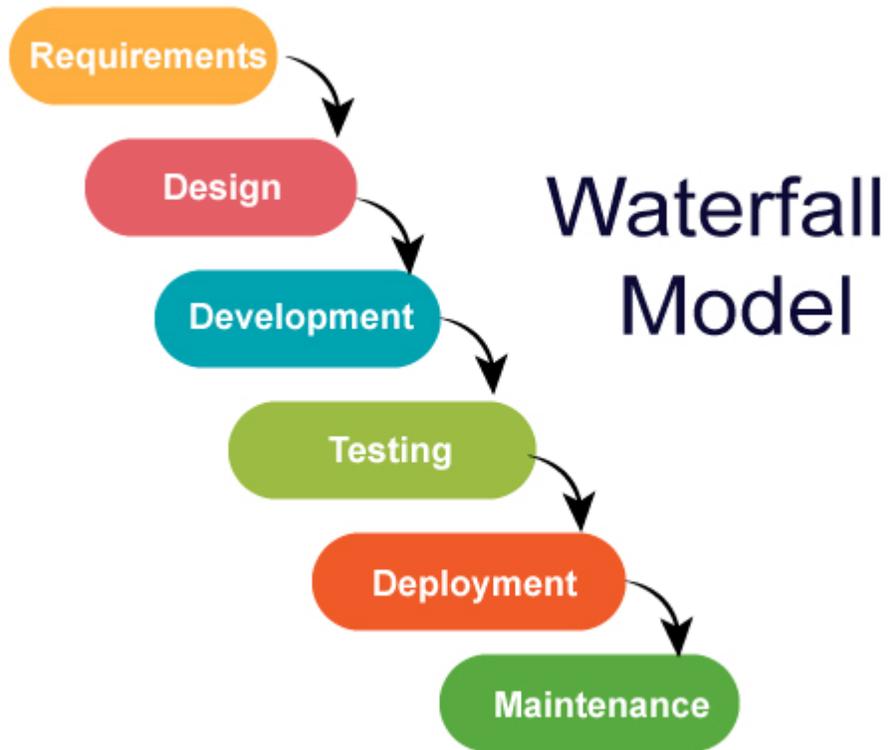
### **2.2.2.1 Metodologías tradicionales**

Estas metodologías centran su atención en la documentación, planificación y control exhaustiva de todo el proyecto. Las metodologías tradicionales imponen su disciplina rigurosa de trabajo sobre el proceso de desarrollo del software haciendo énfasis en la planificación total de proyecto, luego recién se realiza el desarrollo del software. Estas metodologías se centran especialmente en el control del proceso a través de roles, actividades, herramientas y notaciones para el modelado y documentación detallada (Maida y Pacienza, 2015).

Entre las metodologías tradiciones existen varias, pero las más conocidas son waterfall (cascada), prototipado, espiral, incremental, entre otros.

**Waterfall.** Esta metodología tiene una organización vertical similar que las cascadas de agua. Cuenta con diferentes funciones que son desarrollados en distintas etapas con una metodología de orden riguroso. Se caracteriza por dar cumplimiento a cada etapa de desarrollo para continuar con la siguiente etapa. Además, las especificaciones y requisitos establecidos inicialmente no se cambian hasta concluir con el proyecto.

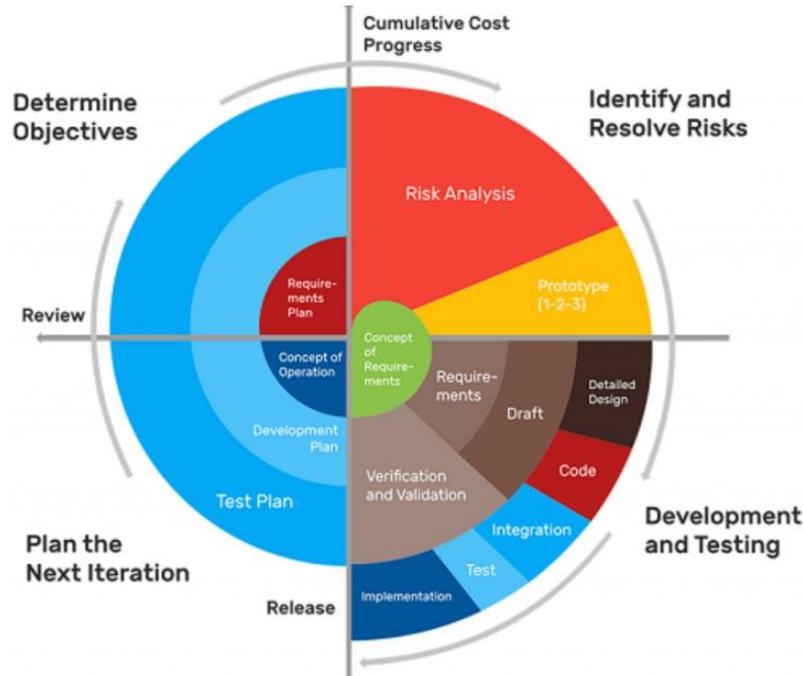
**Figura 2.1.**  
Modelo cascada



**Nota:** Pacienza, 2015

**Espiral.** Esta metodología adquiere el nombre de espiral por el tipo de su funcionamiento espiral, esto implica que el proyecto está más avanzado cuando está más cerca del centro de la espiral. Esta metodología se destaca por añadir el concepto de análisis de riesgos. Cuenta con cuatro etapas de desarrollo: planificación, análisis de riesgo, desarrollo de prototipo y evaluación del cliente.

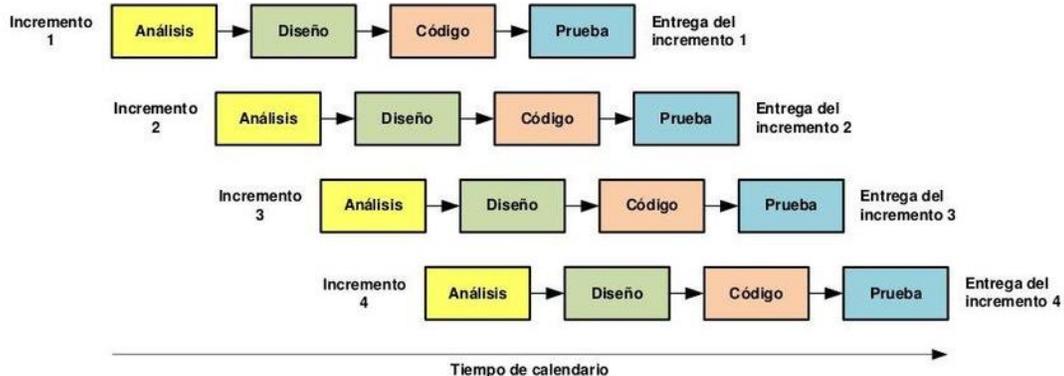
**Figura 2.2.**  
Modelo espiral



Nota: Pacienza, 2015

**Incremental.** Esta metodología desarrolla el producto final de manera progresiva y escalonada. De tal forma, en cada etapa se incrementa una nueva funcionalidad. Una ventaja de esta metodología es la aplicación de secuencias lineales de forma escalonada mientras progresa el tiempo en el calendario. Esto implica que el producto final se puede empezar a utilizar incluso antes de que se complete totalmente.

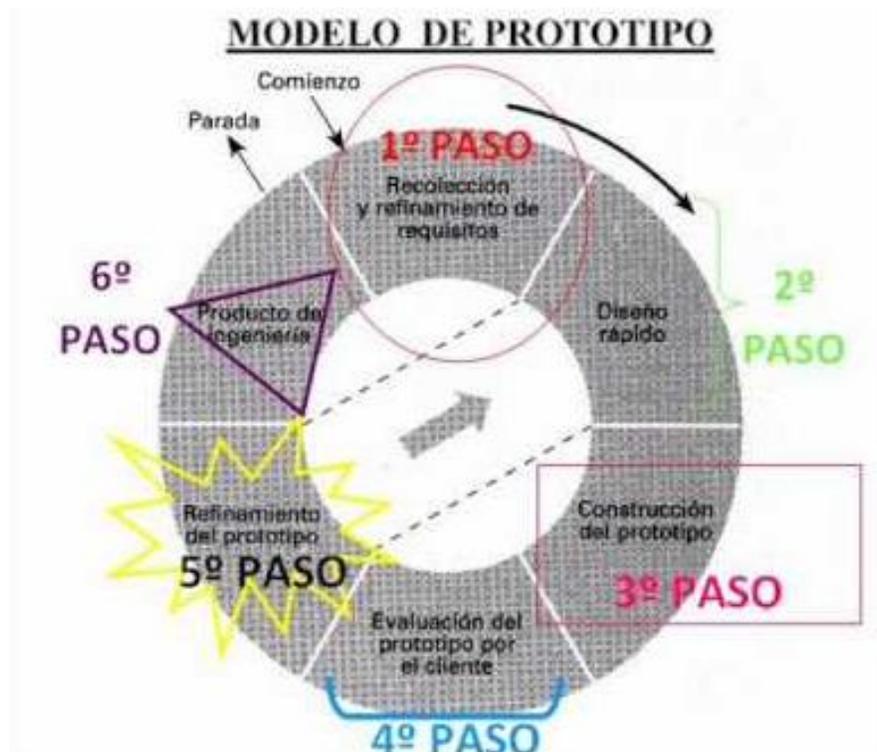
**Figura. 2.3**  
Incremental



Nota: Pacienza, 2015

**Prototipado.** Esta mitología se basa en la mejora continua. Esto implica que el desarrollador pueda corregir los errores e incluir otros requerimientos para continuar con el proyecto. Así, este método se fundamenta en la lógica de la prueba y error para lograr mejor un producto final. Se caracteriza por estar enfocado en diseñar, implementar, medir y ajustar el plan inicial. Esto ha producido que muchos usuarios puedan seguir desarrollando feedback para luego seguir aportando con ciertas ideas.

**Figura. 2.4**  
Prototipado



**Nota:** Pacienza, 2015

### **2.2.2.2 Metodologías ágiles**

El desarrollo ágil es un conjunto de métodos iterativos. Se caracterizan por ser procesos incrementales, cooperativos, flexibles y adaptativos. Otra característica de este tipo de métodos está orientada más en el software que la documentación (Maida y Pacienza, 2015, p. 52). Además, las metodologías ágiles tienen "un enfoque orientado a la participación de los usuarios y clientes (...) trabajando más en el

software y menos en la documentación, colaborando más con los clientes en vez de negociar y responder a los cambios” (p. 51).

Entre los valores principales de esta metodología están:

- Desarrollar un software que funcione
- Colaboración con el cliente
- Responder a los cambios
- Considerar al individuo y sus interacciones

Las ventajas más sobresalientes que tiene la metodología ágil son:

- **Mejora de la calidad del producto.** Para lograr esta ventaja la metodología fomenta el enfoque proactivo de los miembros del equipo para buscar la excelencia del resultado final a través de la integración, comprobación y mejora continua.
- **Mayor satisfacción del cliente.** Esto se logra a través de las demostraciones y entregas al cliente. Así el cliente se siente más satisfecho e involucrado con el proyecto y puede ver el desarrollo del producto en tiempo real.
- **Mayor motivación de los trabajadores.** Los desarrolladores se sienten con una motivación mayor por el trabajo autogestionado. Esto genera que se les facilita desarrollar capacidad creativa e innovativa.
- **Trabajo colaborativo.** Este aspecto permite que los desarrolladores realizan un trabajo más organizado, a través de roles que cumplen cada uno de ellos. Asimismo, las constantes reuniones de evaluación del proyecto ayudan a mejorar las falencias en el equipo de trabajo.
- **Uso de métricas más relevantes.** Debido a los pequeños grupos de trabajo y las fases es posible estimar parámetros como tiempo, coste, rendimiento, etc.
- **Mayor control y capacidad de predicción.** Esta ventaja se logra debido a la revisión constante y adaptaciones adecuadas que realizan los desarrolladores para lograr predecir mejor no solo los costos, sino también el tiempo.

- **Reducción de costes.** Las ventajas mencionadas anteriormente posibilitan que el proyecto tenga pocas probabilidades de fracasar. Esto implica una reducción de costos de enmienda que cliente pueda exigir.

Diferentes autores presentan una clasificación con distintas listas de los métodos ágiles. Sin embargo, los más comunes son: Karban, Scrum y programación extrema (XP). Por otro lado, tenemos a Product Owner, Desarrollo de software adaptativo (ASD), Método de desarrollo dinámico de software (DSDM), Desarrollo impulsado por las características (FDD) y Desarrollo orientado al comportamiento (BDD).

#### **2.2.2.2.1 Metodología ágil SCRUM**

Esta metodología de trabajo tiene sus principios fundamentales en la década de 1980, la cual fue desarrollada por su necesidad en proceso de reingeniería por Goldratt, Takeuchi y Nonaka (Maida y Pacienza, 2015, p. 73). El SCRUM se define como “una colección de procesos para la gestión de proyectos, que permite centrarse en la entrega de valor para el cliente y la potenciación del equipo para lograr su máxima eficiencia dentro de un esquema de mejora continua” (Mariño y Alfonso, 2014, p. 414).

A continuación, se cita algunas conclusiones resaltantes que autores dieron a cerca de la metodología SCRUM. Por su parte, Castañeda Sandoval et al. (2021 citado en Estrada & y otros, 2021) indica que:

La metodología Scrum puede ser utilizada como un instrumento ágil que nace en la industria desarrolladora de software y tecnología, la misma que ha demostrado que su uso, mejora los procesos y procedimientos. En el caso de SAUCO TECHNOLOGIES S.A.S. Utiliza la metodología SCRUM porque busca estar en la vanguardia de procesos y metodologías para realizar sus actividades de una forma eficiente que le permita obtener un alto valor agregado en el mercado con respecto a sus competidores.

Por su parte Armijos et al. (2015 citado en Estrada y otros, 2021) apoya el uso de la metodología SCRUM en proyectos afirmando los siguiente:

La selección de las metodologías ágiles, en particular la metodología Scrum que fue adaptado a las necesidades del proyecto, les permitieron desarrollar un producto con flexibilidad y sin mayor afectación en los cronogramas planificados, considerando el respectivo análisis del producto, recursos disponibles y del entorno para hacer adaptaciones de Scrum o de cualquier metodología de desarrollo.

Rodríguez Bocanegra et al., (2021 citado en Estrada y otros, 2021) destaca que la aplicación de prácticas de la metodología Scrum demostraron que:

El conocimiento y asimilación de los equipos de trabajo, en cuanto a los roles, artefactos, eventos y reglas que componen una metodología ágil basada en el marco de trabajo Scrum; permitieron tener equipos de trabajo más auto gestionados, que se evidenció en el desarrollo de proyectos más eficientes, flexibles y adaptativos.

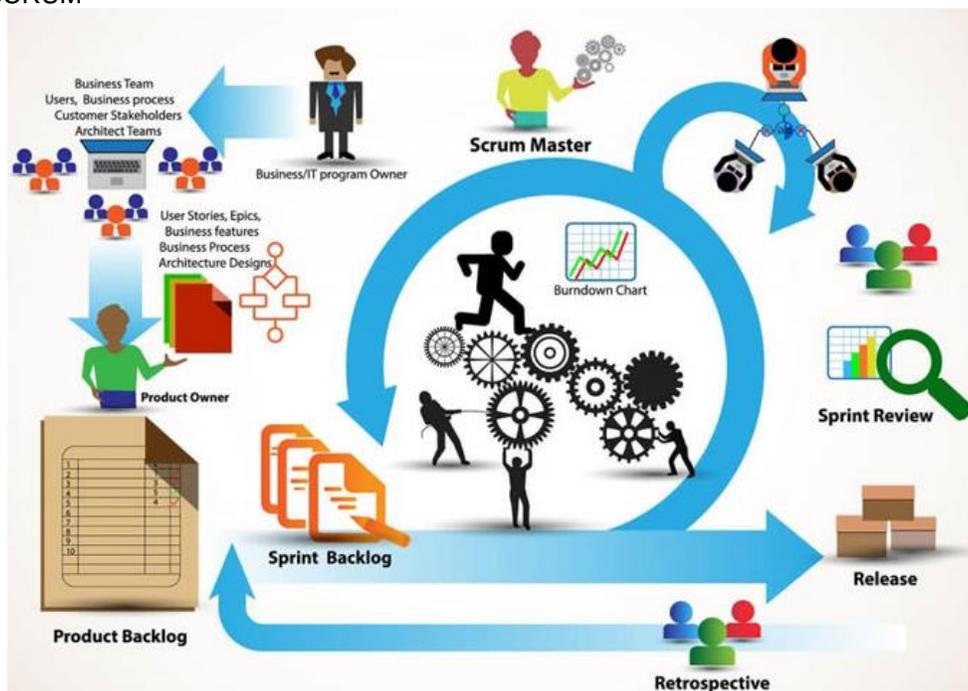
Una de las conclusiones más importantes fue la de Straccia et al., (2016 citado en Estrada y otros, 2021), quien explica que:

Una adaptación de la metodología Scrum para el desarrollo de software en proyectos de I+D en el ámbito universitario, enfocados en la importancia del desarrollo de Software basado en componentes, el papel de la arquitectura de software y la gestión de personas como pool de recursos. Además, explican que una vez puesta en marcha la metodología, se produjeron avances importantes sobre uno de los productos iniciados (CW), pues indican que se logró poner en marcha un proyecto de centralización de datos maestros (DM), generar una primera versión de nuevos productos (RR y TD), iniciar un nuevo producto (AS) y trabajar integradamente con otras instituciones. Por lo que la adaptabilidad en la implementación estuvo acompañada de un proceso de ingeniería de software mucho más completo que el estudio de Hernández et al., (2015)

Luego de revisar las más destacadas conclusiones que autores realizan, se puede considerar que el uso de la metodología SCRUM EN el desarrollo de software permite un entorno de desarrollo controlado, a través actividades vigiladas, controladas y validadas. Además, estas cuentan con características ágiles y flexibles que se ajustan a las necesidades requeridas. Por tanto, en caso de presentarse inconvenientes, estos son resueltos en equipos de trabajo consecuentes y pertinentes que solventan problemas individuales a fin de tener avances exitosos en la implementación del proyecto.

El método SCRUM es un marco de trabajo iterativo e incremental para el desarrollo de proyectos y se estructura en ciclos de trabajo denominados Sprints. Los Sprints son iteraciones de 1 a 4 semanas que suceden una detrás de otra. Al inicio de cada Sprint, el equipo multi-funcional elige los requisitos del cliente de una lista priorizada. Luego estos se comprometen a terminar los elementos al final del Sprint. Durante el Sprint no se pueden cambiar los elementos elegidos. Al final del Sprint, el equipo revisa con los interesados en el proyecto, y les enseña lo que han construido. Finalmente, el equipo obtiene comentarios y observaciones que se puede incorporar al siguiente Sprint.

**Figura 2.5.**  
método SCRUM



**Nota:** Estrada y otros, 2021

Las prácticas empleadas por SCRUM para mantener un control ágil en el proyecto son:

- Revisión de las iteraciones
- Desarrollo incremental
- Desarrollo evolutivo
- Autoorganización del equipo
- Colaboración.

La metodología Scrum tiene tres principales roles: producto owner, Scrum master y development team member.

- **Dueño del producto (Product Owner).** Este rol asegura que el proyecto se desarrolle acorde con la estrategia del negocio. También tiene la función de escribir historias de usuario, priorizarlas, y colocarlas en el Product Backlog. Debe conocer la velocidad del equipo, para realizar estimaciones de cuando estarán implementadas las necesidades en el producto. Es el responsable de cancelar el sprint si ocurre un imprevisto extremo (López, 2018, p. 57).
- **Scrum Master o Facilitador.** Su rol principal es participar en las reuniones y asegurándose del cumplimiento en el tiempo y el objetivo establecido, con el fin de eliminar los obstáculos. Además, el Scrum Master debe “disminuir al mínimo la interacción de los miembros del equipo con agentes externos, con el fin de que cada uno de los miembros se mantenga concentrado en las actividades propias del proceso de desarrollo del SIG” (Arias y Durango, 2018, p. 32).
- **Equipo de desarrollo (Development team Member).** Son los encargados de crear el producto para que pueda estar listo con los requerimientos necesarios. Así, el equipo “se encuentra conformado por los profesionales que intervienen de forma activa en el proceso de construcción de los productos SIG que se entregarán al final de la fase de desarrollo (Sprint)” (Ídem).

Dentro de la metodología SCRUM existen eventos o fases que están diseñados específicamente para habilitar las vitales transparencias e inspecciones. Entre estas están el sprint, cancelación de un sprint, reunión de planificación de un sprint,

planificación del sprint, scrum diario, revisión de sprint y retrospectiva de sprint. Todos estos eventos posibilitan tienen la finalidad de crear regularidad y minimizar la necesidad de reuniones no definidas en Scrum.

- **Sprint.** Para De Toro (2022) el sprint es el “periodo de tiempo en el que el equipo trabaja para finalizar una cantidad de tareas establecidas”. Este es el elemento central de la metodología de desarrollo (framework) y su ejecución correcta potencia la eficacia del trabajo y reduce los problemas a la hora de lanzar el mejor software. se encuentra enmarcado en un periodo de tiempo menor a cuatro (4) semanas, en el cual se realizan las actividades necesarias para elaborar una nueva versión del producto (Increment) de tipo funcional.
- **Planificación del sprint.** Consiste en una reunión de planificación de las actividades que se realizarán durante el próximo Sprint. Las reuniones son planeadas de forma activa por el equipo de desarrollo según los requisitos generales del sistema o Product Backlog. Según Arias y Durango (2018), esta reunión “se encuentra enmarcada en un periodo de tiempo con una duración máxima de ocho (8) horas, en el cual se planean actividades para un máximo de cuatro (4) semanas”. Entonces, el equipo se reúne antes de identificar y establecer las tareas y el objetivo que se llevarán a cabo en el sprint.
- **Scrum diario.** Consiste en una reunión que dura aproximadamente 15 minutos, donde se planean actividades para la siguiente 24 horas. El objetivo principal de esta reunión “es que los empleados estén informados del trabajo de los otros, cuáles son los problemas que van surgiendo o que prevén encontrar y cómo lo resolverán” (De Toro, 2022). Su principal objetivo es sincronizar las actividades del equipo de Desarrollo.
- **Revisión de sprint.** Consta de una reunión donde “se muestra los trabajos finalizados y su duración no debería ser mayor a 4 horas” (De Toro, 2022). Es una reunión donde el cliente si puede asistir porque se revisa el trabajo que se le va entregar. Además, el cliente tiene la oportunidad para dar su opinión de las nuevas tareas y validar los resultados.

- **Retroactiva de sprint.** Este evento consiste en comunicar las impresiones sobre el sprint recién superado. Esta reunión final ayuda al equipo de desarrollo proponer mejoras para el siguiente sprint.

### **2.2.3. Sistema de información**

Según De Migue y Piattini (citados en Tramullas, 1997, p. 220), el sistema de información es “un conjunto de elementos ordenados relacionados entre sí de acuerdo con unas ciertas reglas que aporta al sistema de objeto, (...) la información necesaria para el cumplimiento de sus fines, para lo cual tendrá que recoger, procesar y almacenar datos, procesos tanto de la misma organización como de fuentes externas facilitando la recuperación, elaboración presentación”.

Así, un sistema de información es un conjunto de elementos, los que interactúan entre sí con un fin común. También, permite la disponibilidad de la información para satisfacer las necesidades en una organización. No obstante, las definiciones de sistema de información pueden diferir desde la perspectiva que se enfoca. Por tanto, no será extraño obtener diferentes conceptos, con diferentes enfoques.

### **2.2.4. Lenguaje Unificado de Modelado (UML)**

UML es “un lenguaje de modelado para visualizar, especificar, construir y documentar partes de un sistema software desde distintos puntos de vista” (García y García, 2018, p. 4).

The experience of modeling practices in modern industries shows that UML is found useful by modelers. The amount of modeling projects that use UML, the amount of books written about UML and the number of software tools that support UML are large in relation with the analogous practical achievements of other modeling languages. This proves that UML in its current state is more practical than other modeling solutions, although it doesn't mean that there are no problems with current state of UML (Jézéquel et al, 2022, p. 4).

El lenguaje unificado de modernización (UML) es un estándar publicado por el OMG, que define la notación aceptada universalmente para la creación de modelos de software. Una de sus características más importantes es no indicar los modelos y artefactos a crearse, sino que ofrece una serie de diagramas que los diferentes métodos pueden utilizar para sus artefactos. La finalidad principal del UML es unificar la notación grafica que usan los diferentes métodos de desarrollo con independencia del método empleado. Por tanto, cualquier ingeniero de software tiene la capacidad de comprender los modelos creados por otros ingenieros.

#### **2.2.4.1. Elementos en UML**

UML tiene cuatro elementos que son bloques básicos de construcción orientados a objetos de este método. Estas se utilizan para escribir modelos bien estructurados (Booch et al., 2006).

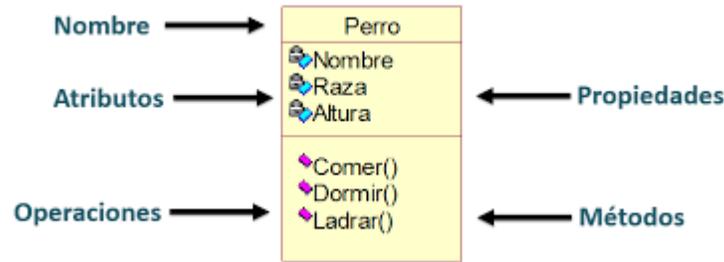
- Elementos estructurales
- Elementos de comportamiento
- Elementos de agrupación
- Elementos de anotación

##### **2.2.4.1.1. Elementos estructurales**

Estos elementos son los nombres de los modelos UML, que conforman la parte estática del modelo. También se los conoce con el nombre de clasificadores. Los elementos estructurales son siete: clase, interfaz, colaboración, casos de uso, clases activas, componentes y nodos.

**Clase.** Son un conjunto de objetos que comparten los mismos atributos, operaciones, relaciones y semántica. Comúnmente, una clase implementa una o más interfaces (Booch et al., 2006).

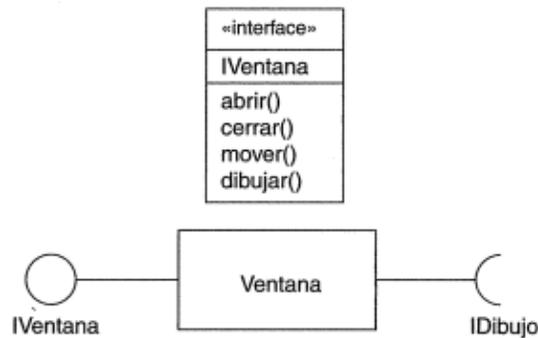
**Figura 2.6**  
Clase



**Nota:** Booch y et al., 2006

**Interfaz.** Se constituye en una colección de operaciones que tienen la finalidad de especificar el servicio de una clase o componente. Una interfaz puede representar el comportamiento total o parcial de una clase o componente. La representación de una interfaz se realiza con una pequeña semicircunferencia unida a los rectángulos de la primera clase por una línea (Booch et al., 2006).

**Figura 2.7**  
Interfaz



**Nota:** Booch Grady, 2006

**Colaboración.** La colaboración es una sociedad de roles y otros elementos que proporcionan un comportamiento cooperativo mayor a la de sus elementos. Las colaboraciones representan la implementación de patrones que configuran un sistema. Entonces, un objeto o clase puede ser parte de varias colaboraciones. Gráficamente una colaboración se representa con una elipse de borde discontinua que generalmente tiene solo su nombre.

**Figura 2.7**  
colaboración



**Nota:** Elaboración en basa a Booch Grady, 2006

**Caso de uso.** Consiste en una descripción de un conjunto secuencial de ejecutados por un sistema y que generan un resultado observable. El caso de uso se utiliza para estructurar los comportamientos de un modelo particular. Gráficamente el caso de uso se representa con una elipse continua que incluye su nombre.

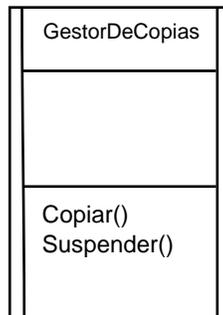
**Figura 2.8**  
Interfaz



**Nota:** elaboración en basa a Booch Grady, 2006

**Clase activa.** Es una clase donde sus objetos tienen de uno o más procesos de ejecución que pueden iniciar una actividad de control de objeto. Gráficamente, una clase activa se representa con líneas dobles a la izquierda y la derecha (Booch y et al., 2006).

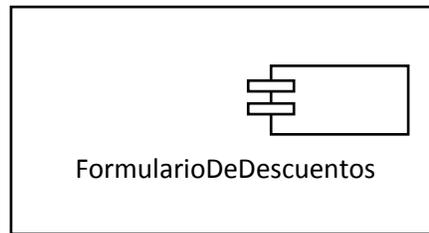
**Figura 2.9**  
Clase activa



**Nota.** Elaboración en base a Booch y et al., 2006

**Componente.** Consiste en una parte modular de un diseño del sistema que implica su implementación tras interfases externa. Los componentes de un sistema tienen la capacidad de sustituirse por compartir el mismo comportamiento lógico. Un componente, gráficamente, se representa como una clase con un icono especial en la esquina derecha.

**Figura 2.10**  
Clase activa



**Nota:** elaboración en base a Booch y et al., 2006

**Artefacto.** Contiene información física, es decir, bits que pueden ser reemplazados de un sistema. Comúnmente, un artefacto representa el empaquetamiento físico del código fuente. Dentro de un sistema existe diferentes tipos de artefactos: archivos de código fuente, ejecutable y scripts. Gráficamente, se representa en forma de un rectángulo con la palabra clave <<artefact>> (Booch et al., 2006).

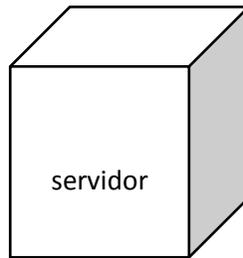
**Figura 2.11**  
Artefacto



**Nota:** elaborado en base a Booch et al., 2006

**Nodo.** Es un elemento físico que se puede percibir en la ejecución. Un nodo representa a un recurso computacional que cuenta con memoria, frecuencia y capacidad de procesamiento. El nodo, gráficamente, se representa con un cubo que incluye un nombre.

**Figura 2.12**  
Clase activa



**Nota:** elaborado en base a Booch et al., 2006

#### **2.2.4.1.2. Elementos comportamiento**

Conforman la parte dinámica del modelo UML, constituyéndose como el núcleo de los modelos que representan el comportamiento de tiempo y espacio. Existe tres tipos de elemento de comportamiento:

- Interacción
- Máquina de estados
- actividad

#### **2.2.4.1.3. Elementos agrupación**

Son las partes organizativas de los modelos UML, que se pueden descomponer en cajas. El principal elemento de la agrupación es el paquete que consta de un mecanismo de propósito general para la organización del propio diseño. A diferencia de los componentes, el paquete es netamente conceptual (Booch & et al., 2006).

#### **2.2.4.1.4. Elementos anotación**

Son las partes que se explican de un modelo de UML, más conocidos como comentarios que se pueden aplicar para hacer observaciones, clarificar o describir los elementos de un modelo. Existe un elemento resaltante dentro las anotaciones denominado nota. Esta sirve para mostrar restricciones y comentarios de un elemento (Booch y et al., 2006).

### 2.2.4.2. Relaciones en UML

Dentro el UML existe cuatro tipos de reacciones:

- Dependencia
- Asociación
- Generalización
- Realización

**Dependencia.** Es una relación semántica entre dos elementos donde los cambios pueden afectar al significado de otro elemento. La representación gráfica de la relación de dependencia se realiza con líneas discontinuas.

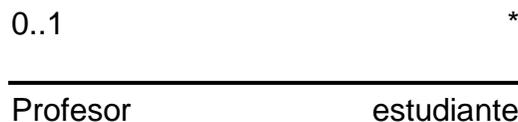
**Figura 2.13**  
Dependencia



**Nota:** elaborado Booch y et al., 2006

**Asociación.** “Es un tipo de relación estructural entre clases que describe un conjunto de clases, los cuales son conexiones entre objetos” (instancias de clases) (Booch & et al., 2006, pág. 27). La representación gráfica se realiza mediante una línea que muchos casos se le añade el nombre del rol o la multiplicidad.

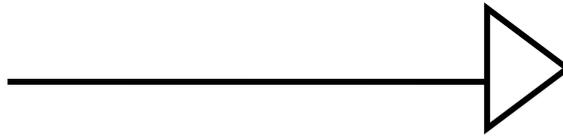
**Figura 2.14**  
Asociación



**Nota:** elaborado Booch y et al., 2006

**Generalización.** Es una relación de generalización o especialización donde un elemento especializado (el hijo) se fundamenta en la especificación del elemento generalizado (el padre). La relación hijo comparte la estructura y el comportamiento de la relación padre. Su representación gráfica se realiza mediante una flecha continua vacía orientado hacia el padre (Booch & et al., 2006).

**Figura 2.15**  
Generalización



**Nota:** elaborado Booch y et al., 2006

**Realización.** Es una relación de tipo semántico entre los clasificadores, en el cual el clasificador especifica un contrato que otro clasificador garantiza el cumplimiento. Este tipo de relaciones es muy común en los sitios de interfaz y clases-componentes. La representación gráfica se realiza con una flecha con líneas discontinuas (Booch et al., 2006).

**Figura 2.16**  
Realización



**Nota:** elaborado Booch y et al., 2006

### **2.2.4.3. Diagramas en UML**

Un diagrama es una proyección de un sistema. Generalmente, un diagrama representa una vista resumida de los elementos que constituyen un sistema. UML consta de trece diagramas:

- Diagramas de clases
- Diagrama de actividades
- Diagrama de secuencia
- Diagrama de casos de uso
- Diagrama de componentes
- Diagrama de estados
- Diagrama de despliegue

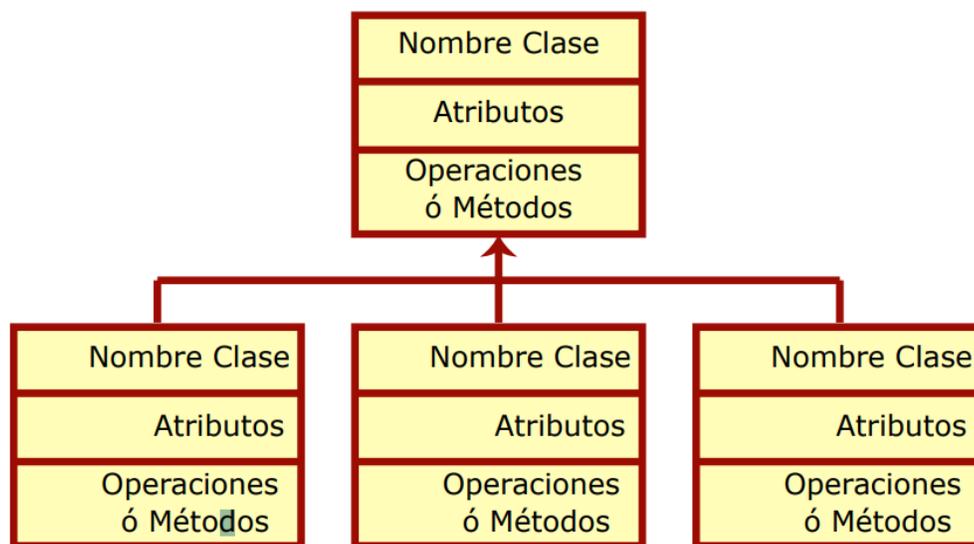
- Diagrama de objetos
- Diagrama de estructura compuesta
- Diagrama de paquetes
- Diagrama de comunicación
- Diagrama de tiempos
- Diagrama de visión global de interacciones

#### 2.2.4.2.1. Diagramas de clase

Este tipo de diagrama “muestra un conjunto de clases, interfaces y colaboraciones, así como sus colaboraciones” (Booch y otros., 2006, p. 28). Los diagramas de clase abarcan la vista de un diseño estática de un sistema.

Los *diagramas de clases* se usan para mostrar las clases de un sistema y las relaciones entre ellas. Una sola clase puede mostrarse en más de un diagrama de clases y no es necesario mostrar todas las clases en un solo diagrama monolítico de clases. El mayor valor es mostrar las clases y sus relaciones desde varias perspectivas, de una manera que ayudará a transmitir la comprensión más útil (Kimmel, 2008, p. 8).

**Figura 2.17**  
Diagramas de clase

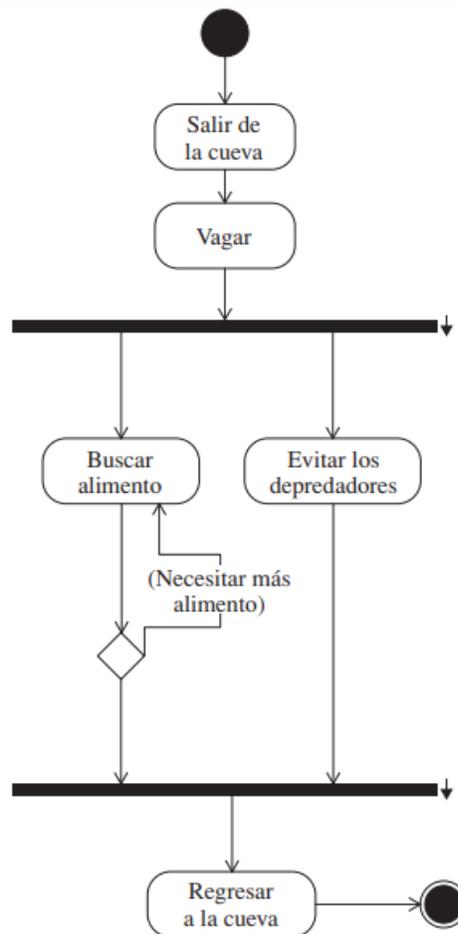


**Nota:** FAVA- Servicio Nacional de Aprendizaje (SENA)

### 2.2.4.2.2. Diagramas de actividades

“Un diagrama de actividades es una herramienta excelente para analizar problemas que el sistema deberá resolver” (Kimmel, 2008, p. 8). Estos diagramas “representan los procesos de negocios de alto nivel, incluidos el flujo de datos. También puede utilizarse para modelar lógica compleja y/o paralela dentro de un sistema” ( Formación en Ambientes Virtuales de Aprendizaje, s/n, p. 5). Además, los diagramas de actividades se pueden usar para entender el problema o incluso refinar los procesos comprendidos por el problema.

**Figura 2.18**  
Diagramas de clase

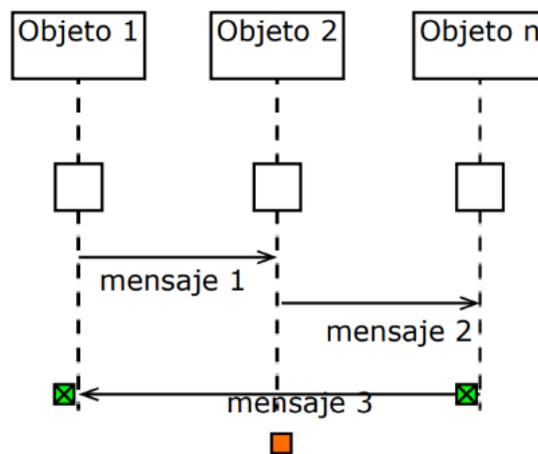


**Nota:** Kimmel, 2008, p. 8

### 2.2.4.2.3. Diagramas de secuencias

Los diagramas de secuencia son de interacción que resaltan el orden temporal de los mensajes. Son conocidos por ser un “diagrama que representa una interacción, poniendo el foco en la secuencia de los mensajes que se intercambian, junto con sus correspondientes ocurrencias de eventos en las Líneas de Vida” ( Formación en Ambientes Virtuales de Aprendizaje, s/n, p. 5).

**Figura 2.19**  
Diagramas de secuencias

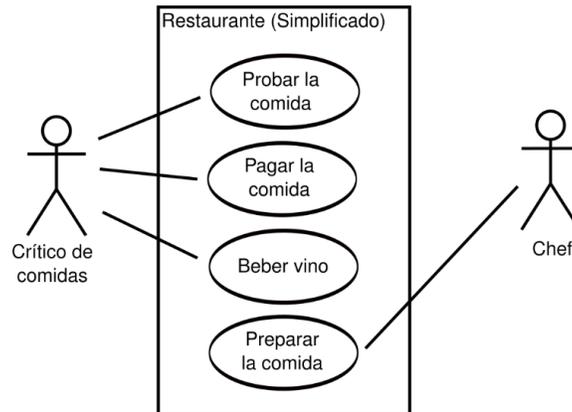


**Nota:** Formación en Ambientes Virtuales de Aprendizaje, s/n, p. 5

### 2.2.4.2.4. Diagramas de casos de uso

“Los diagramas de casos de uso son responsables principalmente de documentar los macrorrequisitos del sistema” (Kimmel, 2008, p. 7). Así, este diagrama “muestra las Media relaciones entre los actores y el sujeto (sistema), y los casos de uso” ( Formación en Ambientes Virtuales de Aprendizaje, s/n, p. 6). Estos diagramas son especialmente importantes en el modelado y organización del comportamiento de un sistema (Booch y et al., 2006, p. 29).

**Figura 2.20**  
Diagrama de caso de uso

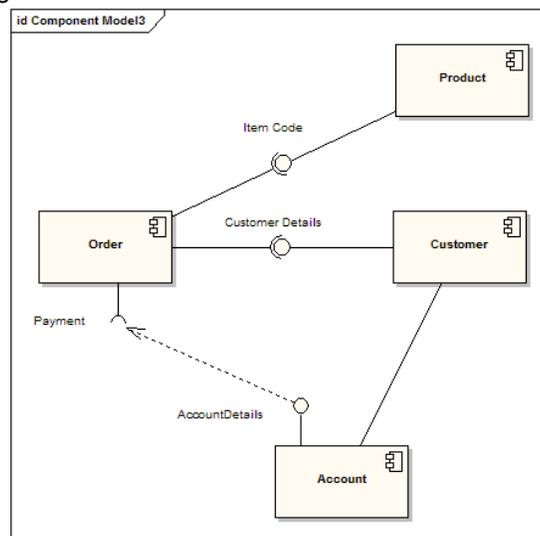


**Nota:** Booch y et al., 2006, p. 29

#### 2.2.4.2.5. Diagramas de componentes

Este tipo de diagramas “representa la encapsulación de una clase, conjunto con sus interfaces, puertos y estructura interna, la cual está formada por otros componentes anidados y conectores” (Booch & et al., 2006, p. 28). Los diagramas de componentes (componentes, sus relaciones, interacciones y sus interfaces públicas) son las que conforman una aplicación, sistema o empresa (Formación en Ambientes Virtuales de Aprendizaje, s/n, p. 6).

**Figura 2.22**  
Diagramas de componentes

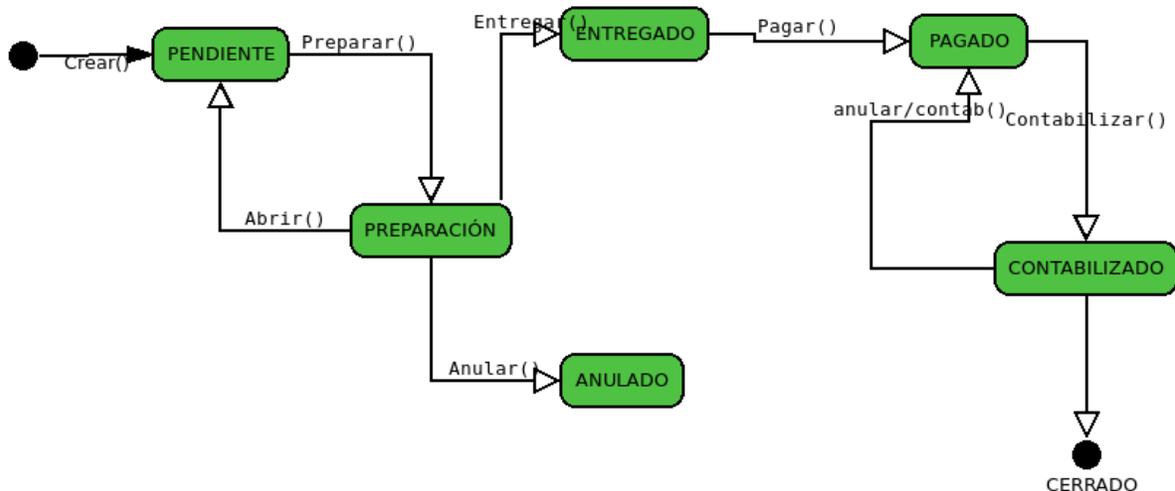


**Nota:** Formación en Ambientes Virtuales de Aprendizaje, s/n, p. 6

### 2.2.4.2.6. Diagramas de estado

El diagrama de estado “ilustra cómo un elemento (...) se mueve entre estados que clasifican su comportamiento, de acuerdo con disparadores de transiciones, guardias de restricciones y otros aspectos de los diagramas de Máquinas de Estados, que representan y explican el movimiento y el comportamiento” (Formación en Ambientes Virtuales de Aprendizaje, s/n, p. 6). Los diagramas de estado son “especialmente importantes en el modelado del comportamiento de una interfaz, una clase o una colaboración y resaltan el comportamiento dirigido por eventos de un objeto, los cual es especialmente útil en el modelado de sistema reactivos” (Booch y et al., 2006, p. 29).

**Figura 2.23**  
Diagramas de estado



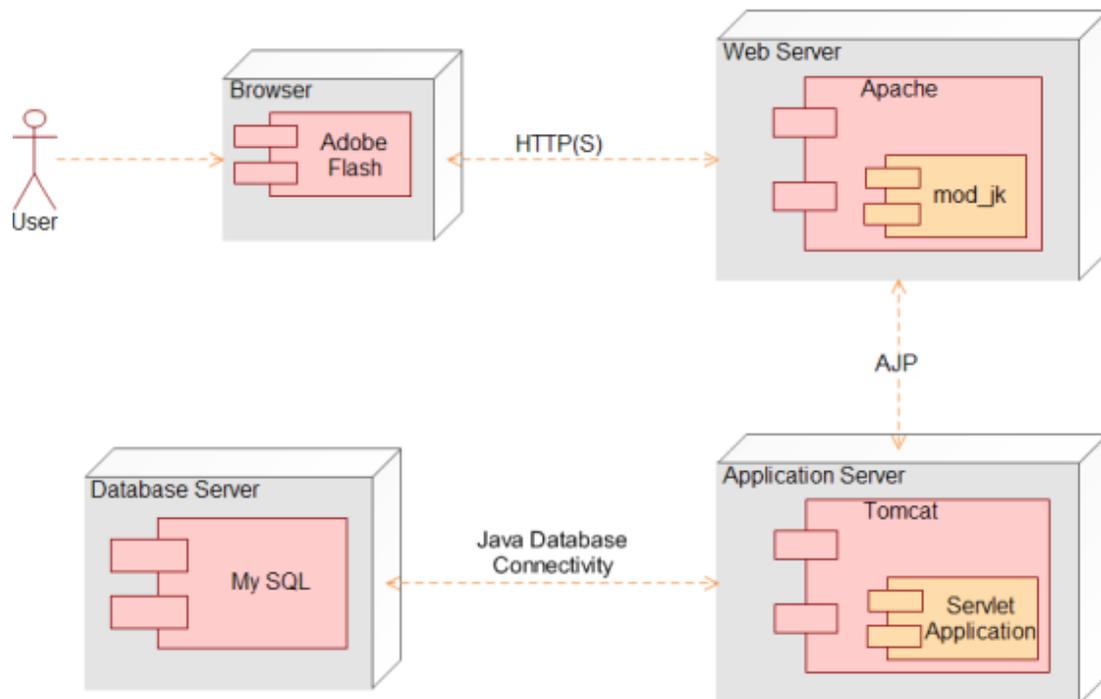
**Nota:** Booch y et al., 2006, p. 29

### 2.2.4.2.7. Diagramas de despliegue

Un diagrama de despliegue “mostrará cómo se verá desplegado su sistema. Comúnmente, un diagrama de este tipo contiene símbolos que representen cosas, como servidores web, servidores de bases de datos y varios dispositivos diversos, así como software que constituye la solución” (Kimmel, 2008, p. 12). “Este tipo de diagramas muestra la configuración de nodos de procesamiento en tiempo de ejecución y artefactos de residen en ellos. Normalmente, un nodo alberga y no o más

artefectos” (Booch y et al., 2006, p. 29). “Las máquinas físicas y los procesadores se representan como nodos y la construcción interna puede ser representada por nodos. Como los artefactos se ubican en los nodos (...), la ubicación es guiada por el uso de las especificaciones de despliegue” (Formación en Ambientes Virtuales de Aprendizaje, s/n, p. 6).

**Figura 2.24**  
Diagramas de despliegue



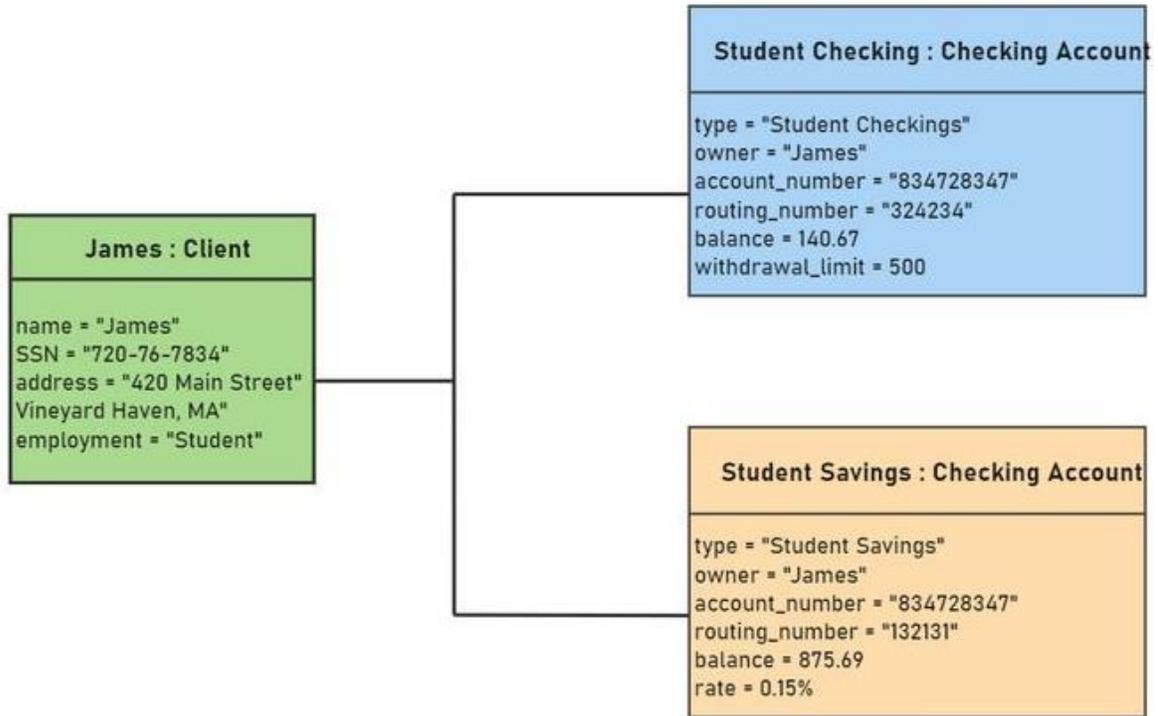
**Nota:** Formación en Ambientes Virtuales de Aprendizaje, s/n, p. 6

#### 2.2.4.2.8. Diagramas de objetos

Son diagramas que representan a los objetos y sus relaciones en un punto en el tiempo. “Estos diagramas representan instantáneas estáticas de instancias de los elementos existentes en los diagramas de clase” (Booch & et al., 2006, pág. 28). “Un diagrama de objetos se puede considerar como un caso especial de un diagrama de clases o un diagrama de comunicaciones” (Formación en Ambientes Virtuales de Aprendizaje, s/n, p. 6).

**Figura 2.25**

Diagramas de objetos

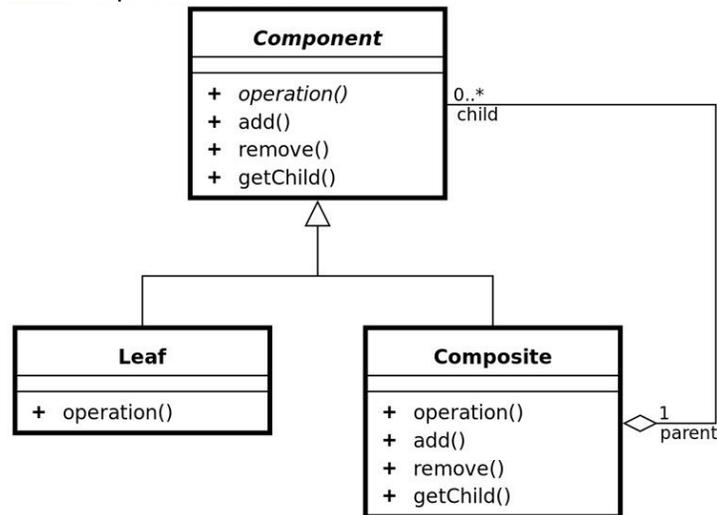


**Nota:** Formación en Ambientes Virtuales de Aprendizaje, s/n, p. 6).

#### 2.2.4.2.9. Diagramas de estructura compuesta

Este tipo de diagrama “representa la estructura interna de un clasificador, incluyendo los puntos de interacción de clasificador con otras partes del sistema” (Formación en Ambientes Virtuales de Aprendizaje, s/n, p. 6). “El diagrama de estructura compuesta es aplicable a cualquier clase, y un diagrama de componente” (Booch y et al., 2006, p. 29).

**Figura 2.26**  
Diagramas de estructura compuesta

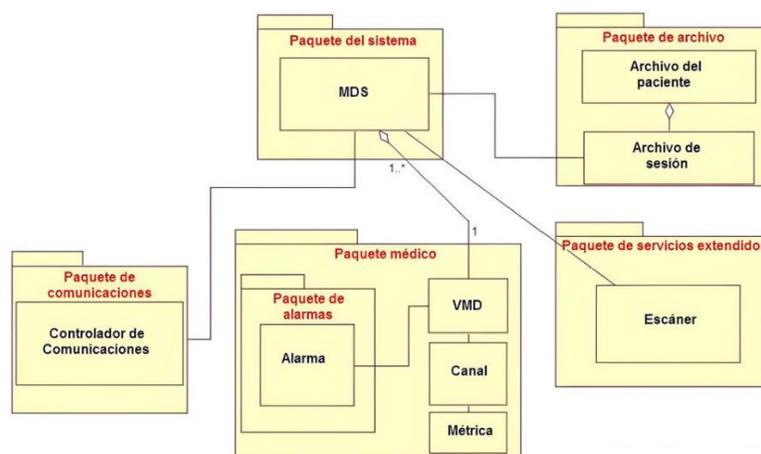


**Nota:** Booch y et al., 2006, p. 29

#### 2.2.4.2.10. Diagramas de paquetes

Este tipo de diagrama “muestra la composición del propio modelo en unidades organizativas y sus dependencias” (Booch et al., 2006, p. 28). Un diagrama “representa cómo se Baja organizan los elementos de modelado en paquetes y las dependencias entre ellos, incluyendo importaciones y extensiones de paquetes” ( Formación en Ambientes Virtuales de Aprendizaje, s/n).

**Figura 2.27**  
Diagramas de paquetes

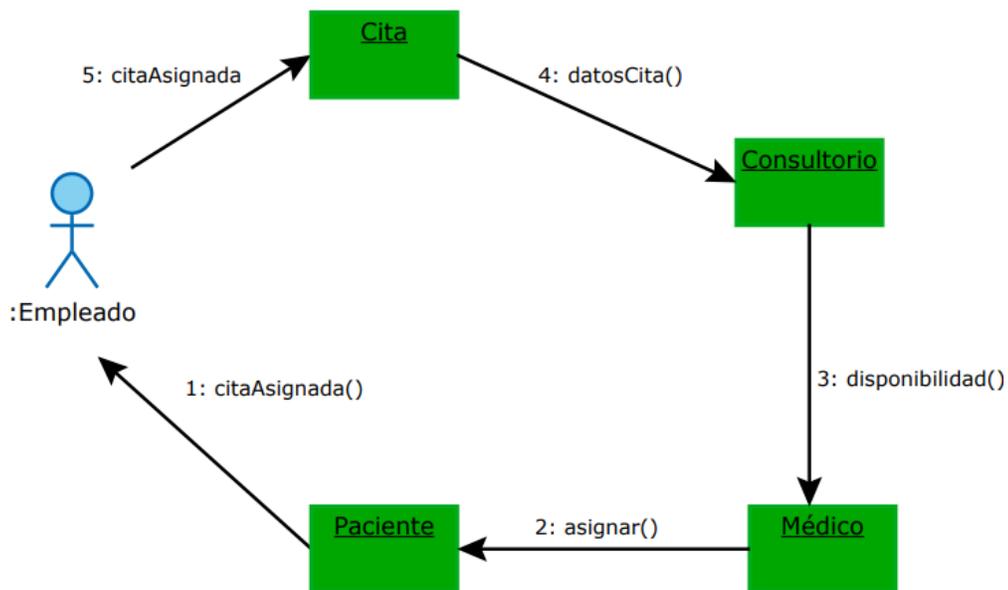


**Nota:** Formación en Ambientes Virtuales de Aprendizaje, s/n

### 2.2.4.2.11. Diagramas de comunicación

Inicialmente, se debe aclarar que diagrama de comunicación o colaboración son denominaciones exactamente iguales, que han sido distinguidos a partir de las versiones 1 y 2 de UML. Son un tipo de diagrama de “interacción que resalta la organización estructural de los objetos o roles que envían y reciben mensajes” (Booch & et al., 2006, p. 29). Así estos resaltan la estructura de los datos a través de la cual fluyen los mensajes. El diagrama de comunicación “hace énfasis en la relación entre los objetos para satisfacer una operación, generando una visión espacial del sistema en la ejecución de un proceso” (Formación en Ambientes Virtuales de Aprendizaje, s/n, p. 18).

**Figura 2.28**  
Diagramas de comunicación



**Nota:** Formación en Ambientes Virtuales de Aprendizaje, s/n

### 2.2.4.2.12. Diagramas de tiempos

“Es un diagrama de interacción que muestra los tiempos reales entre diferentes objetos o roles” (Booch & et al., 2006, p. 30). Siendo así, “El propósito primario del diagrama de tiempos es mostrar los cambios en el estado o la condición de una línea de vida (...) a lo largo del tiempo lineal. El uso más común es mostrar el cambio de estado de

un objeto a lo largo del tiempo, en respuesta a los eventos o estímulos aceptados” ( Formación en Ambientes Virtuales de Aprendizaje, s/n, p. 6).

### **2.2.5. Lenguaje de modelado UWE**

El lenguaje de modelado unificado (UWE) es uno de los modelos más usados especialmente para aplicaciones web. “El enfoque principal es proporcionar un lenguaje de modelado específico de dominio basado en UML que también incluye funciones de seguridad (...) y una metodología basada en modelos” (Nava & Otros, 2019, p. 4). La notación de UWE se define como una extensión liviana que proporciona un llamado perfil de UML.

To ensure the quality of UML-based models and to apply the correct V&V checklist to those models, it is essential to focus on the objectives of the modeling exercise and use the UML-based diagrams that will help the modeler archive these objectives. Thus, the applicability of UML diagrams differs from project to project. The intensity of the application of these diagrams in creating the models also differs, depending on the reason for modeling when creating the diagrams. The purpose for creating the diagrams has a direct bearing on the way they are creating, extended, and of course, verified and validated. The modeling spaces are extremely helpful in clarifying the purpose of modeling, particularly its role (Unhelkar, 2005, p. 8).

Existe muchas ventajas de esta metodología, y uno de ellas es el “estándar UML es la flexibilidad de éste para la definición de un lenguaje de modelado específico para el dominio web y sobre todo la aceptación universal de dicho estándar en el campo de la ingeniería del software” (Flores y et. al, 2015, p. 17).

#### **2.2.5.1. Características de UWE**

UWE “cubre todo el ciclo de vida de este tipo de aplicaciones centrando además su atención en aplicaciones personalizadas o adaptativas (Flores y et. al, 2015, p. 18)”.

Como metodología cuenta con tres importantes características o aspectos donde se fundamenta:

- **Uso de notación estándar.** Para todos los modelos utiliza el lenguaje UML.
- **Definición de métodos.** Siempre realiza la definición de los pasos para la construcción de diferentes modelos.
- **Especificación de restricciones.** Utiliza el recomendado lenguaje de restricciones de objetos (OCL) para aumentar la exactitud de los modelos.

#### **2.2.5.2. Fases de la metodología UWE**

Existe seis fases en la metodología UWE que cubren todo el ciclo de vida de una aplicación:

- Análisis y especificación de requisitos
- Diseño del sistema
- Codificación del software
- Fase de pruebas
- Fase de implementación
- Mantenimiento

##### **2.2.5.2.1. Análisis y especificación de requisitos**

En esta fase “se adquieren, reúnen y especifican las características funcionales y no funcionales que deberá cumplir la aplicación web” (Galiano, 2012).

Trata de diferente forma las necesidades de información, las necesidades de navegación, las necesidades de adaptación y las de interfaz de usuario, así como algunos requisitos adicionales. Centra el trabajo en el estudio de los casos de uso, la generación de los glosarios y el prototipado de la interfaz de usuario (Flores y et. al, 2015, p. 18).

Para este propósito, en esta fase UWE utiliza el diagrama de caso de uso, que “sirven para especificar la comunicación y el comportamiento de un sistema mediante su interacción con los usuarios y otros sistemas” (Flores et. al, 2015, p. 18). De esta

manera, se ilustran los requerimientos del sistema de forma gráfica al mostrar cómo reacciona a eventos realizados por los actores del Sistema.

Los productos obtenidos de esta etapa son: modelos de casos de uso y documentos de descripción de caso de uso, mediante la técnica de modelado conceptual.

#### **2.2.5.2.2. Diseño del sistema**

Esta fase “se basa en la especificación de requisitos producido por el análisis de los requerimientos (fase de análisis), el diseño define cómo estos requisitos se cumplirán, la estructura que debe darse a la aplicación web” (Galiano, 2012). En esta fase de diseño se establecen “el diseño estructural del sistema, (...) normas de desarrollo (diccionario\_datos), los diagramas de desarrollo de las Bases de Datos y la documentación necesaria para realizar la gestión de la Configuración durante el Ciclo de vida del desarrollo de Software” (Flores y et. al, 2015, p. 19).

En esta fase, se indica cómo se enlazan los elementos de navegación. Para esto, se utilizan unidades de navegación denominados “nodos” conectados por enlaces de navegación. Estos nodos pueden ser mostrados en la misma página web, no tienen por qué estar en páginas diferentes. Por tanto, los productos que se obtienen en esta fase son el modelo de entidad relación del BD y los diagramas de navegación a través de las técnicas de modelado conceptual.

#### **2.2.5.2.3. Codificación del software**

Esta consiste en la ejecución de “las tareas que comúnmente se conocen como programación; que consiste, esencialmente, en llevar a código fuente, en el lenguaje de programación elegido, todo lo diseñado en la fase anterior” (Galiano, 2012). El resultado obtenido de esta etapa es el código fuentes que será realizado utilizando un lenguaje de programación.

#### **2.2.5.2.4. Fase de pruebas**

Esta fase se implementa para evitar fallas en el software. Esto implica que “las pruebas se utilizan para asegurar el correcto funcionamiento de secciones de código” (Galiano,

2012). “Generalmente las pruebas se realizan sobre un prototipo del producto software y son ejecutadas por personas ajenas al grupo que escribió los programas originales” (Flores y et. al, 2015, p. 20). Con esto, se desea asegurar que las pruebas sean completas e imparciales, además el software sea confiable.

De acuerdo a Adrian (2010 citado en Flores y et. al, 2015) existen varios tipos de pruebas. Entre los más destacados son:

- **Pruebas de caja Negra:** Se enfoca en los requerimientos establecidos y en la funcionalidad del sistema.
- **Pruebas de Caja Blanca:** Se basan en el conocimiento de la lógica del código del sistema.
- **Pruebas de Integración:** Buscan probar la combinación de las distintas partes de la aplicación para determinar si funcionan correctamente en el conjunto.
- **Pruebas de Contenido:** Buscan verificar que el contenido del sistema sea coherente y consistente a la vez.
- **Pruebas del Sistema:** Buscan probar al sistema como un todo.
- **Prueba de Funcionalidad:** Examina si el sistema cubre sus necesidades de funcionamiento, acorde a las especificaciones de diseño.
- **Prueba de Usabilidad:** Tienen la finalidad de verificar que tan fácil de usar es el sistema.

Al final es esta fase, el producto obtenido será el plan de pruebas, obtenidas a través de las técnicas para la implementación de pruebas de software.

#### **2.2.5.2.5. Fase de implementación**

En esta fase de la metodología UWE, “los programas desarrollados son transferidos apropiadamente al host de destino, inicializados y configurados eventualmente; todo ello con el propósito de ser utilizados por el usuario final” (Flores & et. al, 2015, p. 21). Esto implica la implementación “de la arquitectura, de la estructura del hiperespacio, del modelo de usuario, de la interfaz de usuario, de los mecanismos adaptativos y las tareas referentes a la integración” (Ídem).

Básicamente esta fase es un proceso donde se verifica, se entrena a los usuarios, instala la aplicación y construye todos los archivos de datos necesarios para usarla. Finalmente, esta concluye con la instalación del ejecutable de la aplicación.

#### **2.2.5.2.6. Mantenimiento**

Esta fase consiste en el “control, mejora y optimización del software ya desarrollado e instalado, que también incluye depuración de errores y defectos que puedan haberse filtrado de la fase de pruebas de control” (Flores y et. al, 2015, p. 22).

#### **2.2.6. Arquitectura software**

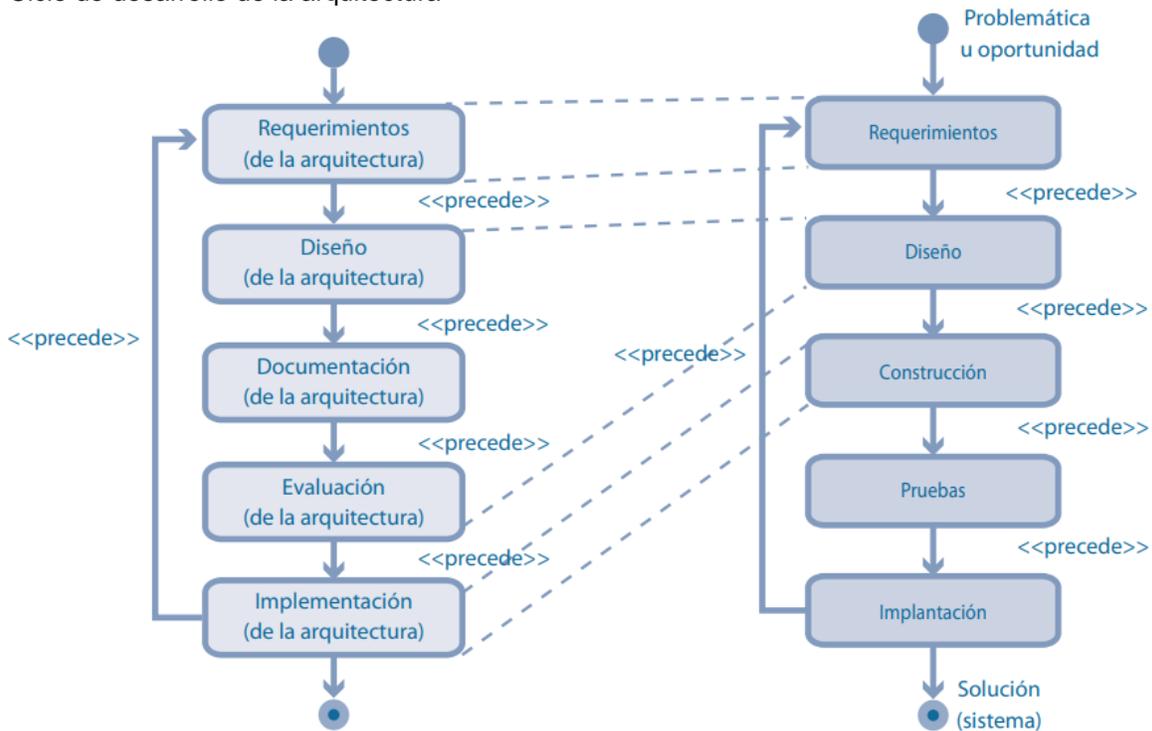
El término arquitectura de software ha sido incursionado por primera vez en el trabajo de Perry y Wolf (1992). Este año se convirtió en “el punto de partida para para lo que hoy conocemos como arquitectura de software” (Fernández, 2006, p. 2). Perry y Wolf son los primeros que proponen un modelo para la arquitectura de software. Según Bass (citado en Fernández, 2006, p. 2) “una arquitectura de software de un programa o un sistema computacional es la estructura del sistema, la cual comprende elementos de software, las propiedades externamente visibles de esos elementos, y las relaciones entre ellos”.

La arquitectura de software es “una rama de la Ingeniería de Software dedicada al estudio de la estructura de los sistemas software complejos. Trata y estudia concretamente uno de los problemas pendientes del campo: la especificación y descripción de arquitecturas de software dinámicas” (Cuesta, 2002).

##### **2.2.6.1 Ciclo de desarrollo de la arquitectura**

El ciclo de desarrollo de la arquitectura está constituido por los requerimientos, diseño, documentación, evaluación e implementación.

**Figura 2.29**  
Ciclo de desarrollo de la arquitectura



**Nota:** Humberto Cervantes, Perla Velasco y Luis Castro Careaga

- **Requerimientos.** “Esta etapa se enfoca en la captura, documentación y priorización de requerimientos que influyen sobre la arquitectura y que, por lo habitual, se conocen en inglés como drivers arquitectónicos” (Cervantes & et. al, 2016).
- **Diseño.** Esta fase se “creación estructural se hace por lo habitual con base en dos clases de soluciones abstractas probadas, llamadas patrones de diseño y tácticas, al igual que en soluciones concretas como las elecciones tecnológicas, tales como los frameworks” (Cervantes y et. al, 2016, p. 5).
- **Documentación.** “La documentación formal involucra la representación sus estructuras por medio de vistas. Una vista representa una estructura y contiene por lo habitual un diagrama, además de información adicional que apoya en la comprensión de este” (Cervantes & et. al, 2016, p. 5).
- **Evaluación.** Esto implica identificar los posibles riesgos o problemas a través de la evaluación del diseño que ha sido documentado. “La ventaja de la evaluación es (...) que puede realizarse de manera temprana (...), y que el

costo de corrección de los defectos identificados por medio de ella es mucho menor al costo que tendría enmendarlos después de que haya sido construido” (Cervantes y et. al, 2016, p. 5).

- **Implementación.** Es la etapa donde se pone en prueba y ejecución el sistema.

### **2.2.6.2. Beneficios de arquitectura**

Existe tres beneficios de la arquitectura de software: aumentar la calidad de los sistemas, mejorar los tiempos en la entrega de los proyectos y reducir los costos del desarrollo.

### **2.2.6.3. Rol del arquitecto**

El rol principal del arquitecto “es tomar decisiones de diseño pertinentes, a efecto de satisfacer los drivers arquitectónicos y demás requerimientos del sistema. El arquitecto debe comunicar sus decisiones y asegurar que durante la construcción del sistema estas sean respetadas por parte de los miembros del equipo de desarrollo” (Cervantes y et. al, 2016, p. 8).

### **2.2.7. Calidad de software**

Calidad de software es un término que se aplica para referirse “al grado de desempeño de las principales características con las que debe cumplir un sistema computacional durante su ciclo de vida, (...) estas garantizan al cliente contar con un sistema confiable, (...) aumentando su satisfacción frente a la funcionalidad y eficiencia del sistema construido” (Callejas et. al, 2017).

Para Pressman (2010, citado en Callejas 2017), la calidad de software se asocia a la “concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos con los estándares de desarrollo plenamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente”.

Por su parte el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE, 1990) define calidad de software como "el grado con el que un sistema, componente o proceso

cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario". Así se enfatiza en los requisitos específicos del sistema y en la búsqueda de la satisfacción del cliente.

Por esta razón, Callejas y otros en el 2017 mencionan "para garantizar la calidad de software es importante implementar algún modelo o estándar de calidad que permita la gestión de atributos en el proceso de construcción de software, considerando la concordancia de los requisitos y su construcción".

### **2.2.7.1. Calidad del producto software**

Según la serie ISO/IEC 25000 existe dos modelos de calidad. La primera está referida a la calidad interna y externa, mientras que el segundo modelo hace referencia a la calidad de uso.

#### **2.2.7.1.1. Calidad interna y externa**

Según la normativa ISO/IEC 25000 del 2010, la calidad interna se define como:

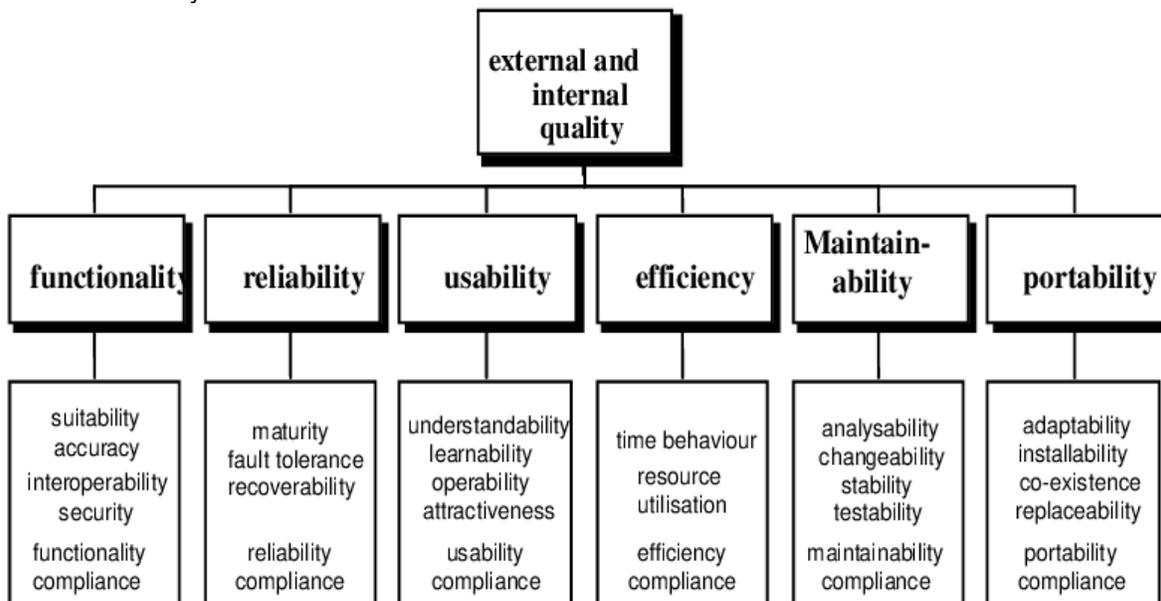
La totalidad de las características del producto software desde una perspectiva interna. La calidad interna es medida y evaluada en base a los requerimientos de calidad interna. Los detalles de la calidad del producto software pueden ser mejorados durante la implementación, revisión y prueba del código software, pero la naturaleza fundamental de la calidad del producto software representada por la calidad interna permanece sin cambios a menos que sea rediseñado (Baldeón, 2015, p. 28).

Según esta misma normativa la calidad externa se define como:

La totalidad de las características del producto software desde una perspectiva externa. Es la calidad cuando el software es ejecutado, la cual es típicamente medida y evaluada mientras se prueba en un ambiente simulado con datos simulados y usando métricas externas. Durante las pruebas, muchas fallas serán descubiertas y eliminadas. Sin

embargo, algunas fallas todavía pueden permanecer después de las pruebas. Como es difícil corregir la arquitectura de software u otros aspectos fundamentales del diseño del software, el diseño fundamental permanece sin cambios a través de las pruebas. (Baldeón, 2015, p. 28).

**Figura 2.30**  
Calidad interna y externa



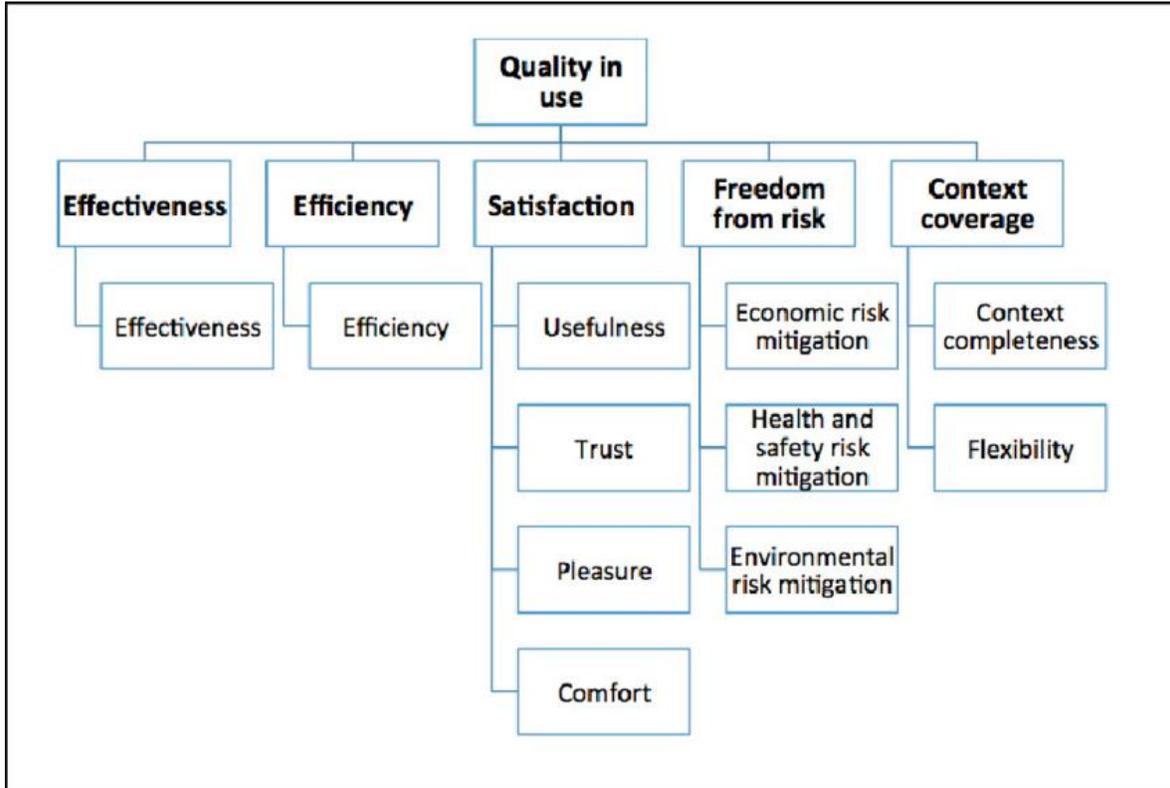
**Nota:** Baldeón, 2015, p. 28

### 2.2.7.1.2. Calidad de uso

Según la norma ISO/IEC 25000 del 2010, la calidad de uso es:

La perspectiva del usuario de la calidad del producto software cuando este es usado en un ambiente específico y un contexto de uso específico. Esta mide la extensión para la cual los usuarios pueden conseguir sus metas en un ambiente particular, en vez de medir las propiedades del software en sí mismo (Baldeón, 2015, p. 29).

**Figura 2.31**  
Calidad de uso



**Nota:** Baldeón, 2015, p. 29

### **2.2.7.2. Calidad en la industria del software (ISO-9126)**

En 1992 fue publicado la ISO-9126 con el carácter de estándar internacional para la evaluación de la calidad de software con el nombre de “Information technology – Software product evaluation: Quality characteristics and guidelines for their use”. Donde se estableció las características de calidad para productos de software. Las características básicas descritas por esta norma son seis: funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portatibilidad (Abud, 2012, p. 1).

#### **2.2.7.2.1. Funcionabilidad**

Para esta característica se considera los siguientes atributos:

- **Adecuación.** Se enfoca a evaluar si el software cuenta con un conjunto de funciones apropiadas para efectuar las tareas que fueron especificadas en su definición (Baldeón, 2015, p. 2).
- **Exactitud.** Este atributo permite evaluar si el software presenta resultados o efectos acordes a las necesidades para las cuales fue creado (Baldeón, 2015, p. 2).
- **Interoperabilidad.** Permite evaluar la habilidad del software de interactuar con otros sistemas previamente especificados (Baldeón, 2015, p. 2).
- **Conformidad.** Evalúa si el software se adhiere a estándares, convenciones o regulaciones en leyes y prescripciones similares (Baldeón, 2015, p. 2).
- **Seguridad.** Se refiere a la habilidad de prevenir el acceso no autorizado, ya sea accidental o premeditado, a los programas y datos (Baldeón, 2015, p. 2).

#### 2.2.7.2.2. Confiabilidad

Esta característica tiene la capacidad de mantener las prestaciones requeridas del sistema con la siguientes sub-características:

- **Madurez.** Se debe verificar las fallas del sistema y si muchas de estas han sido eliminadas durante el tiempo de pruebas o uso del sistema (Coaquira, 2017, p. 33).
- **Recuperabilidad.** Verificar si el software puede reasumir el funcionamiento y restaurar datos perdidos después de un fallo ocasional (Ídem).
- **Tolerancia a fallos.** Evalúa si la aplicación desarrollada es capaz de manejar errores (Coaquira, 2017, p. 33).

#### 2.2.7.2.3. Usabilidad

La usabilidad es un conjunto de atributos que permiten evaluar el esfuerzo necesario que deberá invertir el usuario para utilizar el Sistema. Cuenta con tres sub-características:

- **Comprensibilidad.** Se refiere al esfuerzo requerido por los usuarios para reconocer la estructura lógica del sistema y los conceptos relativos a la aplicación del software (Abud, 2012).
- **Facilidad de Aprender.** Establece atributos del software relativos al esfuerzo que los usuarios deben hacer para aprender a usar la aplicación (Abud, 2012).
- **Operabilidad.** Agrupa los conceptos que evalúan la operación y el control del sistema (Abud, 2012).

#### 2.2.7.2.4. Eficiencia

Permite la evaluación relacionada entre niveles de funcionamiento del software y la cantidad de recurso utilizados. Cuenta con dos aspectos:

- **Comportamiento con respecto al tiempo.** Atributos del software relativos a los tiempos de respuesta y de procesamiento de los datos (Abud, 2012).
- **Comportamiento con respecto a recursos.** Atributos del software relativos a la cantidad de recursos usados y la duración de su uso en la realización de sus funciones (Abud, 2012).

#### 2.2.7.2.5. Mantenibilidad

Esta característica se refiere a los atributos que permiten medir el esfuerzo necesario para realizar la modificación al sistema.

- **Capacidad de análisis.** Relativo al esfuerzo necesario para diagnosticar las deficiencias o causas de fallas, o para identificar las partes que deberán ser modificadas.
- **Capacidad de modificación.** Mide el esfuerzo necesario para modificar aspectos del software, remover fallas o adaptar el software para que funcione en un ambiente diferente.
- **Estabilidad.** Permite evaluar los riesgos de efectos inesperados debidos a las modificaciones realizadas al software.

- **Facilidad de Prueba.** Se refiere al esfuerzo necesario para validar el software una vez que fue modificado (Abud, 2012, p. 2).

#### **2.2.7.2.6. Portabilidad**

Esta característica es relacionada con las habilidades del software para ser transferido de un ambiente a otro de acuerdo a las siguientes sub-características:

- **Adaptabilidad.** Evalúa la oportunidad para adaptar el software a diferentes ambientes sin necesidad de aplicarle modificaciones (Abud, 2012, p. 2).
- **Facilidad de Instalación.** Es el esfuerzo necesario para instalar el software en un ambiente determinado (Abud, 2012, p. 2).
- **Conformidad.** Permite evaluar si el software se adhiere a estándares o convenciones relativas a portabilidad (Abud, 2012, p. 2).
- **Capacidad de reemplazo.** Se refiere a la oportunidad y el esfuerzo usado en sustituir el software por otro producto con funciones similares (Abud, 2012, p. 2).

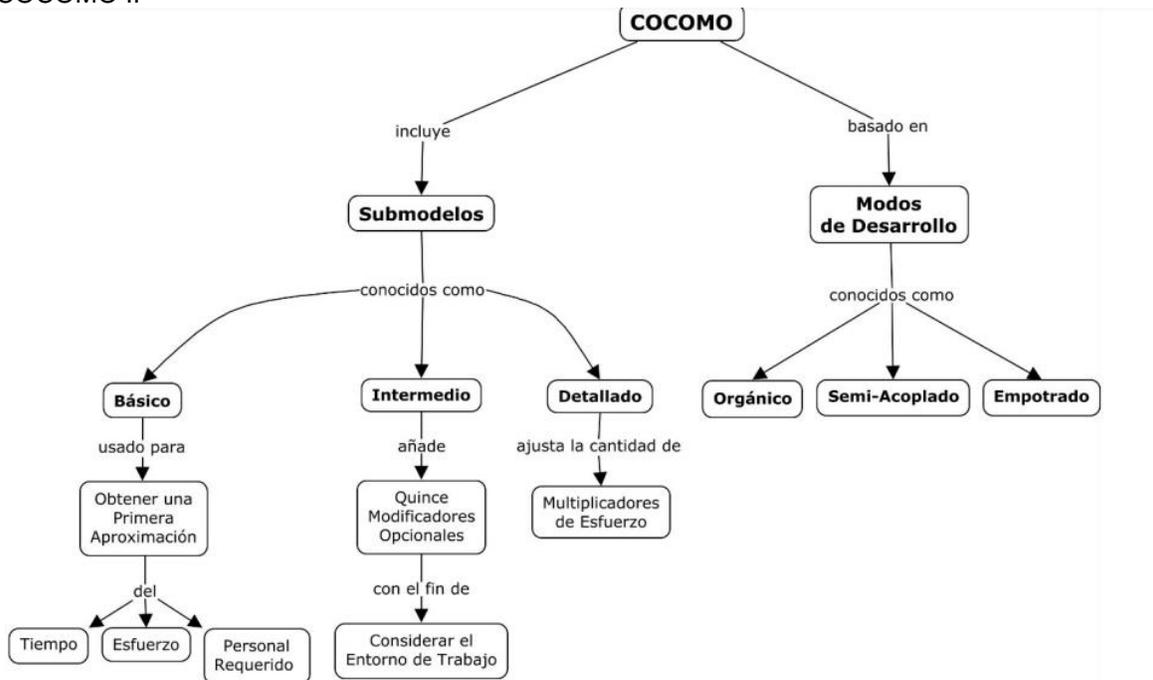
#### **2.2.8. Modelo de costos de software**

Los modelos algorítmicos de estimación de costos en proyectos de software se utilizan para ayudar en la gestión de riesgos, planificar - controlar el proyecto y enriquecer el análisis de la inversión de los proyectos en software.

##### **2.2.8.1. COCOMO II**

Según Garita (2014, citado en Gonzales y Madriz, 2018) COCOMO (COConstructive COSt MOdel) “es un modelo de formulación matemática con un fuerte componente de base empírica, principalmente utilizado para estimación de costos en los proyectos de software”.

**Figura 2.32**  
COCOMO II



**Nota:** Gonzales y Madriz, 2018, pág. 123

Ecuaciones de tipo de modelo COCOMO intermedio:

El esfuerzo necesario en mese/persona para concretar un proyecto de desarrollo de software

$$E = a * (KLDC)^b * ME$$

Tiempo del calendario en meses que transcurre desde la determinación de los requerimientos a la culminación de una actividad.

$$T = c * (E)^d$$

Personal requerido para la ejecución del proyecto.

$$P = E/T$$

# **CAPÍTULO III**

## MARCO APLICATIVO

En este capítulo, se aplica los conceptos descritos del capítulo anterior. También se describe el desarrollo del proyecto “sistema web para la administración en los servicios de traducción en el Consultorio Lingüístico”. La ejecución de este proyecto sigue lineamientos de la metodología SCRUM, utilizando el lenguaje de Modelado UWE. Se aplica las fases del modelo UWE: análisis, diseño, pruebas, implementación y mantenimiento. Por tanto, los modelos que se aplican en este capítulo son; modelo de requerimiento, modelo conceptual, modelo de navegación y modelo de presentación.

### 3.1. Diagnóstico del sistema

#### 3.1.1. Descripción de los actores

Para que el sistema pueda cubrir con los procesos necesarios del Consultorio Lingüístico, se identifica a los principales actores que interactúan con el sistema.

**Tabla 3.1.**  
Lista de actores en el sistema

DESCRIPCIÓN	RESPONSABILIDADES
<b>Administrador</b>	<ul style="list-style-type: none"><li>- Administra espacios del consultorio en el sistema.</li><li>- Asigna permisos a cada usuario</li></ul>
<b>Coordinador (responsable)</b>	<ul style="list-style-type: none"><li>- Administra y dirige el consultorio.</li><li>- Planifica, organiza y ejecuta los servicios de traducción.</li><li>- Controla y realiza el seguimiento de las actividades del consultorio.</li></ul>
<b>Traductor</b>	<ul style="list-style-type: none"><li>- Se encarga de recibir el material de traducción.</li><li>- Traducir el material escrito en otro idioma requerido.</li></ul>

---

<b>Intérprete</b>	- Se encarga de cumplir con la transmisión de un discurso de tipo oral en la institución que lo requiera.
<b>Cliente</b>	<ul style="list-style-type: none"> <li>- Es el usuario final del sistema.</li> <li>- Interactúa con el traductor.</li> <li>- Solicita material escrito a traducir</li> <li>- Adquiere material traducido</li> </ul>

---

### **3.1.2. Descripción de requerimientos**

Un requerimiento se visualiza como una declaración abstracta de muy alto nivel de servicios. UWE – UML- based Web Engineering asegura que el sitio web no solo cumpla con las expectativas funcionales de los usuarios, sino que también esté preparado para desplegar servicios de traducción de manera ubicua y segura en el entorno web, cumpliendo con los más altos estándares de calidad y eficiencia.

#### **3.1.2.1. Requerimientos Funcionales**

Los requisitos funcionales muestran las características necesarias del sistema, donde se muestra un proceso de identificación, análisis y documentación de las funcionalidades específicas del sistema. Esto abarca desde la gestión de solicitudes de traducción, el registro de usuarios y la interacción con los mismos, hasta la administración de documentos y pagos. Cada requerimiento funcional se describe con detalle, especificando su propósito, comportamiento esperado, y los casos de uso relacionados, con el objetivo de asegurar que la aplicación cumpla con las necesidades y expectativas de los usuarios de manera precisa y efectiva, manteniendo la coherencia con los principios de la ingeniería web de UWE. Dichos requerimientos se describen a continuación en la siguiente tabla.

**Tabla 3.2.**  
Requerimientos funcionales

<b>ROL</b>	<b>DESCRIPCIÓN</b>	<b>FUNCIÓN</b>
<b>R1</b>	Administración de usuarios.	Acceder al sistema por tipos de usuario (super administración, coordinador, traductor y cliente).
<b>R2</b>	Administrar Roles (Permisos)	Permite la asignación de permisos a los usuarios (super administración, coordinador, traductor y cliente).
<b>R3</b>	Registro de los traductores e interpretes	Permitir el registro de todos los traductores.
<b>R4</b>	Registro de clientes	Permitir el registro de los datos del cliente
<b>R5</b>	Control y seguimiento de las solicitudes de traducción	Permitir a la administración generar reportes.
<b>R6</b>	Notificación de las solicitudes de	Emitirá notificaciones por las nuevas solicitudes de traducción.
<b>R7</b>	Imprimir comprobante de solicitud de traducción	Emitir comprobante de solicitud de traducción al cliente.
<b>R8</b>	Imprimir reporte de las solicitudes	Emitir un reporte de las actividades realizadas

### **3.1.2.1. Requerimientos no funcionales**

La atención a los requerimientos no funcionales se torna crucial para garantizar la calidad y la eficacia del sitio web de servicios de traducción. Estos requerimientos, de naturaleza más amplia y no relacionados directamente con las funcionalidades

específicas, abarcan aspectos vitales como la seguridad de los datos, la escalabilidad para gestionar posibles aumentos de usuarios, la usabilidad para proporcionar una experiencia amigable y efectiva, y la compatibilidad con diversas plataformas y navegadores. Cada requerimiento no funcional se describe en detalle, especificando criterios medibles y estándares a cumplir, asegurando que el sitio web sea robusto, seguro y satisfaga las necesidades de los usuarios, independientemente de la plataforma o dispositivo desde el cual accedan. Este enfoque garantiza que el proyecto cumpla con los estándares más exigentes de la ingeniería web y ofrezca una experiencia confiable y de alta calidad a los usuarios.

**Tabla 3.3.**  
Requerimientos no funcionales

<b>ROL</b>	<b>FUNCIÓN</b>
<b>R1</b>	Presentar interfaz responsiva para poder adaptarse a los diferentes dispositivos (computadas, tabletas o teléfonos inteligentes).
<b>R2</b>	Visualizar el correcto funcionamiento en distintos navegadores.
<b>R3</b>	Poseer registro histórico de los servicios de traducción
<b>R4</b>	Soporte y mantenimiento periódico para lograr un funcionamiento óptimo del sistema.

### **3.2. Análisis del sistema**

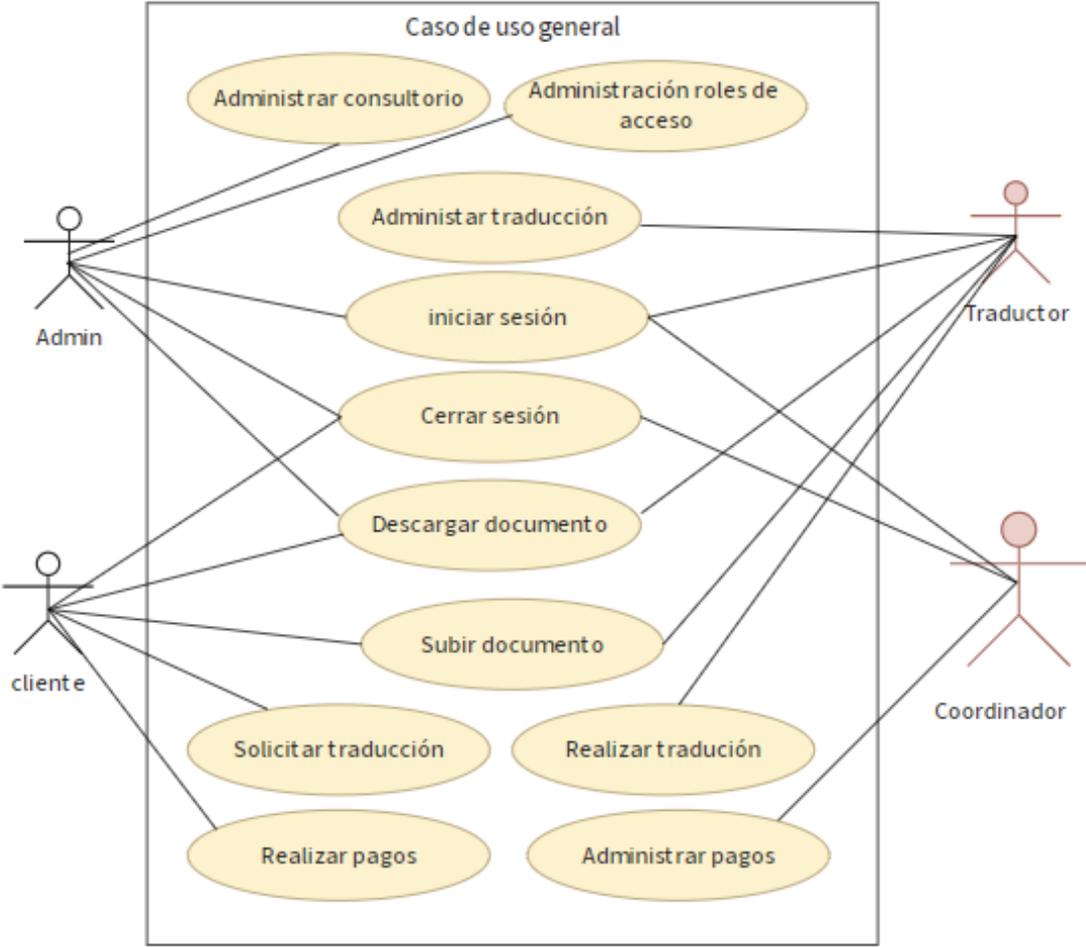
En esta sección se presenta los modelos de caso de uso, los diagramas de secuencia y los diagramas de actividad del sistema informático.

#### **3.2.1. Caso de uso general**

Se aplicará los diagramas de caso de uso para representar e identificar los diferentes procesos y actividades que se realiza en el consultorio y los actores del sistema. Este diagrama, con un enfoque sistemático, visualiza las relaciones y las interacciones entre los diferentes actores y las funcionalidades del sistema en un contexto más amplio. Desde el primer contacto del usuario con la plataforma, como el proceso de registro, hasta la ejecución de acciones clave, como la solicitud de servicios de traducción, el

diagrama de caso de uso general traza un mapa del flujo de trabajo integral, identificando los actores involucrados, sus roles y las actividades que desempeñan. Esta representación gráfica facilita la comprensión global del sistema y sirve como una guía valiosa para la fase de diseño y desarrollo, asegurando que el proyecto siga las directrices de la ingeniería web de UWE y cumpla con los objetivos definidos de manera efectiva y eficiente.

**Figura 3.1.**  
Caso de uso general

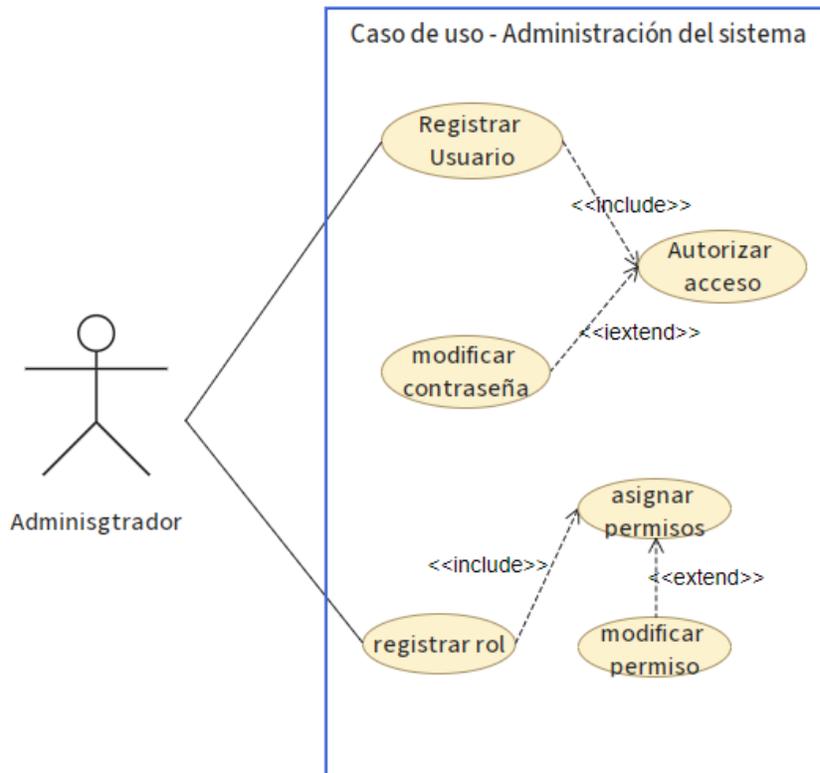


**3.2.2. Caso de uso-administrador**

A través de este diagrama de uso, se delinea el flujo de trabajo específico de los administradores, identificando las acciones clave, los actores involucrados y las funcionalidades que desempeñan. Este diagrama no solo facilita la comprensión profunda de las responsabilidades del rol de administrador, sino que también sirve

como guía esencial para el diseño y desarrollo, garantizando que el sistema cumpla con los estándares de UWE y ofrezca a los administradores las herramientas necesarias para una gestión eficaz y eficiente de la plataforma web.

**Figura 3.2.**  
Caso de uso administrador



**Tabla 3.4.**  
Caso de uso administrador

<b>Caso de uso: Administrador</b>	
<b>Actores:</b>	Administrador
<b>Tipo:</b>	Primario esencial
<b>Descripción:</b>	El administrador tiene el control del sistema. Inicia sesión como super administrador. El administrador puede crear nuevo usuario, cambiar estado, eliminar usuario.

### 3.2.3. Caso de uso\_ Coordinador

El caso de uso "Coordinador del Sistema" representa cómo los coordinadores gestionan la asignación de tareas de traducción, supervisan flujos de trabajo y coordinan recursos. Este diagrama guía el diseño y desarrollo del sistema, asegurando que cumpla con los estándares de UWE y empodere a los coordinadores para una gestión eficaz del sitio. El coordinador del consultorio lingüístico presenta las siguientes funciones:

Figura 3.3.

Caso de uso administrador

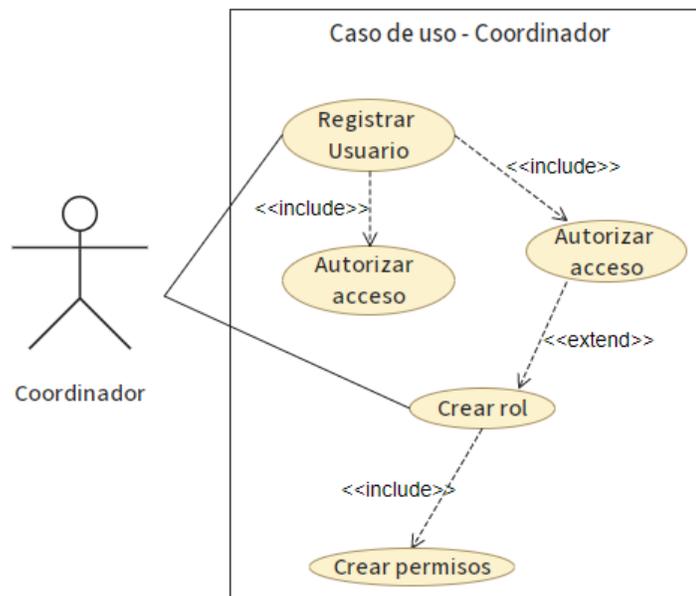


Tabla 3.5.

Caso de uso-coordinador

---

#### Caso de uso: coordinador

---

**Actores:** Coordinador

**Tipo:** Primario esencial

**Descripción:** El coordinador del consultorio lingüístico registra nuevo traductor. Tiene la función de crear roles asignando diferentes permisos de acceso.

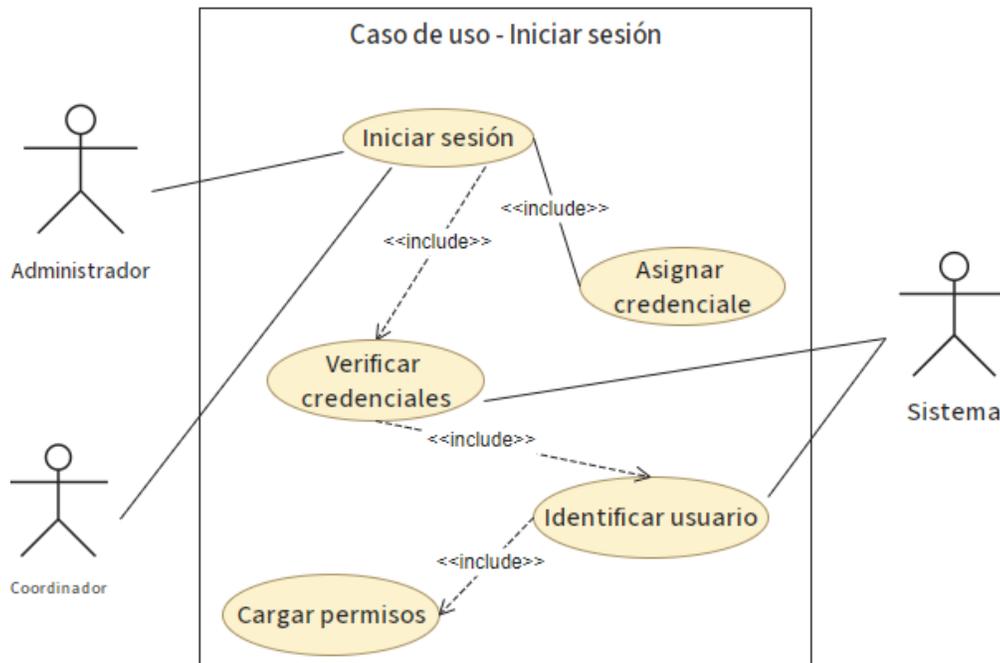
---

### 3.2.4. Caso de uso\_ iniciar sesión

El diagrama de caso de uso "Iniciar Sesión" desempeña un rol central al representar cómo los usuarios acceden a la plataforma. Este diagrama detalla las interacciones, desde la introducción de credenciales hasta la autenticación y acceso a las funcionalidades. Facilita el diseño y desarrollo del sistema, asegurando que el proceso de inicio de sesión cumpla con los estándares de UWE, brindando a los usuarios una experiencia segura y eficaz. Entonces, este caso de uso se plasma los procedimientos que sigue el sistema para verificar las credenciales de un usuario.

**Figura 3.4.**

Caso de uso- iniciar sesión



**Tabla 3.6.**

Caso de uso-iniciar sesión

<b>Caso de uso: iniciar sesión</b>	
<b>Actores:</b>	Administrador, coordinador
<b>Tipo:</b>	Primario esencial
<b>Descripción:</b>	Es el acceso principal al sistema, donde el coordinador y el administrador deben ingresar sus credenciales de acceso.

---

Posteriormente el sistema verificará la información y los permisos de acceso. De otra manera, emitirá un mensaje de error.

---

### 3.2.4. Caso de uso\_ solicitar traducción

En este caso de uso se plasma los procedimientos que sigue el sistema para verificar las credenciales de un usuario.

**Figura 3.5.**  
Caso de uso- solicitar traducción



**Tabla 3.7.**  
Caso de uso- solicitar traducción

---

#### Caso de uso: solicitar traducción

---

<b>Actores:</b>	Administrador, coordinador, cliente
<b>Tipo:</b>	Primario esencial
<b>Descripción:</b>	El cliente inicia sesión y posteriormente realiza la solicitud de traducción profesional. Para ello el cliente seleccionará el idioma y el traductor. Luego enviará el documento a traducir. El cliente podrá visualizar el estado de su solicitud.

---

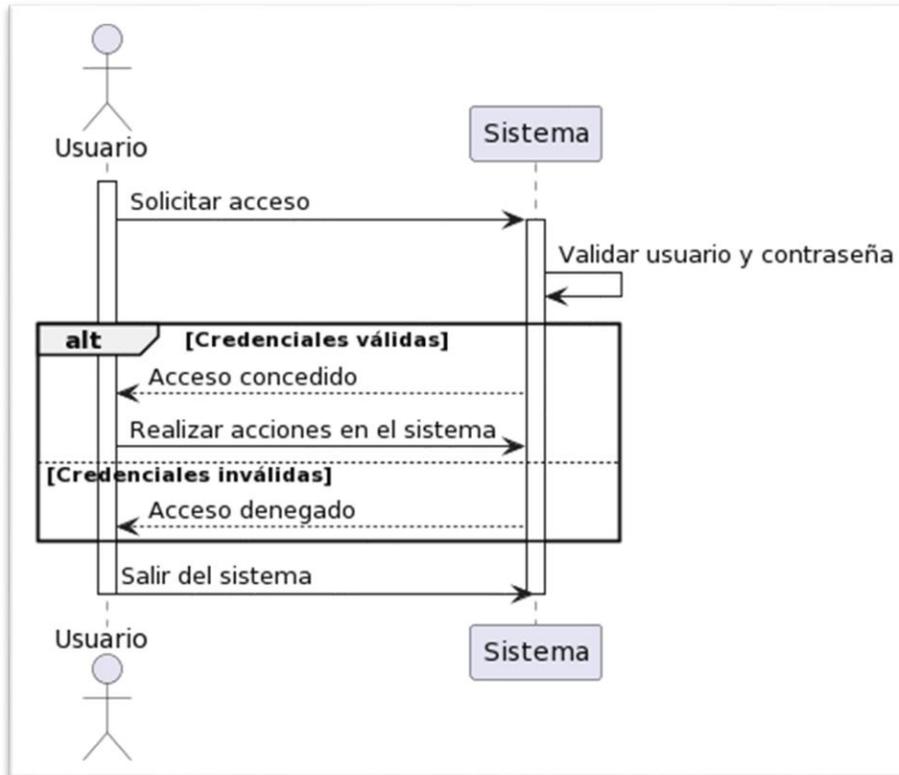
### **3.3. Diagrama de secuencia**

Los diagramas de secuencia ilustran cómo el usuario se comunica con el sistema, detallando de manera progresiva el procedimiento llevado a cabo en cada escenario de uso. Así, el diagrama de secuencia bajo la metodología UWE en este proyecto de sitio web de servicios de traducción es esencial para visualizar el flujo detallado de las operaciones, desde la solicitud hasta la entrega de la traducción. También facilita una comprensión clara de los procesos, permitiendo una planificación efectiva y la identificación de oportunidades de mejora. Además, este diagrama es una herramienta valiosa para detectar problemas y optimizar la eficiencia del sistema, garantizando que el sitio web ofrezca servicios de traducción eficaces y de alta calidad.

#### **3.3.1. Ingresar al Sistema (Usuario)**

El diagrama de secuencia de "Ingreso al Sistema" del usuario en el marco de la metodología UWE representa de manera precisa y visual el flujo de acciones que un usuario sigue al iniciar sesión en el sistema de servicios de traducción. Este diagrama establece la interacción desde la introducción de credenciales hasta el acceso a las funcionalidades, lo que permite diseñar un proceso de inicio de sesión eficiente y seguro, cumpliendo con los estándares de UWE y brindando a los usuarios una experiencia satisfactoria y confiable.

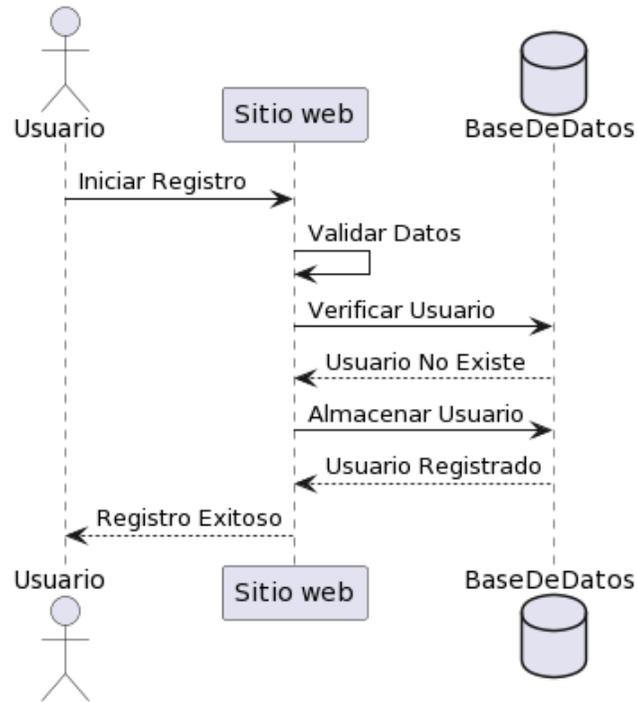
**Figura 3.6**  
Diagrama de Secuencia de Autenticación de Usuario



### 3.3.2. Registrarse en el Sistema (Usuario)

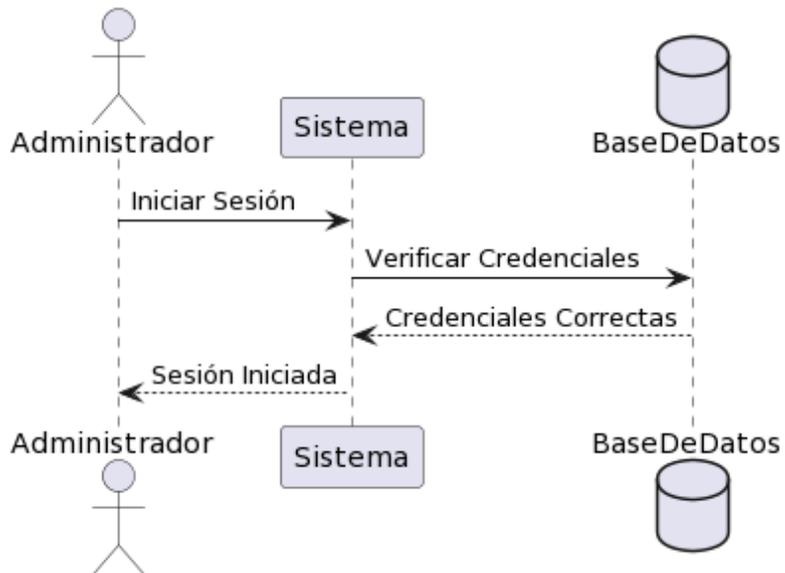
El diagrama de secuencia "Registrarse en el Sistema" por parte usuario ofrece una representación visual esencial del flujo de acciones que un usuario sigue al registrarse en el sistema de servicios de traducción. Este diagrama permite diseñar un proceso de registro claro y eficiente, garantizando que se cumplan los estándares de UWE para una experiencia de usuario satisfactoria y confiable desde el primer contacto con la plataforma.

**Figura. 3.7.**  
Diagrama de Secuencia registro de usuario



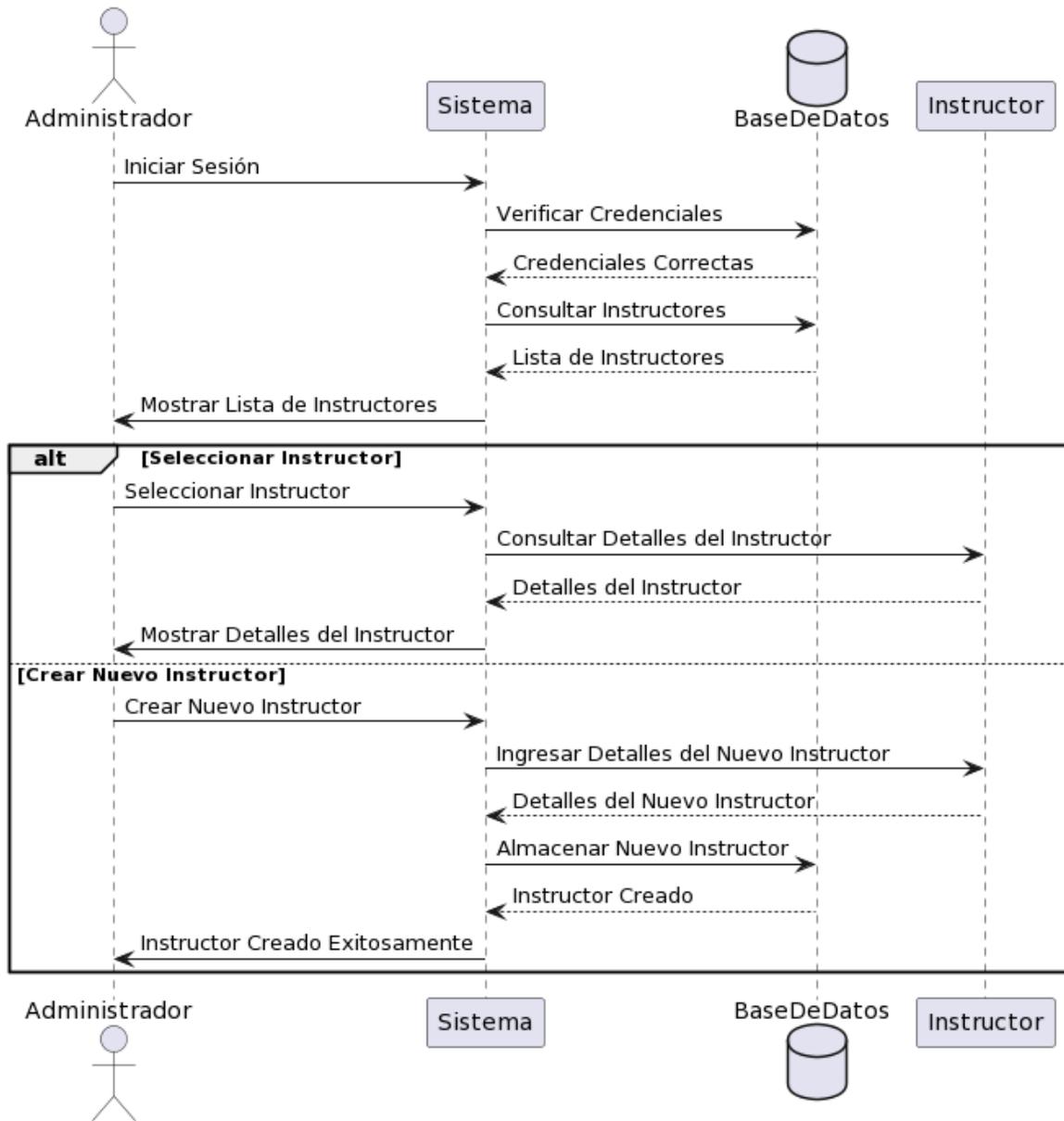
### 3.3.3. Ingreso del Sistema (Administrador)

**Figura. 3.8.**  
Diagrama de Secuencia



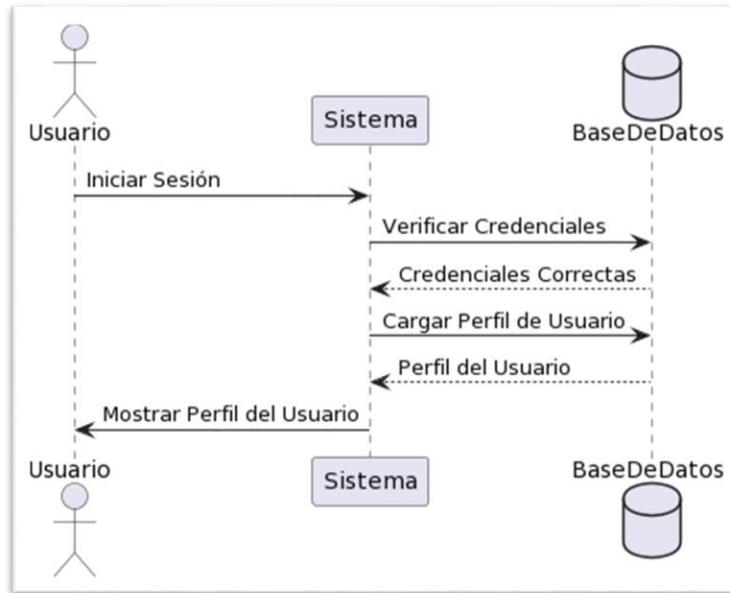
### 3.3.4. Registrarse en el Sistema (Usuario)

Figura. 3.9.  
Diagrama de Secuencia



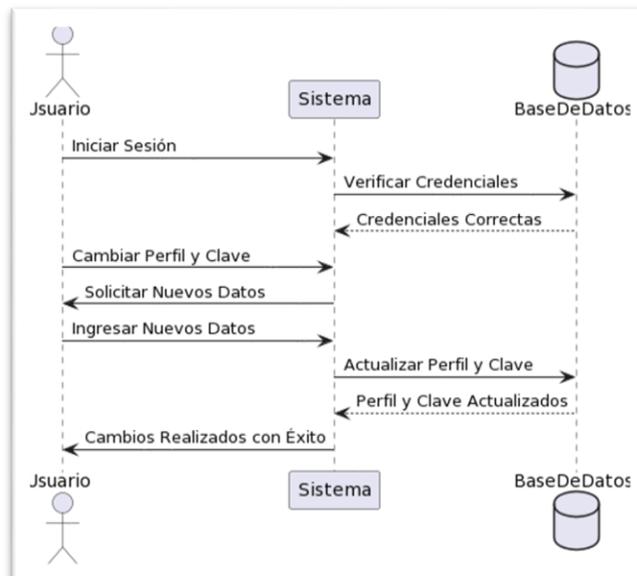
### 3.3.5. Mostrar perfil

Figura. 3.10.  
Diagrama de Secuencia mostrar perfil



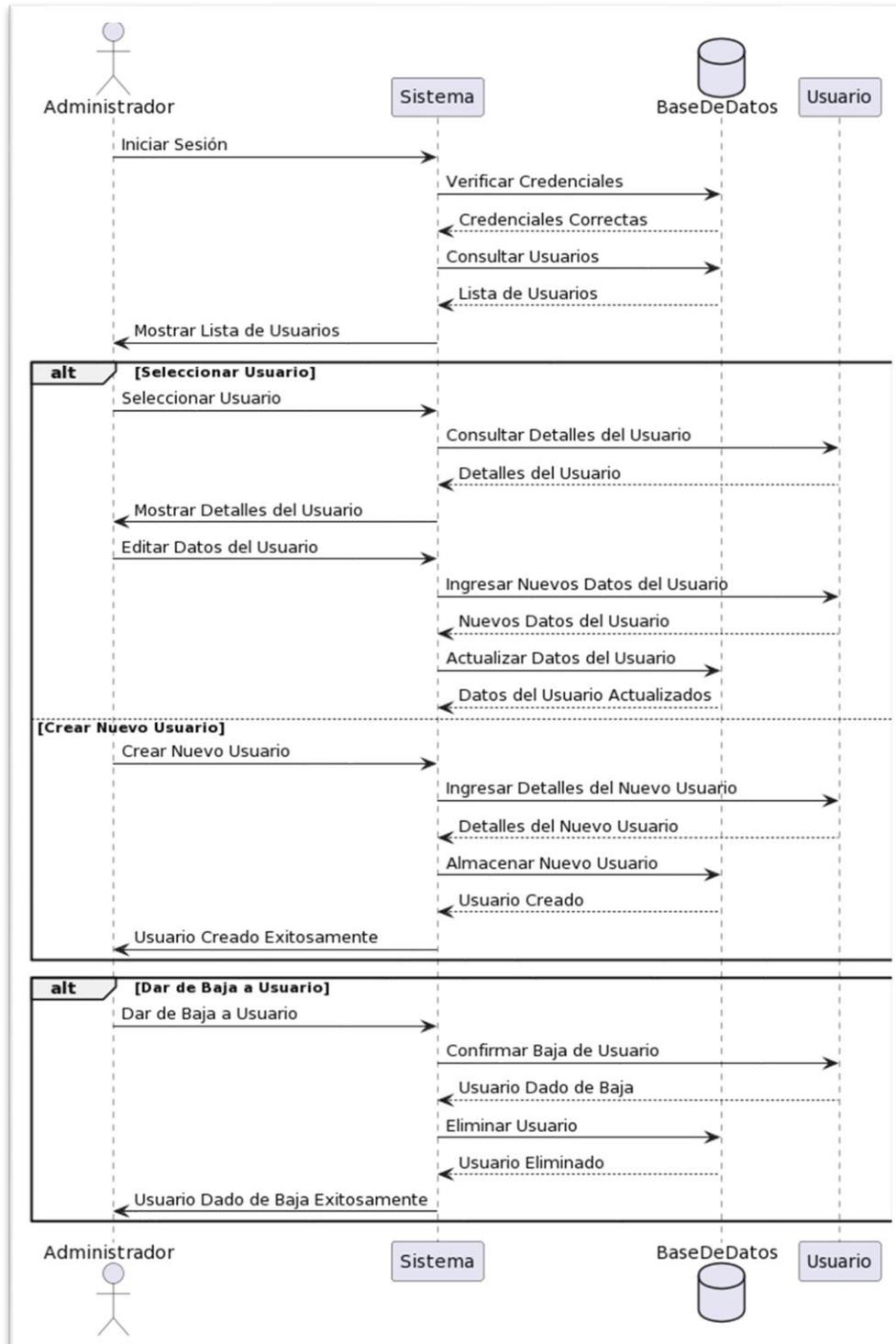
### 3.3.6. Cambiar Perfil y Clave de Usuario

Figura. 3.11.  
Diagrama de Secuencia cambiar perfil y clave de usuario



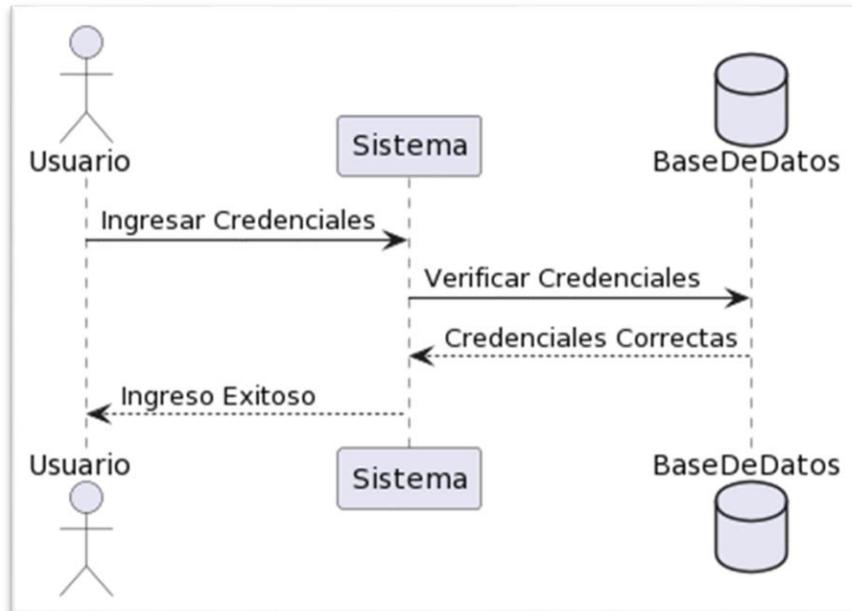
### 3.3.7. Administrar usuario operador

Figura. 3.12.  
Diagrama de Secuencia administrar usuarios



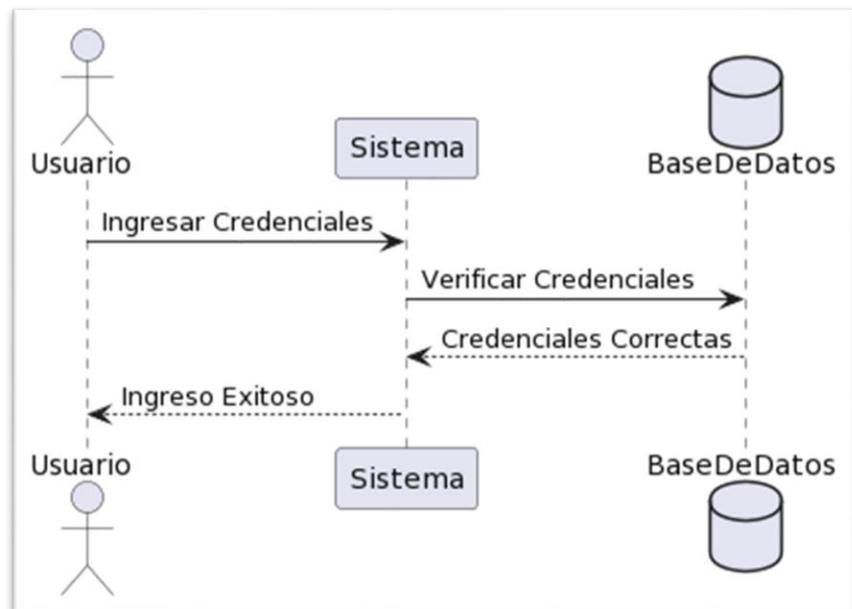
### 3.3.8. Ingresar al Sistema

Figura. 3.13.  
Diagrama de Secuencia ingresar al sistema



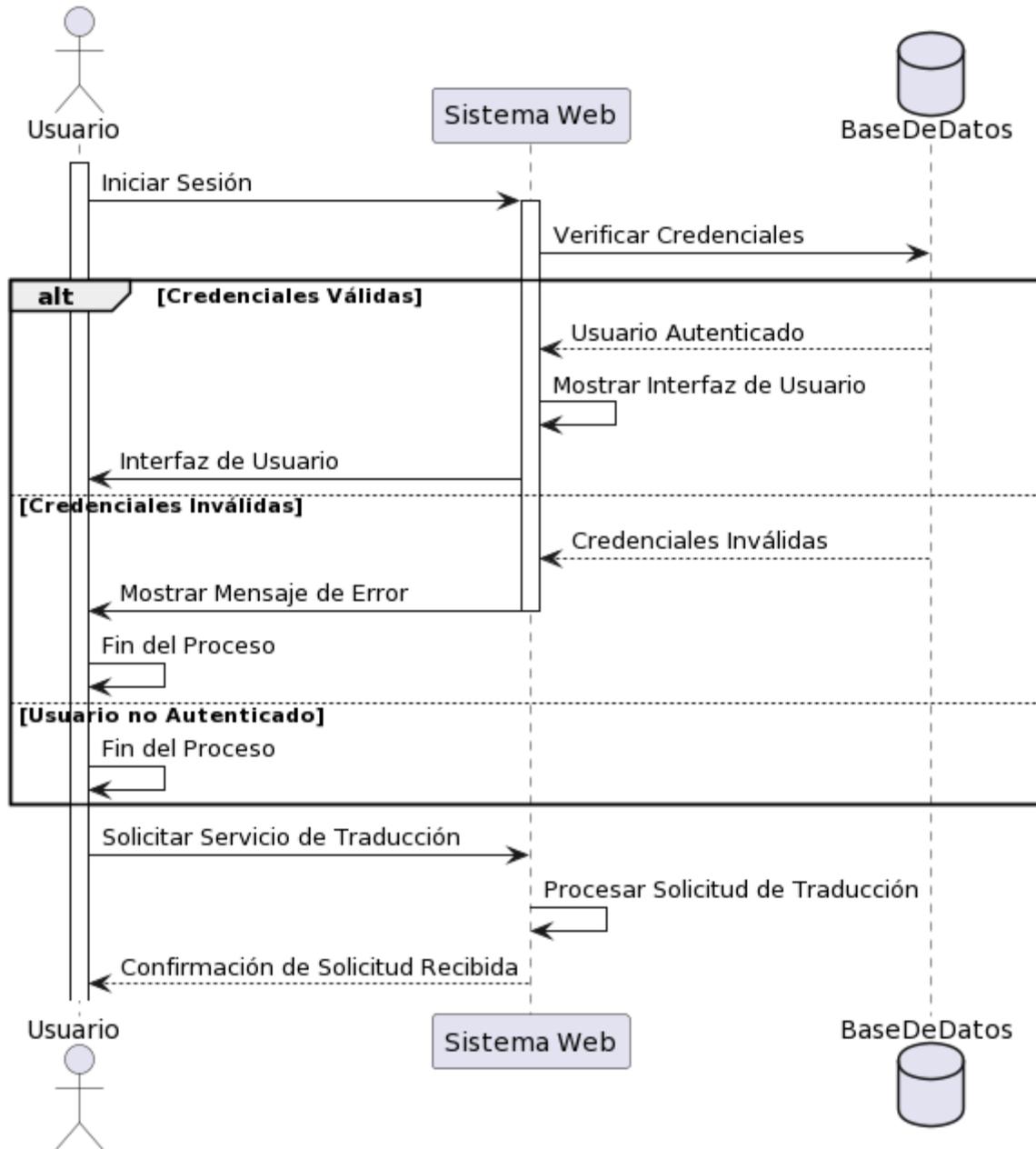
### 3.3.8. Ingresar al Sistema

Figura. 3.14.  
Diagrama de Secuencia ingresar al sistema



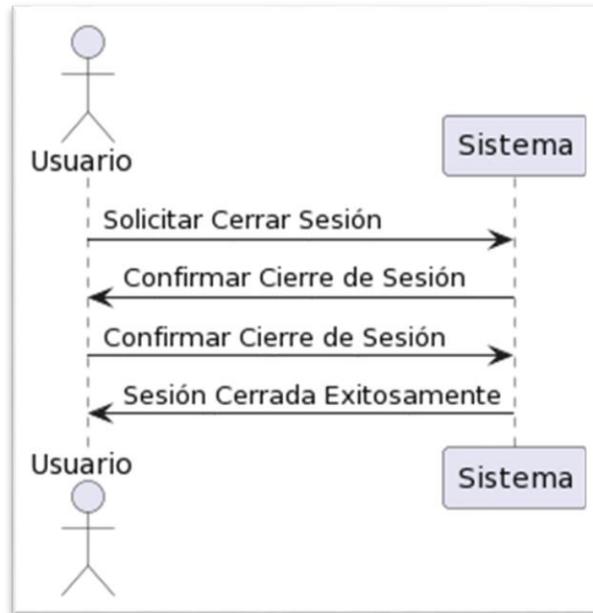
### 3.3.9. Solicitar servicio de traducción

Figura. 3.15.  
Solicitar servicio de traducción



### 3.3.10. Cerrar sesión

**Figura. 3.16.**  
Diagrama de Secuencia cerrar sesión

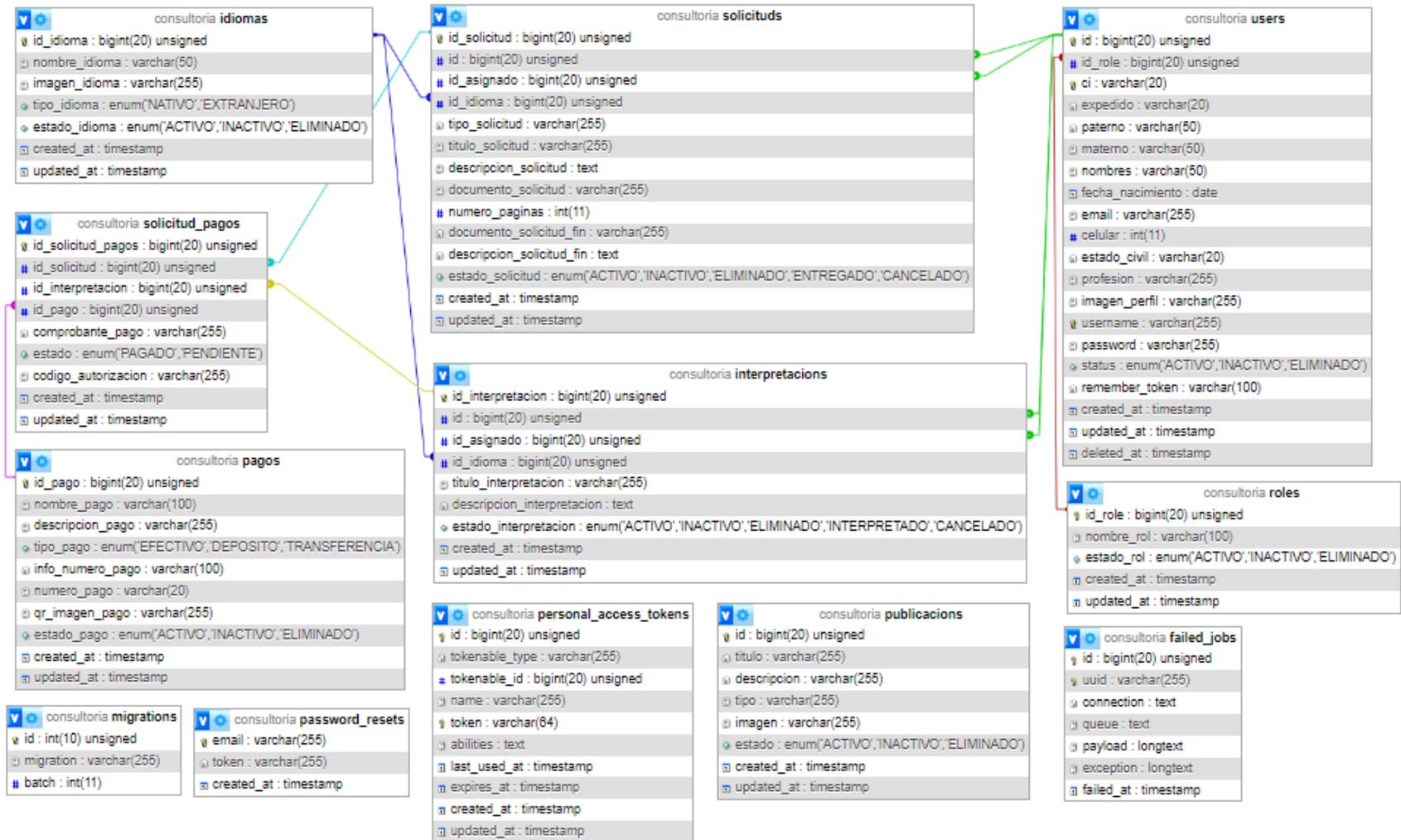


### 3.3. Diagrama de clases

Para el proyecto de un sistema web de servicios de traducción, se desarrolla un modelo conceptual o de clases. Este modelo se utiliza para definir la estructura y las relaciones entre los diferentes componentes clave de la aplicación. En este contexto, se representan entidades como "Usuarios", "Traductores" y "Proyectos de Traducción", asignándoles atributos y métodos que reflejan sus características y funcionalidades específicas. Además, se establecen relaciones entre estas entidades para modelar cómo interactúan en el sistema, como la participación de los usuarios en proyectos de traducción o la asignación de traductores a proyectos. Este modelo sirve como base sólida para el desarrollo y la comprensión del sistema de traducción web, permitiendo diseñar y estructurar eficazmente la aplicación.

Figura. 3.17.

Diagrama de clases

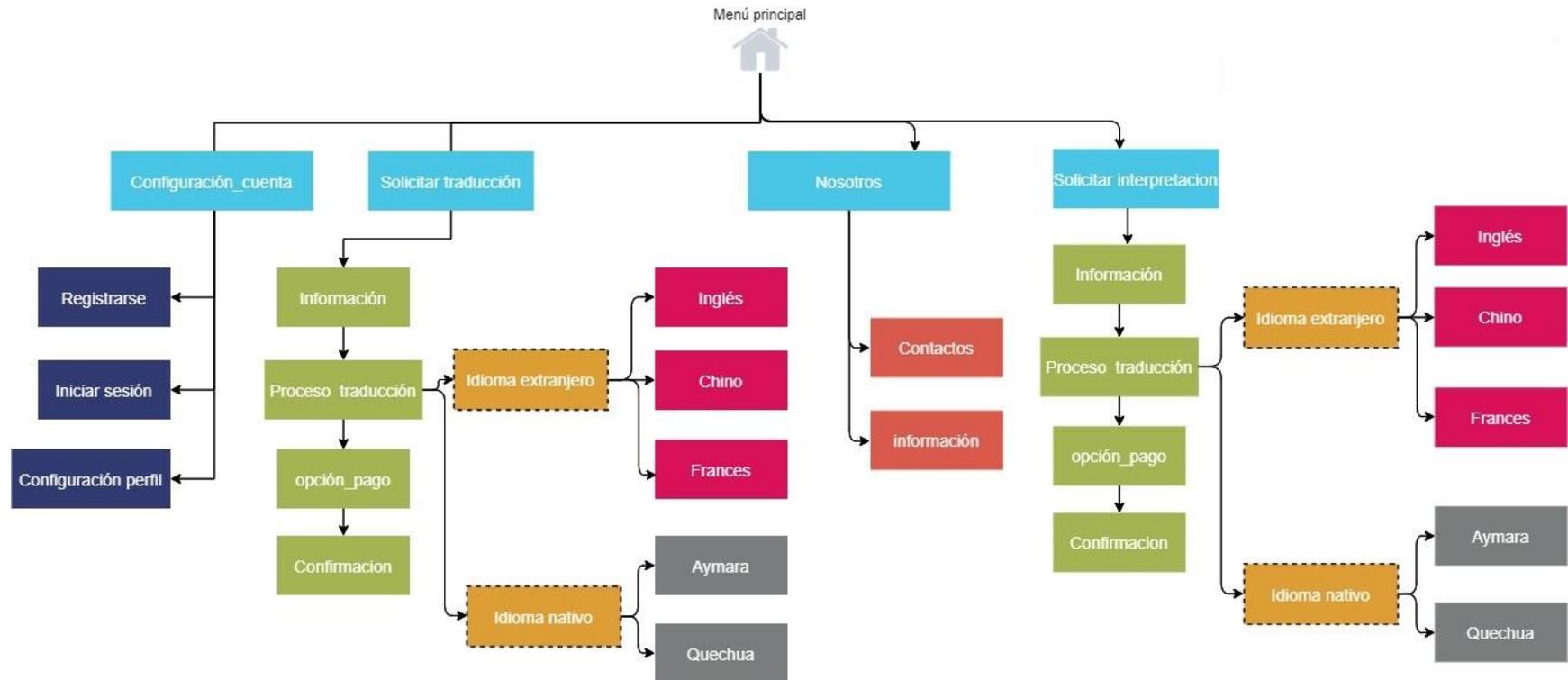


### 3.4. Modelo de navegación

El diseño de navegación debe mostrar conexiones lógicas y de navegación entre lecciones. A continuación, la Figura 3.14 muestra un modelo de navegación de alto nivel de enlaces lógicos de administración de usuarios que le permiten navegar por el sistema WEB.

Figura. 3.18

Diagrama de navegación



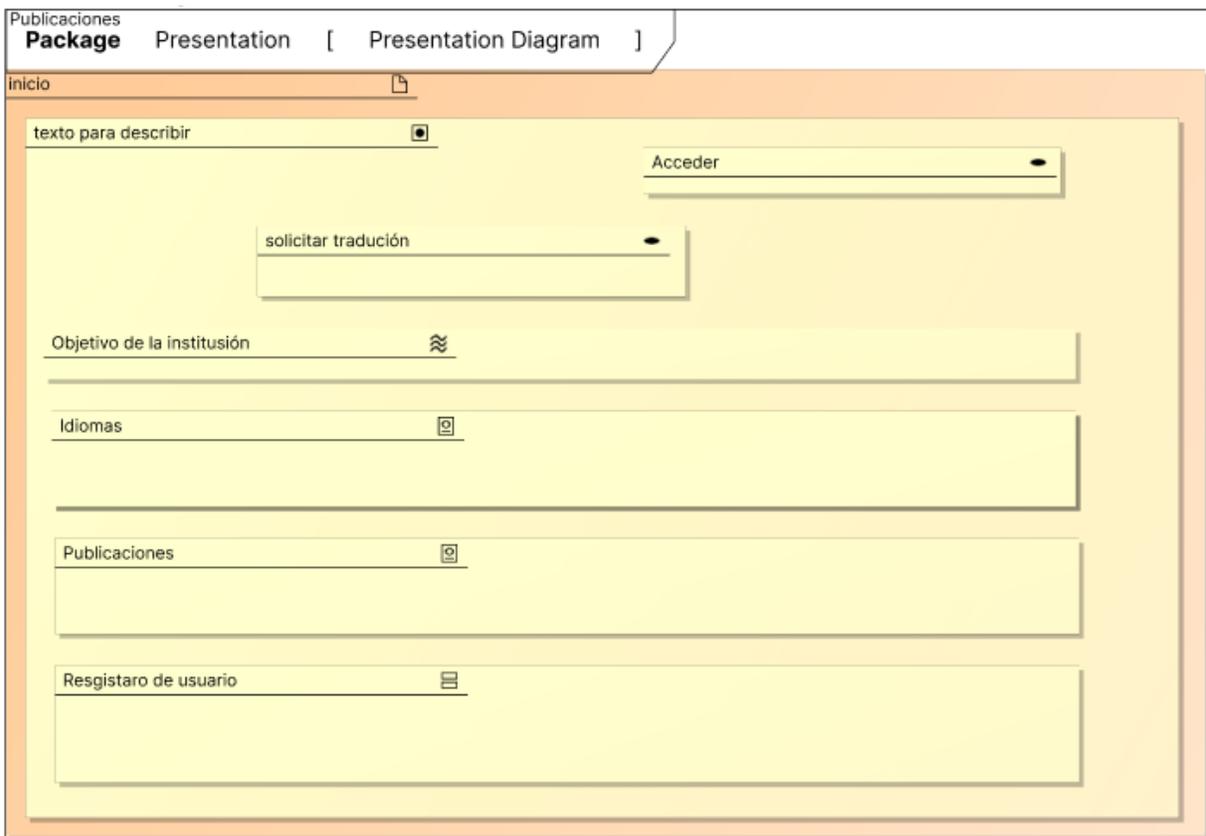
### 3.5. Modelo de presentación

El modelo de presentación del sitio web de servicios de traducción, en línea con la metodología UWE, se enfoca en definir la interfaz de usuario, la estructura y la organización de la información para garantizar una experiencia de usuario efectiva y coherente. Siguiendo los principios de UWE, este enfoque de diseño permite adaptar la presentación a diferentes dispositivos y plataformas, asegurando que el sitio web sea accesible en cualquier momento y lugar, cumpliendo con los estándares de calidad y usabilidad.

#### 3.5.1. Modelo de presentación (Inicio)

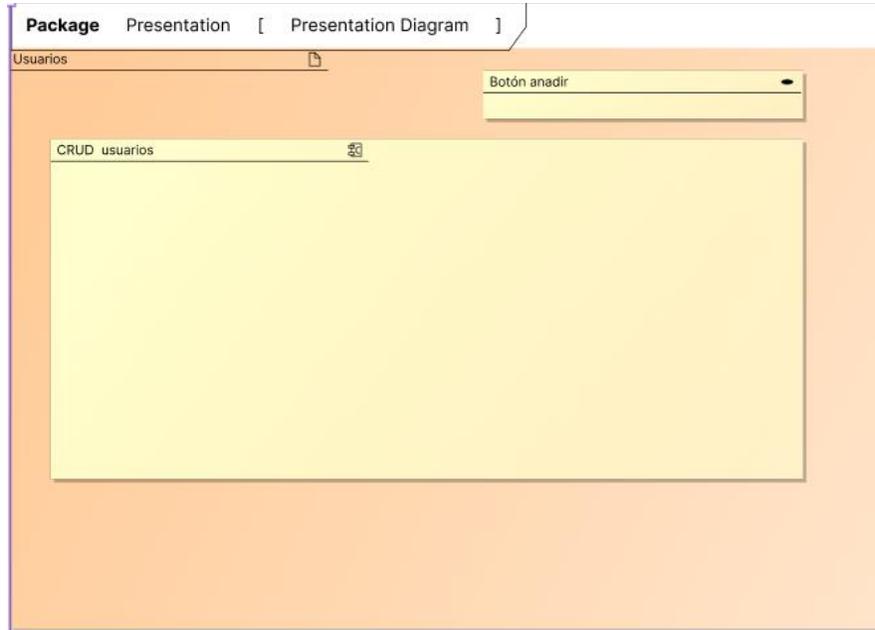
**Figura. 3.19**

Diagrama de presentación - inicio



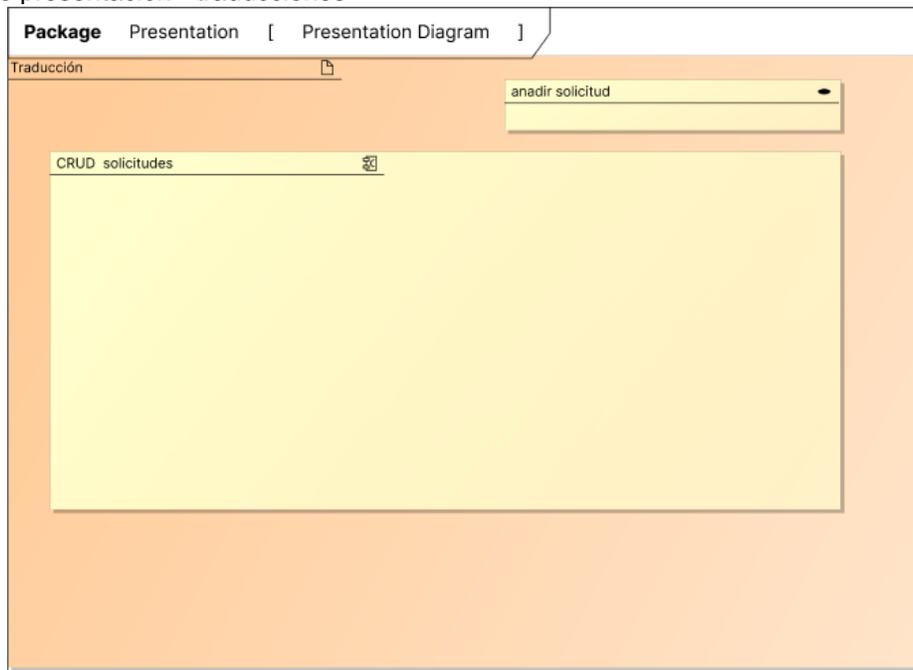
### 3.5.2. Modelo de presentación (Administración de usuarios)

**Figura. 3.20**  
Diagrama de presentación administración de usuarios



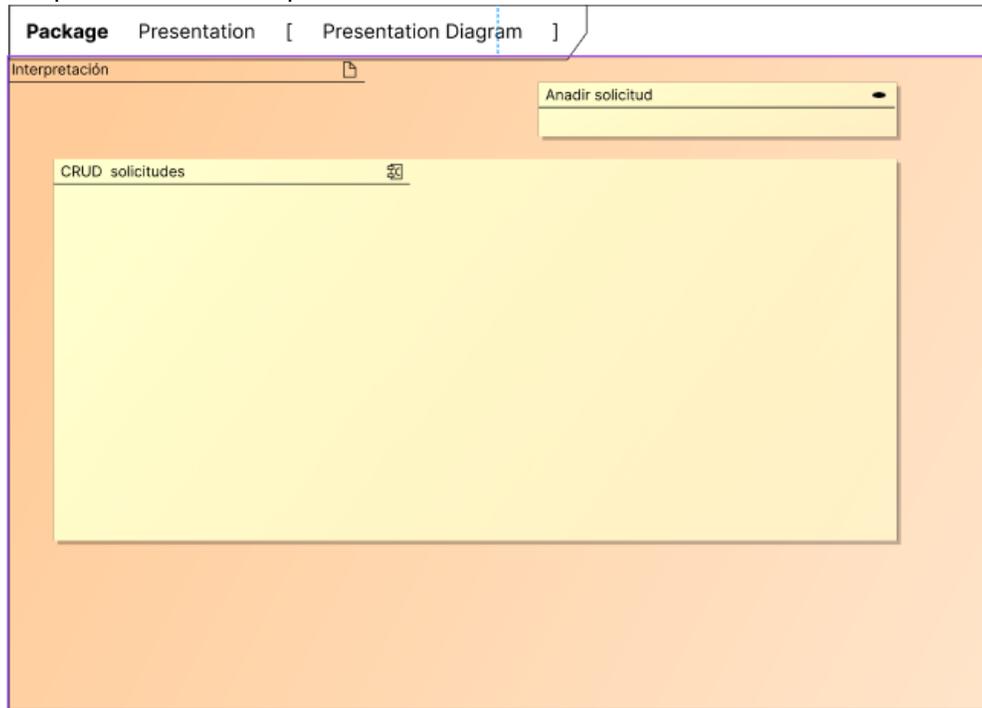
### 3.5.3. Modelo de presentación (Traducciones)

**Figura. 3.21**  
Diagrama de presentación - traducciones



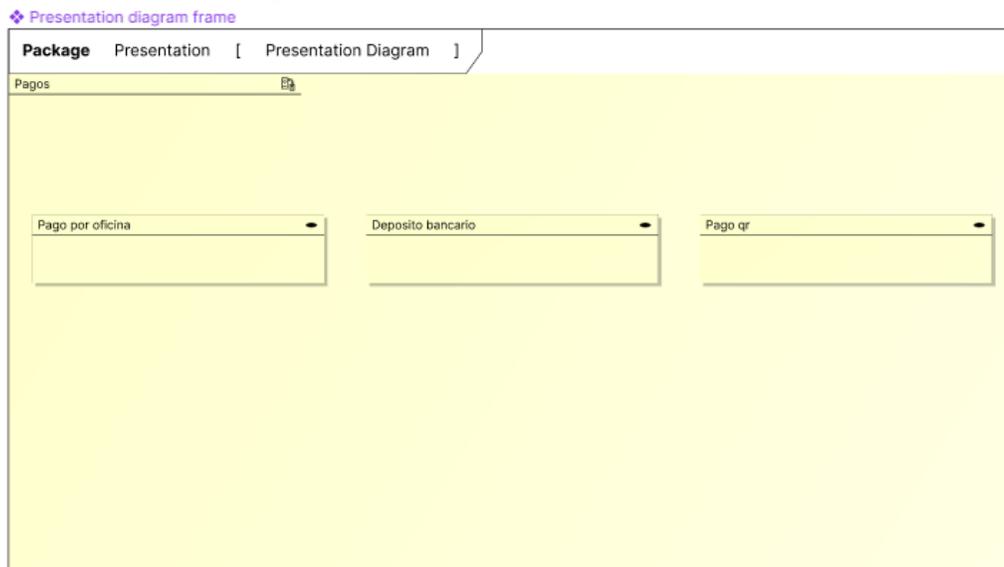
### 3.5.4. Modelo de presentación (Interpretación)

**Figura. 3.22**  
Diagrama de presentación – interpretación



### 3.5.5. Modelo de presentación (Pagos)

**Figura. 3.23**  
Diagrama de presentación - pagos



### 3.6. Codificación de software

La codificación de software de "Usuarios\_blade" en el sistema web de "Servicios de Traducción" se centra en la creación y gestión de las vistas que permiten la interacción de los usuarios con la plataforma. Este componente juega un papel crucial en la autenticación, registro de cuentas, edición de perfiles y otras funciones relacionadas con la identidad y las preferencias de los usuarios. El código de "Usuarios\_blade" está diseñado para proporcionar una interfaz de usuario intuitiva y segura, garantizando que los usuarios puedan acceder y gestionar sus cuentas de manera eficiente, contribuyendo así a una experiencia de usuario satisfactoria en el sistema de Servicios de Traducción.

**Figura. 3.24**  
Usuarios.blade.php

```
@extends('wrapper')
@section('contenido-dashboard')
    <div class="page-content-wrapper">
        <div class="page-content">
            <div class="page-breadcrumb d-none d-sm-flex align-items-center mb-3">
                <div class="breadcrumb-title pe-3">Usuarios</div>
                <div class="ps-3">
                    <nav aria-label="breadcrumb">
                        <ol class="breadcrumb mb-0 p-0 align-items-center">
                            <li class="breadcrumb-item"><a href="{{ route('admin') }}"><ion-icon
                                name="home-outline"></ion-icon></a>
                            </li>
                            <li class="breadcrumb-item active" aria-current="page">Usuarios</li>
                        </ol>
                    </nav>
                </div>
                <div class="ms-auto">
                    <div class="btn-group">
                        <a href="{{ route('nuevoUsuario') }}" class="btn btn-outline-primary">Nuevo usuario</a>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
```

```

@include('mensajes')

<div class="card">
  <div class="card-body">
    <div class="d-flex align-items-center">
      <h5 class="mb-0">Usuarios habilitados</h5>
      {{-- <form class="ms-auto position-relative">
        <div class="position-absolute top-50 translate-
middle-y search-icon px-3"><ion-icon
        name="search-sharp"></ion-icon></div>
        <input class="form-control ps-5" type="text"
placeholder="search">
      </form> --}}
    </div>
    <div class="table-responsive mt-3">
      <table class="table align-middle">
        <thead class="table-secondary">
          <tr>
            <th>#</th>
            <th>Nombres</th>
            <th>Apellido Paterno</th>
            <th>Apellido Materno</th>
            <th>CI</th>
            <th>Expedido</th>
            <th>Email</th>
            <th>Rol</th>
            <th>Rol</th>
            <th>Acciones</th>
          </tr>
        </thead>
        <tbody>
          @foreach ($usuarios as $usuario)
            <tr>
              <td>{{ $usuario->id }}</td>
              {{-- <td>
                <div class="d-flex align-items-center
gap-3 cursor-pointer">
                  
                  <div class="">
                    <p class="mb-0"></p>
                  </div>
                </td>
          }}
          </tr>
        </tbody>
      </table>
    </div>
  </div>
</div>

```

```

</td> --}}
        <td>{{ $usuario->nombres }}</td>
        <td>{{ $usuario->paterno }}</td>
        <td>{{ $usuario->materno }}</td>
        <td>{{ $usuario->ci }}</td>
        <td>{{ $usuario->expedido }}</td>
        <td>{{ $usuario->email }}</td>
        <td>{{ $usuario->nombre_rol }}</td>
        <td>
            @if ($usuario->id == 1)
                <span class="badge bg-success
border-0">{{ $usuario->status }}</span>
            @else
                @if ($usuario->status == 'ACTIVO')
                    <form action="{{
route('post_status', $usuario->id) }}" method="POST">
                        @csrf
                        <input type="hidden"
value="{{ $usuario->status }}"
                        name="status">
                        <button class="badge bg-
success border-0"
type="submit">{{
$usuario->status }}</button>
                    </form>
                @elseif ($usuario->status ==
'INACTIVO')
                    <form action="{{
route('post_status', $usuario->id) }}" method="POST">
                        @csrf
                        <input type="hidden"
value="{{ $usuario->status }}"
                        name="status">
                        <button class="badge bg-
danger border-0"
type="submit">{{
$usuario->status }}</button>
                    </form>
                @endif
            @endif
        </td>
        <td>
            @if ($usuario->id != 1)
                <a href="{{ route('editarUsuario',
$usuario->id) }}" class="btn btn-sm btn-warning"

```

```

data-bs-toggle="tooltip"
data-bs-placement="bottom"
bs-original-title="Edit info"
aria-label="Edit">
  <ion-icon name="pencil-
outline"></ion-icon>
  Editar
</a>
  <form action="{{
route('eliminarUsuario', $usuario->id) }}" method="POST"
style="display: inline">
  @csrf
  @method('DELETE')
  <button type="submit"
class="btn btn-sm btn-danger"
data-bs-toggle="tooltip"
data-bs-placement="bottom"
data-bs-original-title="Delete"
aria-label="Delete">
  <ion-icon name="trash-
outline"></ion-icon>
  Eliminar
</button>
</form>
@endif
</td>
</tr>
@endforeach
</tbody>
</table>
</div>
</div>
</div>
</div>
</div>
</div>
@include('components.footer')
@endsection

```

### 3.6. Implementación del sistema

En la captura de pantalla que sigue, se puede observar el formulario de inicio, que proporciona acceso al menú de inicio y al de inicio de sesión, así como un espacio que ofrece información sobre la descripción de la institución.

**Figura. 3.25**  
Barra de navegación y portada



**Figura. 3.26**  
Página principal



**Figura. 3.27**  
Idiomas



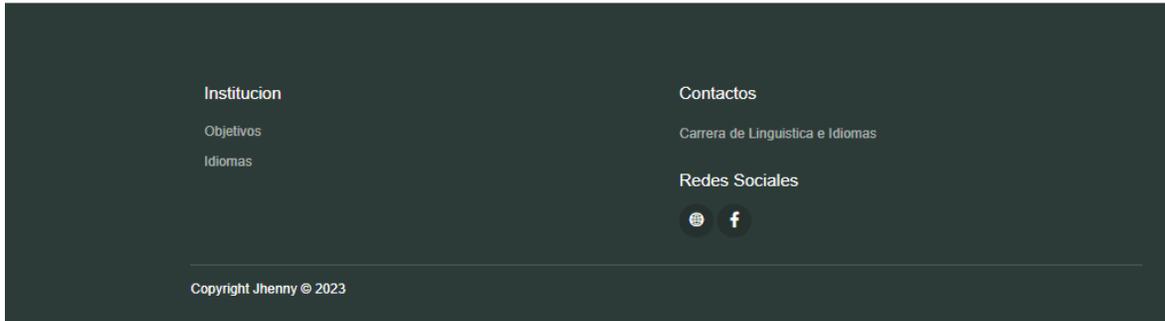
**Figura. 3.28**  
Publicaciones



**Figura. 3.29**  
Registro de Usuario

Cedula de Identidad:	Expedido:
<input type="text" value="CI"/>	<input type="text" value="EXPEDIDO"/>
Nombres:	Apellido paterno:
<input type="text" value="Nombres"/>	<input type="text" value="Apellido Paterno"/>
Apellido materno:	Fecha de nacimiento:
<input type="text" value="Apellido Materno"/>	<input type="text" value="mm/dd/yyyy"/>
Estado civil:	Direccion Email:
<input type="text" value="ESTADO CIVIL"/>	<input type="text" value="Email"/>
Numero de celular:	Profesión:
<input type="text" value="Numero de celular"/>	<input type="text" value="Profesion"/>
Contraseña:	Confirmar contraseña:
<input type="text" value="Contraseña"/>	<input type="text" value="Confirmar contraseña"/>
<input type="button" value="Registrarse"/>	

**Figura. 3.30**  
Footer



### 3.6.1. Iniciar sesión

Para acceder al sistema web en calidad de administrador, se debe seguir un procedimiento específico. Este proceso implica utilizar credenciales de administrador o una función de inicio de sesión diseñada para usuarios con privilegios administrativos.

**Figura. 3.31**  
Iniciar sesión

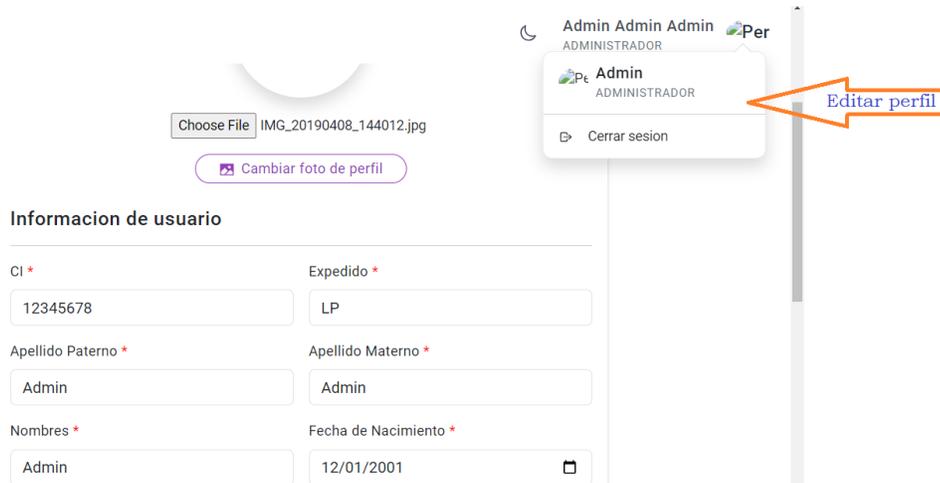


### 3.6.2. Editar perfil \_ administrador

La función de "Editar Perfil" en el sistema web del Centro de Traducción es una característica esencial que permite a los usuarios modificar y actualizar su información

personal. Este proceso, parte integral de la experiencia del usuario, garantiza que los perfiles se mantengan precisos y personalizados, optimizando así la interacción y colaboración dentro del centro de traducción.

**Figura. 3.32**  
Editar perfil



### 3.6.3. Cerrar sesión \_ administrador

La función de "Cerrar Sesión" en el sitio web del Centro de Traducción es un componente fundamental que permite a los usuarios finalizar su sesión de manera segura y proteger su información personal. Al utilizar esta función, los usuarios pueden garantizar que nadie más tenga acceso a su cuenta y salvaguardar la privacidad de sus datos.

**Figura. 3.33**  
Cerrar sesión



### 3.6.4. Administración de usuarios

La "Administración de Usuarios" en el sitio web del Centro de Traducción es una función crítica que permite a los administradores y moderadores supervisar y gestionar las cuentas de los usuarios de manera eficiente. Esto abarca la creación de nuevas cuentas, la asignación de roles y privilegios, así como la suspensión o eliminación de cuentas en caso necesario.

**Figura. 3.34**  
Administración de usuarios

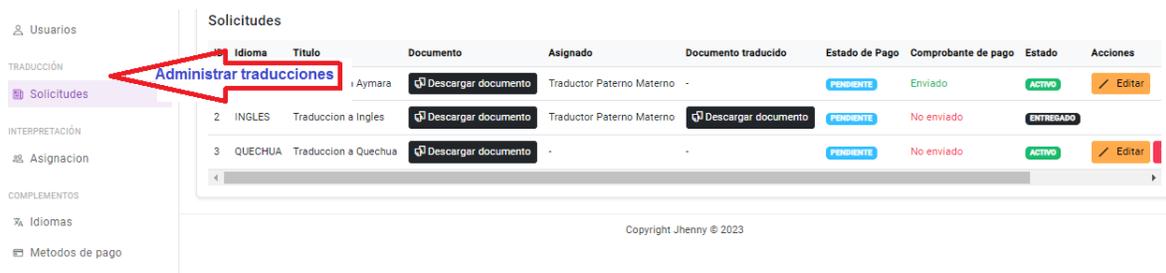


			Apellido Materno	CI	Expedido
1	Admin	Admin	Admin	12345678	LP
2	Coordinador	Paterno	Materno	11111111	LP
3	Traductor	Paterno	Materno	22222222	LP

### 3.6.5. Traducciones

La "Administración de Solicitudes de Traducción" en el sitio web del Centro de Traducción desempeña un papel crucial en el proceso de gestión de proyectos de traducción. Esta función permite a los usuarios con privilegios de coordinador y administración supervisar, revisar y coordinar las solicitudes de traducción presentadas por los usuarios de la comunidad. Dentro de este módulo, los encargados pueden evaluar la calidad y pertinencia de las solicitudes, asignar traductores, realizar un seguimiento del progreso y asegurarse de que se cumplan los plazos establecidos.

**Figura. 3.35**  
Administrar traducciones



### 3.6.6. Interpretaciones

Esta función permite a los usuarios con permisos de coordinador y administración supervisar y gestionar las solicitudes de interpretación presentadas por los usuarios de la comunidad. Los responsables de esta tarea pueden evaluar la disponibilidad de traductores, programar eventos o sesiones de interpretación, y asegurarse de que las solicitudes se atiendan de manera efectiva y eficiente.

**Figura. 3.36**  
Administrar interpretaciones

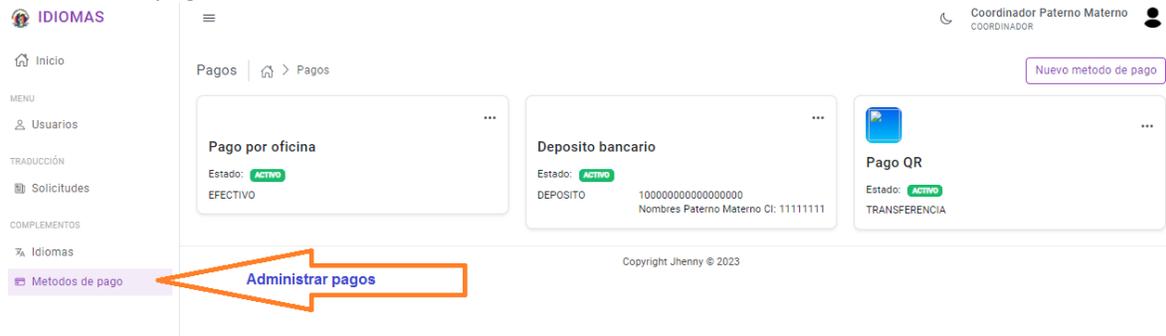


### 3.6.7. Pagos

La "Administración de Pagos de Servicios" en el sitio web del Centro de Traducción constituye una función crítica para asegurar una gestión financiera eficiente. Este módulo permite a los responsables y administradores realizar un seguimiento detallado de los pagos asociados a los servicios de traducción proporcionados por el centro. Los

usuarios pueden registrar, verificar y gestionar los pagos de los clientes y realizar un seguimiento de los saldos solicitudes.

**Figura. 3.37**  
Administrar pagos



### 3.7. Pruebas de software

En el ámbito de la ingeniería de software, las pruebas desempeñan un papel crucial en la garantía de la calidad, y entre las estrategias más comunes se encuentran las pruebas de caja negra, caja blanca y la prueba de estrés. La prueba de caja negra evalúa la funcionalidad externa del sistema sin conocer su estructura interna, centrándose en las entradas y salidas. Por otro lado, la prueba de caja blanca examina la lógica interna del código fuente, revelando posibles vulnerabilidades y evaluando la cobertura del código. En contraste, la prueba de estrés evalúa la capacidad del sistema bajo condiciones extremas. Estas metodologías combinadas permiten una evaluación exhaustiva, asegurando la robustez y la eficiencia del software.

#### 3.7.1. Pruebas de Caja Blanca

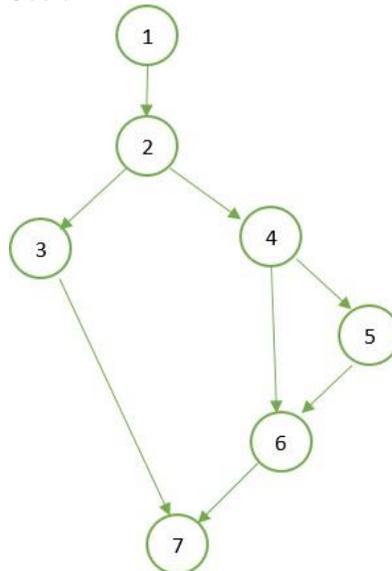
En el marco del desarrollo de nuestro proyecto de Sistema de Servicios de Traducción, la aplicación de pruebas de caja blanca emerge como una estrategia esencial. Al examinar de manera detallada la estructura interna del código fuente, estas pruebas no solo permitirán identificar posibles errores y asegurar la fiabilidad del sistema, sino que también facilitarán la evaluación de la cobertura del código. Dada la naturaleza crítica de la precisión en los servicios de traducción, la implementación de pruebas de caja blanca se presenta como una medida fundamental para garantizar la calidad y la

seguridad del software, contribuyendo así a la entrega de un sistema robusto y confiable.

### 3.7.1.1. Pruebas de Caja Blanca \_Inicio sesión

**Figura. 3.38**

Prueba de Caja Blanca – Inicio de Sesión



Donde:

Inicio de Sesión del sistema (Inicio Flujo) (1)

Verificación de Ingreso de datos (2)

Carga de mensajes de errores (3)

Verificación de credenciales (4)

Carga de permisos de acceso al sistema (5)

Carga de mensajes (6)

Fin del ciclo (7)

Analizando el grafo se obtiene lo siguiente:

Reemplazando se obtiene:

$$V(G) = A - N + 2$$

$$V(G) = 8 - 7 + 2 = 3$$

$$V(G) = 3$$

Luego se determina el conjunto de caminos independientes. Los caminos deben

ser probados dadas ciertas variables.

**Tabla 3.8.**  
Evaluación de flujo

<b>N<sup>o</sup></b>	<b>Ruta</b>	<b>Parámetro de entrada</b>
1	1, 2, 3, 7	Se verifica la información ingresada. Se solicita que ambos campos sean introducidos. En caso que existiera un error se lanza el mensaje de alerta.
2	1, 2, 4, 6, 7	Se verifica que ambos campos sean ingresados, y posteriormente se verifica las credenciales. En caso que exista un error se carga los mensajes correspondientes del error.
3	1, 2, 4, 5, 6, 7	Se verifica que ambos campos sean ingresados, posteriormente se verifican las credenciales. En caso de tener éxito, se identifica el tipo de usuario para posteriormente cargar los permisos asignados y permitir el ingreso al sistema.

### **3.7.2. Pruebas de Caja Negra**

En el contexto de nuestro proyecto de Sistema de Servicios de Traducción, la aplicación de pruebas de caja negra se revela como un componente crucial en la evaluación de la funcionalidad externa del sistema. Estas pruebas permitirán identificar posibles fallos en la lógica de procesamiento y asegurar que el sistema responda de manera eficiente a diversas entradas. Dada la importancia de la precisión y la coherencia en los servicios de traducción, la implementación de pruebas de caja negra se presenta como una medida esencial para garantizar la calidad y la efectividad del sistema, contribuyendo así a ofrecer una experiencia de usuario fluida y fiable.

#### **3.7.2.1. Prueba de caja negra - Inicio de Sesión**

La Figura 55, muestra el formulario de inicio de sesión, donde se aplicará la prueba de la caja negra, ingresando diferentes valores se registrará el comportamiento y se

verificará si el formulario cumple con la función programada en ella.

**Figura. 3.39**  
Prueba de Caja Negra – Inicio de Sesión

The image shows a login form with the following elements:

- Title: **Iniciar sesión**
- Subtitle: *Accede a tu cuenta*
- Field: **Nombre de usuario** (text input)
- Field: **Contraseña** (password input)
- Button: **Iniciar sesión** (purple)
- Link: [Registrarse](#) (purple)
- Link: [Volver al inicio](#) (purple)

**Tabla 3.9.**  
Valores límite (Inicio de sesión)

<b>Campo</b>	<b>Entradas validas</b>	<b>Entradas invalidas</b>
Nombre de usuario	Cadena de texto	No se permiten caracteres especiales como, por ejemplo: espacios en blanco
Contraseña o clave	Cadena de texto alfanumérico	No se permiten caracteres como espacio

**Tabla 3.10**  
Prueba de caja negra (Inicio de sesión)

<b>Entrada de datos</b>			
<b>Usuario</b>	<b>Contraseña</b>	<b>Salida</b>	<b>Resultados</b>
Jhenny	Jhenny1234 56789	Se despliega un mensaje de bienvenida al	El sistema valida la existencia del usuario y verifica la correspondencia de la contraseña

		sistema.	con los registros de la base de datos. Esto posibilita la identificar el tipo de usuario y cargar los permisos de acceso en el sistema.
Jhenny	123456789	Despliega un mensaje acerca del error que se está cometiendo con las credenciales.	El sistema comprueba la existencia del usuario y verifica la concordancia de la contraseña con los registros de la base de datos. En caso de detectar algún error, se despliega un mensaje indicativo específico que señala la naturaleza del error cometido.

La aplicación de la prueba de caja negra en el proceso de inicio de sesión de una aplicación proporciona una evaluación exhaustiva de su funcionalidad externa. Al simular las interacciones del usuario sin conocer la estructura interna del código, estas pruebas revelan la capacidad del sistema para manejar diversos escenarios de acceso. Este enfoque no solo contribuye a identificar posibles fallos en la autenticación, sino que también asegura la confiabilidad y la seguridad del proceso de inicio de sesión, garantizando así una experiencia fluida y segura para los usuarios.

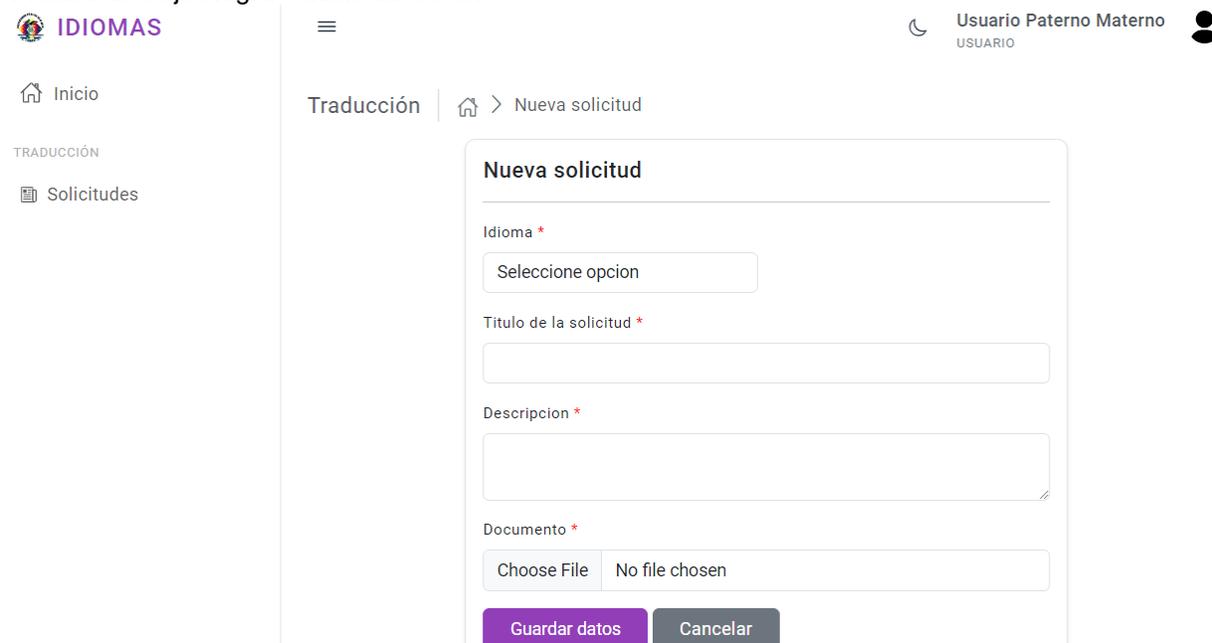
### 3.7.2. Pruebas de estrés

Las pruebas de estrés desempeñan un papel fundamental al validar el comportamiento de la aplicación frente a condiciones de carga máxima, llevando al sistema a sus límites para evaluar su capacidad de respuesta. En este contexto, se empleó la herramienta JMeter, una aplicación especializada en analizar y medir tanto el rendimiento como el comportamiento funcional de aplicaciones web. Este enfoque no solo permite identificar posibles cuellos de botella y mejorar la eficiencia del sistema, sino que

también proporciona una comprensión detallada de su rendimiento en diversos escenarios, asegurando así una experiencia de usuario óptima incluso en situaciones de alta demanda.

La aplicación de JMeter en la prueba de estrés para el sistema web de servicios de traducción se convierte en un recurso esencial para evaluar y optimizar su rendimiento bajo condiciones de carga máxima. JMeter actúa como un instrumento integral al simular múltiples usuarios concurrentes que interactúan con la aplicación, permitiendo así evaluar su capacidad de respuesta y detectar posibles puntos críticos en el sistema. Al generar cargas significativas, esta herramienta proporciona información detallada sobre el comportamiento funcional del sistema, identificando posibles cuellos de botella, tiempos de respuesta y la capacidad general de manejar solicitudes simultáneas. La aplicación de JMeter en estas pruebas de estrés no solo contribuye a mejorar la robustez y la eficiencia del sistema de servicios de traducción, sino que también garantiza una experiencia fluida y fiable para los usuarios, incluso en momentos de alta demanda.

**Figura. 3.40**  
Prueba de Caja Negra – Inicio de Sesión



The screenshot shows the 'IDIOMAS' application interface. On the left is a sidebar with a home icon and 'Inicio', a 'TRADUCCIÓN' section with a folder icon and 'Solicitudes', and a hamburger menu icon. The main header shows 'Traducción' and a breadcrumb 'Inicio > Nueva solicitud'. The top right corner displays 'Usuario Paterno Materno' and 'USUARIO' next to a user profile icon. The central form, titled 'Nueva solicitud', contains the following fields: 'Idioma \*' with a dropdown menu showing 'Seleccione opcion'; 'Titulo de la solicitud \*' with a text input field; 'Descripcion \*' with a larger text area; and 'Documento \*' with a file selection field showing 'Choose File' and 'No file chosen'. At the bottom of the form are two buttons: 'Guardar datos' (purple) and 'Cancelar' (grey).

Se evaluó el sistema de servicios de traducción sometiéndolo a pruebas de estrés mediante el envío de diversas solicitudes en un periodo de 1 minuto. Utilizando JMeter,

se obtuvieron resultados detallados en una tabla, revelando la capacidad de respuesta del sistema ante cargas variables. Estos datos son esenciales para identificar áreas de mejora y garantizar un rendimiento eficiente incluso en momentos de alta demanda.

**Tabla 3.11**  
Prueba de estrés

<b>N°</b>	<b>Muestra</b>	<b>Media</b>	<b>Min</b>	<b>Max</b>	<b>Desviación Estándar</b>	<b>%Error</b>	<b>Rendimiento</b>
<b>1</b>	100	4075	1	92001	13652,12	3%	24,2/sec
<b>2</b>	500	3249	1	107456	9897,75	24,50%	67,8/sec
<b>3</b>	1000	2523	1	136789	8456,89	55,60%	118/sec
<b>4</b>	2000	2145	1	142654	6783,02	89,50%	287/sec

# **CAPÍTULO IV**

## CALIDAD DE SOFTWARE

En este segmento, se llevará a cabo una evaluación exhaustiva destinada a demostrar la calidad del software que ha sido desarrollado. Para llevar a cabo esta evaluación, se aplicarán las métricas de calidad que la norma ISO-9126 propone. Esta norma establece un marco que abarca múltiples dimensiones esenciales de la calidad del software, lo que incluye funcionalidad, confiabilidad, mantenibilidad, usabilidad y portabilidad. Cada una de estas características se convierte en un punto de referencia clave para medir y garantizar la calidad del software, asegurando que cumple con los estándares y expectativas requeridos, además de ser eficiente y adaptable en diversos contextos. Esta evaluación exhaustiva garantiza que el software no solo cumple con sus funcionalidades básicas, sino que también es fiable, sencillo de mantener, fácil de usar y versátil en términos de su despliegue en diferentes entornos.

### 4.1. Funcionalidad

La evaluación de la funcionalidad en el software no se lleva a cabo mediante medidas directas, ya que requiere la derivación de indicadores tangibles. Esto se logra a través de métricas como los puntos de función, que permiten cuantificar tanto el tamaño como la complejidad del sistema en función de las tareas realizadas por los usuarios, además de considerar las particularidades del dominio. En el proceso de calcular el recuento total con factores de ponderación se considera a Pressman (2010), lo cual es crucial tener en consideración los siguientes aspectos:

Tabla 4.1  
Factor de ponderación

<b>Parámetro de medición</b>	<b>Cuenta</b>	<b>Factor medio</b>	<b>Total</b>
<b>N° de entradas del usuario</b>	41	4	164
<b>N° de salidas del usuario</b>	21	5	105
<b>N° de peticiones del usuario</b>	6	4	24
<b>N° de archivos</b>	21	10	210
<b>N° de interfaces externas</b>	0	10	0
<b>Cuenta Total</b>			503

A continuación, se muestra la tabla de los valores de ajuste de complejidad para obtener la variable  $f_i$ :

**Tabla 4.2**  
Valores de ajuste de la complejidad

<b>Escala</b>	<b>Complejidad</b>
<b>0</b>	No influye
<b>1</b>	Incidencia
<b>2</b>	Moderado
<b>3</b>	Medio
<b>4</b>	Significativo
<b>5</b>	Esencial

Los valores que reflejan la complejidad se determinan considerando una evaluación basada en una serie de preguntas clave, que son las siguientes:

**Tabla 4.3**  
Ajuste de complejidad

<b>FACTOR</b>	<b><math>f_i</math></b>
El sistema requiere copias de seguridad y recuperación fiable	5
Requiere comunicación de datos	5
El rendimiento es crítico	2
El sistema ejecuta en un entorno operativo existente y fuertemente utilizado	3
El sistema requiere entrada de datos interactivos	5
El sistema requiere de entrada de datos interactivos donde las entradas lleven múltiples opciones	1
Los archivos maestros se actualizan interactivamente	2
Las entradas, las salidas y archivos o peticiones son complejas	4
El procesamiento interno es complejo	2
La conversión y la instalación están incluidas en el diseño	2
El sistema requiere que la entrada de datos interactivos de transacciones se ocupe de múltiples entradas u operaciones	1

Son complejas las entradas y salidas	3
El procesamiento interno es complejo	1
El código está diseñado para ser reutilizable	5
La aplicación se ha diseñado para facilitar cambios y uso sencillo del usuario	5
<b>∑ factor de complejidad</b>	<b>50</b>

Para calcular el punto de función (PF) se considera la siguiente relación:

$$PF = cuentaTotal * [0,65 + 0.01 * \sum (f_i)]$$

Donde la cuenta total es la suma de todas las entradas del factor de ponderación. Además, se considera la confiabilidad del proyecto con un valor de 0,65 y un error mínimo de 0.01. Utilizando esta fórmula se tiene lo siguiente:

$$PF = 503 * [0,65 + 0.01 * 50]$$

$$PF = 503 * [0,65 + 0,5]$$

$$PF = 503 * [1,15]$$

$$PF = 578,45$$

Tomado en consideración el 100% del factor de complejidad la sumatoria de  $f_i = 70$ . Este dato nos permite establecer el valor del factor de complejidad máximo:

$$PF_{maximo} = 503 * [0,65 + 0.01 * 70]$$

$$PF_{maximo} = 503 * [0,65 + 0,7]$$

$$PF_{maximo} = 503 * 1,35$$

$$PF_{maximo} = 679,05$$

Así la funcionalidad se obtiene con la siguiente fórmula:

$$funcionabilidad = \frac{PF}{PF_{maximo}}$$

$$funcionabilidad = \frac{578,45}{679,05}$$

$$funcionabilidad = 0,85$$

Por tanto, la funcionalidad del sistema web se pondera en un 85% sin riesgos de fallas y operatividad y un 15% susceptible a fallas o colapso.

## 4.2. Confiabilidad

La confiabilidad del sistema se refiere a la probabilidad de que el sistema funcione sin fallos en un entorno dado durante un período de tiempo específico. Para evaluar la confiabilidad de un software, se define el momento en que comienza a operar, es decir,  $t = 0$ , y a partir de ese punto realizamos observaciones relevantes hasta un tiempo  $t = n$ .

$P(T \leq t)$  Probabilidad de fallas

$P(T \leq t) = 1 - F(t)$ , probabilidad de trabajo sin fallas

Para calcular la confiabilidad del sistema, se considera el período de tiempo durante el cual el sistema se encuentra en funcionamiento, y a partir de ese momento, se van recolectando las muestras correspondientes. Para obtener los datos se utiliza la siguiente fórmula:

$$F_{(t)} = f * e^{(-\mu * t)}$$

Donde:

$f$ : Funcionalidad del sistema

$\mu$ : Probabilidad de erros que pueda tener el sistema

$t$ : Tiempo de duración de gestión en el sistema

Se tomará un período de prueba de 30 días, en el cual se establece que, de cada diez ejecuciones, se experimenta un fallo en el sistema. Los cálculos apropiados se llevan a cabo a continuación:

$$F_{(t)} = f * e^{(-\mu * t)}$$

$$F_{(t)} = 0.85 * e^{(-\frac{1}{10} * 30)}$$

$$F_{(t)} = 0,0423$$

$$F_{(t)} = 4,23\%$$

Considerando el dato obtenido anteriormente, se reemplaza en la fórmula de probabilidades obteniendo lo siguiente:

$$F_{(T \leq t)} = F_{(t)} \Rightarrow P_{(T \leq t)} = 0,0423 = 4,23\%$$

$$P_{(T \leq t)} = 1 - F_{(t)} \Rightarrow P_{(T \leq t)} = 1 - 0,0423$$

$$P_{(T \leq t)} = 0.9577$$

$$P_{(T \leq t)} = 95,77\%$$

Por tanto, la confiabilidad del sistema es 96% en un periodo de 30 días como tiempo prueba.

### 4.3. Mantenibilidad

La mantenibilidad es un conjunto de características vinculadas con la dificultad requerida para efectuar ciertas modificaciones en el producto. Estas características incluyen su capacidad de análisis, su adaptabilidad, su estabilidad y su aptitud para ser sometido a pruebas. El estándar IEEE-982.1 recomienda el uso de un índice de madurez del software (IMS) que ofrezca una evaluación de la estabilidad del producto de software, este índice se calcula mediante la siguiente relación:

$$IMS = [M_t - (F_c + F_a + F_d)]/M_t$$

Donde:

$M_t$ : Número de módulos en la versión actual

$F_c$ : Número de módulos en la versión actual que se ha cambiado

$F_a$ : Número de módulos en la versión actual que se han añadido

$F_d$ : Número de módulo en la versión anterior que se han borrado en la versión actual

Se procede a calcular el IMS:

$$IMS = [M_t - (F_c + F_a + F_d)]/M_t$$

$$IMS = [7 - (0 + 1 + 0)]/7$$

$$IMS = 0,857$$

El sitio web tiene una estabilidad del 86% para su mantenibilidad. El 14% consta el margen de error por los cambios o modificaciones que sufriera el sistema.

### 4.4. Usabilidad

En el ámbito de la usabilidad, se espera que un sistema sea de fácil comprensión y aprendizaje. Es importante destacar que, según la norma ISO 9126, la usabilidad no se ve afectada por la funcionalidad y la eficiencia. La usabilidad se define en función

de las experiencias y percepciones de los usuarios finales o aquellos relacionados con el sistema. Para evaluar la usabilidad, se utiliza un cuestionario específico.

**Tabla 4.4**  
Valores de ajuste de la usabilidad

Escala	Complejidad
0	No influye
1	Incidencia
2	Moderado
3	Medio
4	Significativo
5	Esencial

**Tabla 4.5**  
Evaluación de usabilidad

Factor	Valor
El sitio web es fácil de usar	5
Las plantillas son de su agrado	5
El sitio web tiene vistas fáciles de comprender	5
El sitio web responde rápidamente a las solicitudes	4
El sitio web facilita el trabajo	5
El sitio web reduce el tiempo de trabajo	5
El sitio web tiene facilidad de navegación en sus distintas opciones	4
Las operaciones del sitio web no son complicadas	5
El sitio web proporciona las repuestas requeridas	3
El sitio web no presenta errores	4
Total	45

Para calcular la usabilidad del sistema se tiene la siguiente fórmula:

$$USABILIDAD = \frac{\left(\frac{\sum i}{n}\right) * 100}{5}$$

Donde:

$\Sigma i$ : Es la sumatoria de valores

$n$ : Es el número de preguntas

$$USABILIDAD = \frac{\left(\frac{\Sigma i}{n}\right) * 100}{5}$$

$$USABILIDAD = \frac{\left(\frac{45}{10}\right) * 100}{5}$$

$$USABILIDAD = 90\%$$

Según los datos recopilados, se puede concluir que el sitio web posee un nivel de usabilidad del 90%.

#### **4. 5. Portabilidad**

Es la facilidad con la que el software puede llevarse de un ambiente a otro según lo indican los siguientes atributos: adaptable, instalable, conformidad y sustituible. La portabilidad del sitio web constituye un atributo esencial que asegura su óptimo desempeño en diversas plataformas y dispositivos. Este enfoque técnico posibilita una implementación sin inconvenientes en múltiples entornos, siempre que dispongan de un servidor web y un sistema de gestión de bases de datos, en nuestro caso, MySql. En virtud de su orientación hacia tecnologías web, el sitio web se presenta como plenamente accesible desde una amplia gama de dispositivos, tanto computadoras como dispositivos móviles, que cuenten con acceso a la red y un navegador compatible. Este nivel de versatilidad garantiza una experiencia de usuario consistente, independientemente de la plataforma empleada para acceder al sitio web.

#### **4. 7. Seguridad**

La seguridad del sistema web de servicios de traducción se aborda de manera completa, considerando tanto aspectos físicos como lógicos para garantizar la protección integral de la información almacenada. Desde una perspectiva física, se han implementado medidas preventivas, como la instalación de estabilizadores de energía y extintores, y se ha restringido el acceso físico a personal de confianza. Estas

precauciones buscan evitar daños causados por eventos como sobretensiones eléctricas o situaciones de emergencia.

En el ámbito lógico, se emplea codificación y autenticación para garantizar que solo usuarios autorizados puedan acceder a la información almacenada. La implementación de roles de usuario facilita la gestión de derechos de acceso, mientras que el uso de claves de cifrado evolutivo y una autenticación estándar basada en algoritmos hash seguros refuerza la robustez del sistema contra posibles amenazas cibernéticas.

La combinación de medidas físicas y lógicas demuestra un enfoque equilibrado y proactivo hacia la seguridad, salvaguardando la integridad y confidencialidad de la valiosa información manejada por la institución. Con el cierre automático de conexiones y otras prácticas seguras implementadas, se establece un entorno en el que la protección de datos es una prioridad constante, contribuyendo a la fiabilidad y solidez del sistema web de servicios de traducción.

#### **4. 6. Estimación de costos del software- COCOMO II**

Los procedimientos de evaluación de gastos en proyectos de desarrollo de software desempeñan un papel fundamental al señalar las justificaciones presupuestarias, orientar la planificación y supervisión del proyecto, y mejorar el análisis de inversión en los proyectos de software. Estos métodos permiten establecer una relación cuantitativa entre la inversión de recursos y el lapso de desarrollo.

Para calcular la estimación del costo de construcción del software, se utilizará el valor derivado de los puntos de función reales (PF = 639). Al aplicar el factor de conversión a KLDCH de 10 para el lenguaje de programación PHP, se obtiene el siguiente resultado:

$$LDC = PF_{real} * Factor \frac{LDC}{PF}$$

$$LDC = 639 * 7$$

$$LDC = 6390$$

De acuerdo al resultado, las líneas de código en su totalidad son 18531. Estimado el código reutilizable al 100% será:

$$KLDC = \frac{LDC}{1000}$$

$$KLDC = \frac{6390}{1000}$$

$$KLDC = 6.39$$

En consecuencia, se encuentran asignadas un total de 4,473 líneas de código a lo largo de la distribución del proyecto de desarrollo de software.

Según el modelo COCOMO es importante establecer las personas requeridas, el tiempo requerido y el esfuerzo empleado por persona al mes. Las ecuaciones se presentan a continuación:

$$E = a * (KLDC)^b * FAE$$

$$T = c * E^d$$

$$P = \frac{E}{T}$$

Donde:

*E*: es el esfuerzo requerido por el proyecto de una persona en un mes.

*T*: es el tiempo requerido por el proyecto en un mes

*P*: es el número de personas requerido por el proyecto

*a, b, c* y *d* son las constantes con valor definido en una tabla según cada submodelo.

*KL*: es la cantidad de líneas de códigos en miles

**Tabla 4.6**  
Valores constates

<b>Escala</b>	<b>A</b>	<b>B</b>	<b>c</b>	<b>D</b>
Orgánico	3.20	1.5	2.50	0.38
Semi orgánico	3.00	1.12	2.50	0.35
Empotrado	2,80	1.20	2.50	0.32

A continuación, se enumeran los distintos valores de los multiplicadores de esfuerzo seleccionados para el sistema.

Tabla 4.7

Multiplicadores de esfuerzo para la estimación de costos

Multiplicador de esfuerzo (ME)		VALORACIÓN					
		Muybajo	Bajo	Nominal	Alto	MuyAlto	ExtraAlto
<b>Atributos del producto</b>							
<b>RELY</b>	Fiabilidad requerida del software	0.75	0.88	1.00	<b>1.15</b>	1.40	-
<b>DATA</b>	Tamaño de la base de datos	-	<b>0.94</b>	1.00	1.08	1.16	-
<b>CPLX</b>	Complejidad del producto	0.70	<b>0.85</b>	1.00	1.15	1.30	1.65
<b>Atributos de la computadora</b>							
<b>TIME</b>	Restricciones del tiempo de ejecución	-	-	<b>1.00</b>	1.11	1.30	1.66
<b>STOR</b>	Restricciones del almacenamiento princ.	-	-	<b>1.00</b>	1.06	1.21	1.56
<b>VIRT</b>	Inestabilidad de la máquina virtual	-	0.87	<b>1.00</b>	1.15	1.30	-
<b>TURN</b>	Tiempo de respuesta del computador	-	0.87	1.00	<b>1.07</b>	1.15	-
<b>Atributos del personal</b>							
<b>ACAP</b>	Capacidad del analista	1.46	1.19	1.00	<b>0.86</b>	0.71	-
<b>AEXP</b>	Experiencia en la aplicación	1.29	<b>1.13</b>	1.00	0.91	0.82	-
<b>PCAP</b>	Capacidad de los programadores	1.42	1.17	1.00	0.86	<b>0.70</b>	-
<b>VEXP</b>	Experiencia en S.O. utilizado	1.21	-	1.00	<b>0.90</b>	-	-
<b>LEXP</b>	Experiencia en el lenguaje de programación	1.14	-	1.00	<b>0.95</b>	-	-
<b>Atributos del proyecto</b>							
<b>MODP</b>	Práctica de programación moderna	1.24	1.10	1.00	<b>0.91</b>	0.82	-
<b>TOOL</b>	Uso de herramientas de software	1.24	1.10	1.00	0.91	<b>0.83</b>	-
<b>SCED</b>	<b>Limitaciones de planificación del proyecto.</b>	<b>1.24</b>	<b>1.08</b>	<b>1.00</b>	<b>1.04</b>	<b>1.10</b>	-

Al reemplazar los valores de la tabla de factores de esfuerzo en la ecuación de la FAE, obtenemos lo siguiente:

$$FAE = 1.15 * 0.94 * 0.85 * 1.00 * 1.00 * 1.00 * 1.07 * 0.86 * 1.00 * 0.70 * 0.90 * 0.95 * 0.91 * 0.83 * 1.00$$

$$FAE = 0.38$$

Mediante la aplicación de las ecuaciones previamente mencionadas y considerando las diversas constantes, se generan los resultados que se detallan a continuación.

Donde esfuerzo es:

$$E = a * (KLDC)^b * FAE$$

$$E = 3.20 * (6.39)^{1.05} * 0.38$$

$$E = 8.525$$

Donde tiempo es:

$$T = c * E^d$$

$$T = 2.50 * 8.525^{0.38}$$

$$T = 5.644$$

Donde persona es:

$$P = \frac{E}{T}$$

$$P = \frac{8.525}{5.644}$$

$$P = 1,510$$

Un desarrollador web junior en Bolivia percibe un sueldo entre básico. Cálculo de costo del proyecto se tiene lo siguiente:

$$\text{Costo Mes} = \text{Número Personas} * \text{Salario Promedio}$$

$$\text{Costo Mes} = 1 * 2362$$

$$\therefore \text{Costo Mes} = 2362 \text{ Bs (mes)}$$

Costo Total de Proyecto:

$$\text{Costo Total del Sistema} = \text{Costo Mes} * T$$

$$\text{Costo Total del Sistema} = 2363 * 5.644$$

$$\text{Costo Total del Sistema} = 13336.7 \text{ Bs}$$

# **CAPÍTULO V**

## CONCLUSIONES Y RECOMENDACIONES

### 4.1. Conclusiones

El desarrollo y la implementación del "Sistema Web para Servicio de Traducción" en la Carrera de Lingüística e Idiomas de la Universidad Pública de El Alto representan un hito importante en la optimización de los servicios de traducción. El problema principal, la falta de un sitio web para acceder a estos servicios, había generado demoras y dificultades en la atención al usuario, quienes debían esperar la recomendación de un traductor, contactarlos por teléfono o buscarlos personalmente. Esto condujo a una falta de eficiencia y eficacia en la gestión de traducciones.

La implementación del sistema web abordó con éxito los problemas secundarios identificados, como la falta de información oportuna para los usuarios, la gestión manual de documentos por parte de los traductores y la dificultad para ubicar las oficinas en la universidad. Los objetivos planteados, que incluyen el análisis de requerimientos, la estructuración del sistema y la ejecución con evaluación de reportes, se han cumplido de manera satisfactoria.

Desde una perspectiva técnica, este sistema web se ha diseñado para eliminar las dificultades que enfrentaban los usuarios y traductores en la obtención y prestación de servicios de traducción, especialmente en idiomas nativos. Además, el sistema agiliza la gestión y organización de archivos de traducción, reduciendo la burocracia y facilitando el acceso a los servicios. Esto contribuye a una planificación estratégica eficaz y a una gestión organizada de los archivos, mejorando significativamente la calidad de los servicios de traducción ofrecidos por la Carrera de Lingüística.

En resumen, el "Sistema Web para Servicio de Traducción" es una herramienta valiosa que ha logrado superar los desafíos identificados, mejorando la eficiencia y eficacia de los servicios de traducción en la universidad. Su implementación sienta las bases para una administración más eficaz y un acceso simplificado a estos servicios, beneficiando tanto a los usuarios como a los traductores y administradores. Este proyecto ha

alcanzado con éxito sus objetivos y representa un avance significativo en la modernización de los servicios de traducción.

#### **4.2. Recomendaciones**

Una vez desarrollado el sistema de Servicios de traducción se tiene las siguientes recomendaciones:

- Se recomienda realizar copias de seguridad de la base de datos de forma periódico a la institución con el fin de poseer respaldo de información que aseguren la integridad del sistema.
- Capacitar a los nuevos usuarios para que puedan operar correctamente.
- Se recomienda establecer un plan de mantenimiento continuo para realizar actualizaciones y mejoras según sea necesario.

## **BIBLIOGRAFÍA**

Abud, M. (2012). *Calidad en la Industria del Software. La Norma ISO-126*. Obtenido de <https://www.nacionmulticultural.unam.mx/empresasindigenas/docs/2094.pdf>

Arias, J., & Durango, C. (15 de Enero de 2018). Propuesta de un método para desarrollar Sistemas de Información Geográfica a partir de la metodología de desarrollo ágil - SCRUM. *Cuaderno activa*, págs. 29 - 41.

Baldeón, E. (2015). *MÉTODO PARA LA EVALUACIÓN DE CALIDAD DE SOFTWARE BASADO EN ISO/IEC 25000*. Lima: USMP- Facultad de ingeniería y arquitectura.

Booch, G., & et al. (2006). *El lenguaje Unificado de Modelado- Guía del usuario*. Madrid: ADDISSON WESLEY.

Buitrón, L., & y Otros. (2021). *Desarrollo De Una Aplicación Web Para La Gestión De Citas E Historia Clínica De Pacientes, Utilizando Metodologías De Desarrollo Ágil. Caso De Estudio "Consultorio Médico Medicina Integral" Del Dr. Cando Herrera Johnny Stalin*. Sangolquí - Ecuador: Universidad de las Fuerzas Armadas.

Callejas, M., & et. al. (2017). *Ingeniería y tecnología- Entramado- Universidad Libre de Cali*. Obtenido de Modelos de calidad del software, un estado del arte: <https://www.redalyc.org/journal/2654/265452747018/html/>

Cervantes, H., & et. al. (2016). *Arquitectura de software. Concepto y ciclo de desarrollo*. Mexico, D.F.: Cengage Learning.

Chuquimia, S. (2021). *Sistema web de consulta médica virtual para el servicio de ginecología del Hospital de la Mujer*. La Paz - Bolivia: Universidad Mayor de San Andres.

Coaquira, E. (2017). *Sistema Web de reserva de hoteles caso: Agencia de Viajes Definitely Bolivia*. La Paz: Universidad Mayor de San Andres.

- Coaquira, E. (2017). *SISTEMA WEB DE RESERVA DE HOTELES CASO: AGENCIA DE VIAJES DEFINITELY BOLIVIA*. La Paz - Bolivia: Universidad Mayor de San Andres.
- Cobo, Á., & y Otros. (2005). *PHP y MySQL Tecnologías para el desarrollo de aplicaciones web*. España: Rústica-Hilo.
- Cuesta, C. (2002). *Arquitectura de software dinámica basada en reflexión*. España: Universidad de Valladolid.
- De Toro, A. (18 de Agosto de 2022). *Business Review revista digital - Universidad Europea Miguel de Cervantes*. Obtenido de Se trata del periodo de tiempo en el que el equipo trabaja para finalizar una cantidad de tareas establecidas. El sprint es el corazón de este framework y hacerlo de forma correcta potenciará la eficacia del trabajo y reducirá los problemas a la hora de l: <https://www.escueladenegociosydireccion.com/revista/business/scrum-framework-agiliza-trabajo-equipo/>
- Estrada, M., & y otros. (09 de Diciembre de 2021). *Dominio de las ciencias*. Obtenido de Revisión Sistemática de la Metodología Scrum para el Desarrollo de Software: <https://dominiodelasciencias.com/ojs/index.php/es/article/view/2429/0>
- Fernández, L. (Junio de 2006). *Arquitectura de software*. (C. Montes de Oca, Entrevistador)
- Flores, D., & et. al. (2015). *SISTEMA DE GESTIÓN DE E-PORTAFOLIOS DE APRENDIZAJE PARA LA CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA UTILIZANDO LA METODOLOGÍA UWE-UML*. SANGOLQUÍ: ESPE-Universidad de las Fuerzas Armadas.
- Formación en Ambientes Virtuales de Aprendizaje. (s/n). *Introducción al Lenguaje de Modelado Unificado (UML)*. *Servicio Nacional de Aprendizaje*.
- Galiano, L. (2012). Informe de la metodología aplicada en mi solución informática de mi proyecto. En I. U. BOLÍVAR, *PROGRAMA NACIONAL DE FORMACIÓN EN*

*INFORMÁTICA*. Bolívar: MINISTERIO DEL PODER POPULAR PARA LA EDUCACIÓN UNIVERSITARIA.

García , F., & García, A. (2018). *Fundamentos de la vista de casos de uso*. Salamanca: Universiad de Salamanca. Obtenido de <https://repositorio.grial.eu/bitstream/grial/1155/1/UML%20-%20Casos%20de%20uso.pdf>

Gonzales , G., & Madriz, F. (junio de 2018). *Scielo*. Obtenido de Uniciencia: [https://www.scielo.sa.cr/scielo.php?script=sci\\_arttext&pid=S2215-34702018000100118](https://www.scielo.sa.cr/scielo.php?script=sci_arttext&pid=S2215-34702018000100118)

Gonzales, C. (2020). *Sistema web de administración de historias clínicas caso: (Hospital de la Mujer)*. El Alto - Bolivia: Universidad Pública de El Alto.

Jézéquel, J.-M., & et al. (2022). *The Unified Modeling Language*. Germany: Springer.

Kimmel, P. (2008). *Manual de UML*. Mexico: McGRAW-HILL INTERAMERICANA EDITORES.

Laravel. (21 de 06 de 2022). *Hostalia*. Obtenido de <https://pressroom.hostalia.com/contents/ui/theme/images/framework-laravel-wp-hostalia.pdf>

López, A. (2018). *Estudio comparativo de metodologías tradicionales y ágiles para proyectos de Desarrollo de Software*. España: UNIVERSIDAD DE VALLADOLID. Obtenido de <https://uvadoc.uva.es/bitstream/handle/10324/32875/TFG-I-1015.pdf?sequence=1&isAllowed=y>

Maida, E., & Pacienza, J. (2015). *Metologías de desarrollo de software*. Argentina: Universidad Católica Argentina.

Maldonado, J. (2016). *DESARROLLO E IMPLEMENTACION DE UN SISTEMA WEB DE SEGUIMIENTO Y EVALUACION DE LAS PRÁCTICAS PRE-PROFESIONALES PARA LA FACULTAD DE INGENIERIA ESCUELA CIVIL DE LA PUCE*. Quito: PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR.

Obtenido de [http://repositorio.puce.edu.ec/bitstream/handle/22000/12562/Tesis\\_Teoria.pdf?sequence=1&isAllowed=y](http://repositorio.puce.edu.ec/bitstream/handle/22000/12562/Tesis_Teoria.pdf?sequence=1&isAllowed=y)

Mariño, S., & Alfonzo, P. (2014). *Implementación de SCRUM en el diseño del proyecto del Trabajo Final de Aplicación*. Colombia: Universidad Tecnológica de Pereira. Obtenido de <https://www.redalyc.org/articulo.oa?id=84933912009>

Nava, C., & Otros. (2019). Ingeniería Web: Aplicando UWE para modelado de una aplicación web. En I. d. Orizaba, *Comprender el entorno. Camino hacia el conocimiento* (pág. 4). Mexico: UniVo.

Ñaupas, H., & y Otros. (2014). *Metodología de la investigación Cuantitativa-Cualitativa y Redacción de la Tesis*. Bogotá, Colombia: Ediciones de la U.

Pradel, J., & Raya, J. (2013). *Introducción a la ingeniería del software*. Barcelona: Universidad Oberta de Catalunya.

Rashid, A., & y Otros. (2021). *PatDoc – Online Appointment Management System*. Karnataka - India: Jain University.

Rodríguez , E. (2005). *Metodología de la investigación*. Mexico: Universidad Juárez Autónoma de Tabasco.

San Juan, V. (27 de Abril de 2016). *Sitio web de Aeurus*. Obtenido de Ventajas de los sistemas web: <https://www.aeurus.cl/blog/ventajas-de-los-sistemas-web/#:~:text=Se%20denomina%20sistema%20web%20a,una%20intranet%20mediante%20un%20navegador.>

Silva, K. (2018). *Online Appointment System For Kotelawala Defence University Hospital*. Sri Lanka- South Asia: University of Colombo School of Computing.

Tramullas, J. (1997). Los sistemas de información: una reflexión sobre información, sistema y documentación. En U. d. Zaragoza, *Revista General de Información y Documentación* (págs. 207- 223). Madrid: Servicio de Publicaciones Universidad Computense.

Unhelkar, B. (2005). *Verification and Validation for Quality of UML 2.0 Models*. Canada: Wiley-Interscience.

Universidad de Piura. (10 de Septiembre de 2023). *Administración de servicios: mucho más que la buena atención*. Obtenido de <https://www.udep.edu.pe/admision/lima/administracion-de-servicios-mucho-mas-que-la-buena-atencion/>

Valdivia, C. (2022). *Anales de Filología Francesa. Universidad de Murcia*. Obtenido de <https://doi.org/10.6018/analesff>

Valenzuela, M. (2021). *Implementación de un Sistema Web para la Gestión de Consultas Médicas en una Entidad de Salud Privada*. Lima - Perú: Universidad Tecnológica del Perú.

Vukovic, J. (2012). ¿Cómo definimos el concepto de traducción? *Memoria académica*.

Zambrano, J. (2020). *Implementación de un Sistema Web para mejorar el proceso de Reserva de Citas en el Policlínico Dr. Nixon*. Bagua Grande - Perú: Universidad Politécnica Amazónica.

# **ANEXOS**

El Alto, 27 de noviembre de  
2023

Señor:  
Lic. Ing. William Roque Roque  
**DIRECTOR DE CARRERA**  
**INGENIERÍA DE SISTEMAS**  
Presente. –

**REF. AVAL DE CONFORMIDAD**

Distinguido director de carrera:  
Mediante la presente tengo a bien comunicarle mi conformidad del Trabajo de Grado:

**TÍTULO:** “Sistema web para administración en servicios de traducción”

**CASO:** Consultorio Lingüístico de la Carrera Lingüística e Idiomas

**MODALIDAD:** Proyecto de Grado

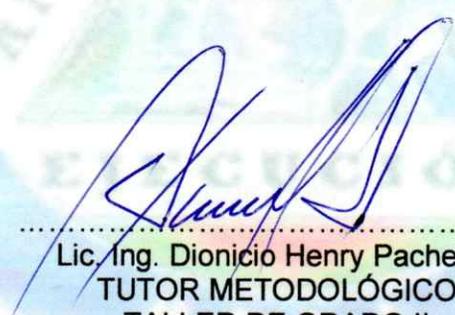
Univ. Jhenny Celia Cusicanqui Maldonado

**REGISTRO UNIVERSITARIO:** CP0010036514

**CÉDULA DE IDENTIDAD:** 8365192LP

Para su defensa pública y evaluación correspondiente a la materia de Taller de Grado II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de sistemas de la Universidad Pública de El Alto.

Atentamente,



.....  
Lic. Ing. Dionicio Henry Pacheco Rios  
TUTOR METODOLÓGICO  
TALLER DE GRADO II

El Alto, 27 de noviembre de  
2023

Señor:  
Lic. Ing. Dionicio Henry Pacheco Rios  
**TUTOR METODOLÓGICO**  
**TALLER DE GRADO II**  
Presente. –

**REF. AVAL DE CONFORMIDAD**

Distinguido tutor metodológico:  
Mediante la presente tengo a bien comunicarle mi conformidad del Trabajo de Grado:

**TÍTULO:** "Sistema web para administración en servicios de traducción"  
**CASO:** Consultorio Lingüístico de la Carrera Lingüística e Idiomas  
**MODALIDAD:** Proyecto de Grado  
Univ. Jhenny Celia Cusicanqui Maldonado  
**REGISTRO UNIVERSITARIO:** CP0010036514  
**CÉDULA DE IDENTIDAD:** 8365192LP

Para su defensa pública y evaluación correspondiente a la materia de Taller de Grado II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de sistemas de la Universidad Pública de El Alto.

Atentamente,



M.Sc. Lic. Ing. Ramiro Kantuta  
Limachi  
TUTOR REVISOR

El Alto, 27 de noviembre de  
2023

Señor:  
Lic. Ing. Dionicio Henry Pacheco Rios  
**TUTOR METODOLÓGICO**  
**TALLER DE GRADO II**  
Presente. –

**REF. AVAL DE CONFORMIDAD**

Distinguido tutor metodológico:  
Mediante la presente tengo a bien comunicarle mi conformidad del Trabajo de Grado:

**TÍTULO:** "Sistema web para administración en servicios de traducción"  
**CASO:** Consultorio Lingüístico de la Carrera Lingüística e Idiomas  
**MODALIDAD:** Proyecto de Grado  
Univ. Jhenny Celia Cusicanqui Maldonado  
**REGISTRO UNIVERSITARIO:** CP0010036514  
**CÉDULA DE IDENTIDAD:** 8365192LP

Para su defensa pública y evaluación correspondiente a la materia de Taller de Grado II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de sistemas de la Universidad Pública de El Alto.

Atentamente,



.....  
Lic. Maria Magdalena Aguilar Guanto  
TUTOR ESPECIALISTA



# Universidad Pública de El Alto

Creada por Ley 2115 del 5 de Septiembre de 2000 y Autónoma por Ley 2556 del 12 de Noviembre de 2003

El Alto, 27 de noviembre de 2023

Señor:  
Lic. Ing. Dionicio Henry Paccheco Rios  
**TUTOR METODOLÓGICO**  
**TALLER DE GRADO II**  
Presente. –

## REF. AVAL DE CONFORMIDAD

Distinguido tutor metodológico:  
Mediante la presente tengo a bien comunicarle mi conformidad del Trabajo de Grado:

**TITULO:** "Sistema web para administración en servicios de traducción"  
**CASO:** Consultorio Lingüístico de la Carrera Lingüística e Idiomas  
**MODALIDAD:** Proyecto de Grado  
Univ. Jhenny Celia Cusicanqui Maldonado  
**REGISTRO UNIVERSITARIO:** CP0010036514  
**CEDULA DE IDENTIDAD:** 8365192

De tal forma cabe recalcar que el "Sistema web para administración en servicios de traducción" ha sido **IMPLEMENTADO** y satisface los requerimientos de la institución, de esta forma se dio cumplimiento de los objetivos del presente.

En cuanto certifico, en honor a la verdad, para fines consiguientes del interesado para su defensa pública y desarrollo correspondiente a la materia de Taller de Grado II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Atentamente,

  
**Lic. Teodosia N. Tancara Caltecusi**  
**DIRECTORA**  
CARRERA DE LINGÜÍSTICA E IDIOMAS  
UNIVERSIDAD PÚBLICA DE EL ALTO



# **MANUAL DE USUARIO**

**SISTEMA WEB PARA  
ADMINISTRACIÓN EN SERVICIOS  
DE TRADUCCIÓN**

**AUTORA: JHENNY CELIA CUSICANQUI  
MALDONADO**

**VERSIÓN: 1.0**

**FECHA: 27/11/2023**

# MANUAL DE USUARIO

## 1. INTRODUCCIÓN

Este manual no tiene como objetivo proporcionar una instrucción detallada acerca del aprendizaje individual de las distintas herramientas utilizadas en el proceso de desarrollo del sistema web, sino más bien su finalidad es la documentación de su implementación en el contexto del desarrollo del sistema web.

## 2. OBJETIVO

- Otorgar soporte a los usuarios del sistema del centro de traducciones de la carrera de Lingüística e Idiomas que prestará servicios a la población en general.

## 3. CONTENIDO

### **Ingreso al sistema**

Este sistema web es compatible con los navegadores: Chrome y FireFox. Después de ingresar la sistema, debe ingresar las credenciales a través del botón **Acceder** en los campos de contraseña.

## Barra de navegación y portada

[Inicio](#) [Objetivos](#) [Idiomas](#) [Registrarse](#)

[Acceder](#)

Botón acceder



## Sección Body

### Objetivos del Centro de Traducción

El servicio del Centro de Traducción e Interpretación de la Carrera de Lingüística e Idiomas brinda ayuda de traducción e interpretación de documentos que requieran los estudiantes, docentes administrativos y público en general de escasos recursos de la ciudad de El Alto, cumpliendo de esta manera el rol social al interior de la universidad, así mismo con la sociedad.





<input type="text" value="CI"/>	<input type="text" value="Expedido"/>
<input type="text" value="Apellido Paterno"/>	<input type="text" value="Apellido Materno"/>
<input type="text" value="Nombres"/>	<input type="text" value="mm/dd/yyyy"/>
<input type="text" value="Email"/>	<input type="text" value="ESTADO CIVIL"/>
<input type="text" value="Profesion"/>	<input type="text" value="admin"/>
<input type="password" value="....."/>	<input type="text" value="Confirmar contraseña"/>

## Footer

**Institucion**  
Objetivos  
Idiomas

**Contactos**  
Carrera de Linguistica e Idiomas

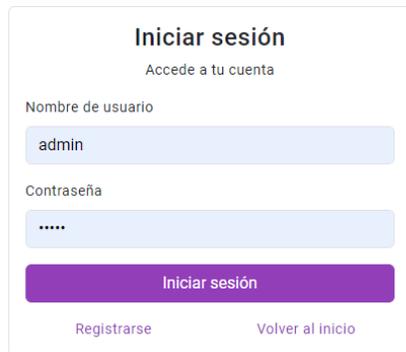
**Redes Sociales**  
 

Copyright Jhenny © 2023

## Iniciar sesión \_administrador

Para acceder al sistema web en calidad de administrador, se debe seguir un procedimiento específico. Este proceso implica utilizar credenciales de

administrador o una función de inicio de sesión diseñada para usuarios con privilegios administrativos.



**Iniciar sesión**  
Accede a tu cuenta

Nombre de usuario  
admin

Contraseña  
.....

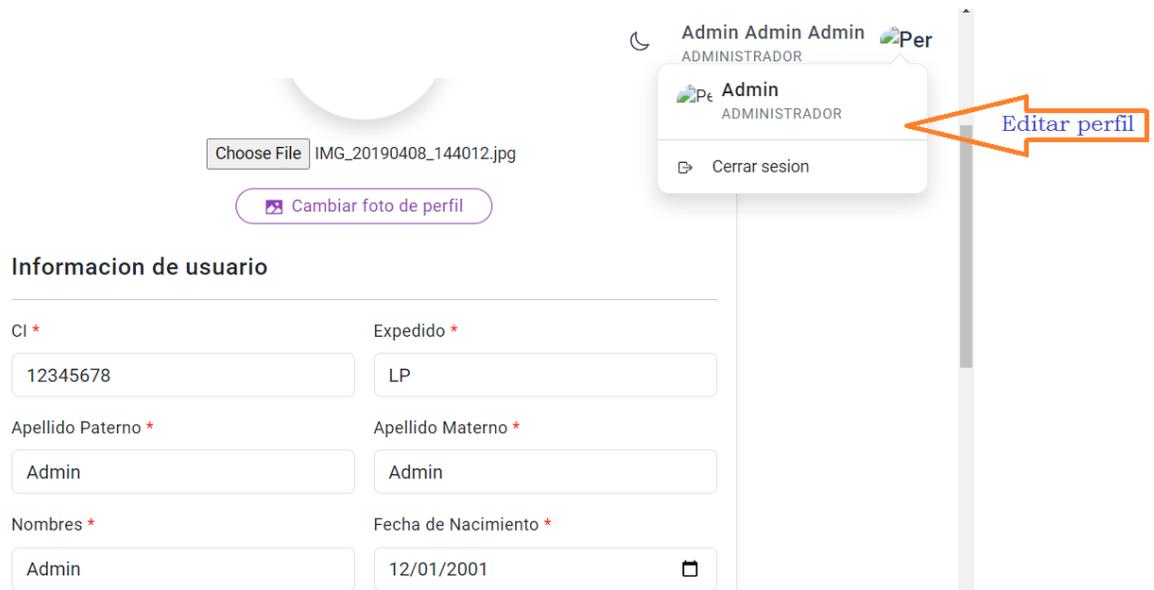
**Iniciar sesión**

[Registrarse](#) [Volver al Inicio](#)



### **Editar perfil \_ administrador**

La función de "Editar Perfil" en el sistema web del Centro de Traducción es una característica esencial que permite a los usuarios modificar y actualizar su información personal. Este proceso, parte integral de la experiencia del usuario, garantiza que los perfiles se mantengan precisos y personalizados, optimizando así la interacción y colaboración dentro del centro de traducción.



### Cerrar sesión \_ administrador

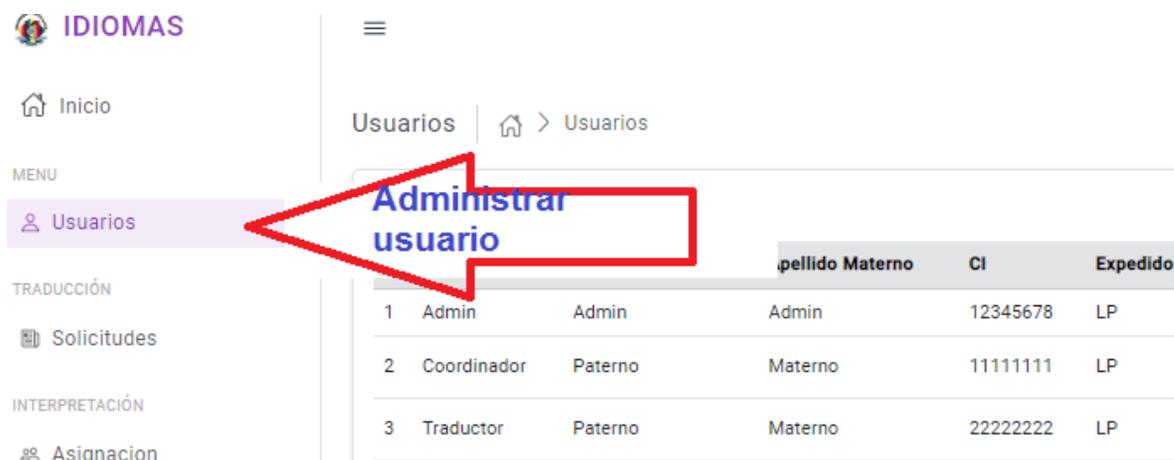
La función de "Cerrar Sesión" en el sitio web del Centro de Traducción es un componente fundamental que permite a los usuarios finalizar su sesión de manera segura y proteger su información personal. Al utilizar esta función, los usuarios pueden garantizar que nadie más tenga acceso a su cuenta y salvaguardar la privacidad de sus datos.



### Administrar usuarios \_ administrador

La "Administración de Usuarios" en el sitio web del Centro de Traducción es una función crítica que permite a los administradores y moderadores supervisar y gestionar las cuentas de los usuarios de manera eficiente. Esto abarca la creación

de nuevas cuentas, la asignación de roles y privilegios, así como la suspensión o eliminación de cuentas en caso necesario.



The screenshot shows the 'IDIOMAS' system interface. On the left, there is a sidebar menu with options: 'Inicio', 'MENU' (containing 'Usuarios'), 'TRADUCCIÓN' (containing 'Solicitudes'), and 'INTERPRETACIÓN' (containing 'Asignación'). The main content area is titled 'Usuarios' and contains a table of users. A red arrow points to a button labeled 'Administrar usuario' which is positioned over the table. The table has columns for 'Apellido Materno', 'CI', and 'Expedido'.

			Apellido Materno	CI	Expedido
1	Admin	Admin	Admin	12345678	LP
2	Coordinador	Paterno	Materno	11111111	LP
3	Traductor	Paterno	Materno	22222222	LP

### **Administración de solicitudes de traducción**

La "Administración de Solicitudes de Traducción" en el sitio web del Centro de Traducción desempeña un papel crucial en el proceso de gestión de proyectos de traducción. Esta función permite a los usuarios con privilegios de coordinador y administración supervisar, revisar y coordinar las solicitudes de traducción presentadas por los usuarios de la comunidad. Dentro de este módulo, los encargados pueden evaluar la calidad y pertinencia de las solicitudes, asignar traductores, realizar un seguimiento del progreso y asegurarse de que se cumplan los plazos establecidos.

Para acceder al módulo de "Administración de Solicitudes de Traducción", los usuarios con privilegios correspondientes deben iniciar sesión en el sistema y navegar hacia la sección izquierda. Una vez dentro, podrán visualizar una lista de las solicitudes pendientes y tomar medidas como la revisión y aprobación, la asignación de traductores, la comunicación con los solicitantes y el seguimiento del progreso de las traducciones. Además, tendrán la capacidad de actualizar el estado de las solicitudes, lo que garantiza una gestión efectiva y una comunicación transparente en todo el proceso de traducción.

Idioma	Título	Documento	Asignado	Documento traducido	Estado de Pago	Comprobante de pago	Estado	Acciones
Aymara		Descargar documento	Traductor Paterno Materno		PENDIENTE	Enviado	ACTIVO	Editar
2 INGLES	Traduccion a Ingles	Descargar documento	Traductor Paterno Materno	Descargar documento	PENDIENTE	No enviado	ENTREGADO	
3 QUECHUA	Traduccion a Quechua	Descargar documento			PENDIENTE	No enviado	ACTIVO	Editar

## Administración de interpretaciones

Esta función permite a los usuarios con permisos de coordinador y administración supervisar y gestionar las solicitudes de interpretación presentadas por los usuarios de la comunidad. Los responsables de esta tarea pueden evaluar la disponibilidad de traductores, programar eventos o sesiones de interpretación, y asegurarse de que las solicitudes se atiendan de manera efectiva y eficiente.

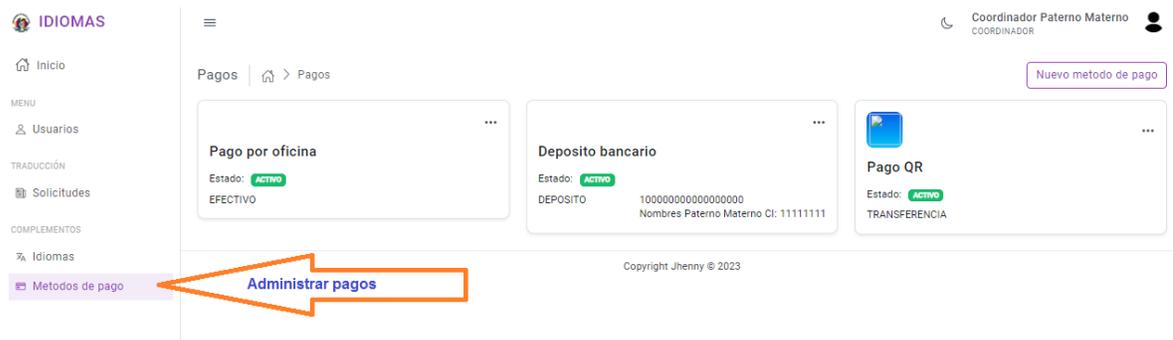
Para acceder al módulo de "Administración de Solicitudes de Interpretación," los usuarios con permisos correspondientes deben iniciar sesión en el sistema y dirigirse a la sección izquierda. Dentro de esta área, podrán visualizar un listado de las solicitudes pendientes y realizar diversas acciones, como la asignación de intérpretes, la programación de eventos, el seguimiento del estado de las solicitudes y la comunicación con los solicitantes. Además, tendrán la capacidad de actualizar la información sobre las solicitudes, garantizando una gestión efectiva y una comunicación transparente en todo el proceso de interpretación.

ID	Idioma	Título	Descripcion	Documento	Documento traducido	Estado del pago	Tipo pago	Comprobante	Estado	Acciones
asignadas										

## Administrar pagos

La "Administración de Pagos de Servicios" en el sitio web del Centro de Traducción constituye una función crítica para asegurar una gestión financiera eficiente. Este módulo permite a los responsables y administradores realizar un seguimiento detallado de los pagos asociados a los servicios de traducción proporcionados por el centro. Los usuarios pueden registrar, verificar y gestionar los pagos de los clientes y realizar un seguimiento de los saldos solicitudes.

Para acceder al módulo de "Administración de Pagos de Servicios," los usuarios autorizados deben iniciar sesión en el sistema y dirigirse a la sección izquierda el sitio web. Una vez dentro, podrán visualizar un resumen de los pagos pendientes, verificar la información de los clientes y establecer fechas límite para los pagos pendientes.



# **MANUAL TÉCNICO**

**SISTEMA WEB PARA  
ADMINISTRACIÓN EN SERVICIOS  
DE TRADUCCIÓN**

**AUTORA: JHENNY CELIA CUSICANQUI  
MALDONADO**

**VERSIÓN: 1.0**

**FECHA: 27/11/2023**

## OBJETIVOS

- Proporcionar una guía concisa para desarrolladores y administradores, abordando la implementación, configuración y mantenimiento del Sistema Web de Administración de Servicios de Traducción, con el fin de garantizar una gestión eficiente de la plataforma.
- Facilitar a los desarrolladores una comprensión clara de la estructura y lógica del código fuente, así como las mejores prácticas de programación, con el objetivo de optimizar el proceso de desarrollo y garantizar la escalabilidad del Sistema Web de Administración de Servicios de Traducción.
- Capacitar a los administradores del sistema en la identificación rápida y eficaz de problemas, junto con la implementación de soluciones efectivas. El objetivo es minimizar el tiempo de inactividad y mantener la resiliencia del sistema, asegurando así un servicio de traducción continuo y confiable

## **REQUERIMIENTOS MÍNIMOS DE HARDWARE:**

Procesador: Core

Memoria RAM: mínimo: 1 Gigabytes (GB)

Disco Duro: 250Gb.

## **REQUERIMIENTOS MÍNIMOS DE SOFTWARE**

Privilegios de administrador

Sistema Operativo: Windows 7/ 8 /8.1 /10

Apache

MySQL

## **HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO**

### **PHP:**

Que significa "Hypertext Preprocessor", es un lenguaje de programación de código abierto ampliamente utilizado para el desarrollo web. Diseñado especialmente para la creación de sitios dinámicos, PHP se ejecuta del lado del servidor y permite la incrustación de código directamente en el HTML, facilitando la interacción con bases de datos, manipulación de archivos y otros servicios web. Su sintaxis simple y su integración fluida con HTML hacen que sea una elección popular entre los desarrolladores para construir aplicaciones web robustas y dinámicas.

PHP se destaca por su versatilidad y compatibilidad con diversos sistemas operativos y servidores web. Además de su capacidad para generar contenido dinámico, PHP ofrece una amplia gama de funciones predefinidas y una comunidad activa que contribuye con extensiones y bibliotecas, lo que lo convierte en una herramienta poderosa para la construcción de aplicaciones web escalables y eficientes.

### **MYSQL:**

MySQL es un sistema de gestión de bases de datos relacional (RDBMS) de código abierto ampliamente utilizado en el desarrollo de aplicaciones web y empresariales. Desarrollado por Oracle Corporation, MySQL ofrece una plataforma confiable y eficiente para almacenar y recuperar datos de manera organizada. Su capacidad para gestionar grandes cantidades de información, combinada con un rendimiento sólido, lo convierte en una opción popular para aplicaciones que requieren almacenamiento y recuperación de datos eficientes.

Con una estructura basada en tablas, MySQL utiliza el lenguaje SQL (Structured Query Language) para realizar consultas y manipular datos. Su naturaleza multiplataforma y su compatibilidad con varios sistemas operativos hacen que sea fácil de integrar en entornos diversos. Además, MySQL cuenta con una comunidad activa que contribuye al desarrollo continuo y mejora de este sistema, lo que lo posiciona como una solución robusta y confiable para la gestión de bases de datos en una variedad de contextos.

XAMPP:

XAMPP es un entorno de desarrollo de código abierto que simplifica la configuración y gestión de servidores web en entornos locales. La palabra "XAMPP" proviene de las iniciales de los componentes principales que incluye: Apache, MySQL, PHP y Perl. Este conjunto de software permite a los desarrolladores crear un servidor web local en sus computadoras, lo que facilita la prueba y el desarrollo de aplicaciones web antes de ser desplegadas en un entorno de producción.

XAMPP es compatible con múltiples sistemas operativos, como Windows, Linux y macOS, lo que lo convierte en una herramienta versátil para desarrolladores de diferentes plataformas. Además de los componentes mencionados, XAMPP también incluye otros servicios como phpMyAdmin para gestionar bases de datos MySQL de manera gráfica. En resumen, XAMPP proporciona un entorno de desarrollo integrado que simplifica la creación y prueba de aplicaciones web de manera local.

## VISUAL STUDIO CODE (VS Code)

Es un editor de código fuente desarrollado por Microsoft que se ha convertido en una herramienta popular entre los desarrolladores. Aunque es ligero y de carga rápida, VS Code ofrece una amplia gama de características poderosas que incluyen resaltado de sintaxis, autocompletado inteligente, depuración integrada y control de versiones. Su diseño modular permite a los usuarios personalizar su entorno de desarrollo mediante la instalación de extensiones que admiten una variedad de lenguajes de programación y frameworks.

Además de su capacidad para trabajar con una variedad de tecnologías, Visual Studio Code se destaca por su interfaz de usuario intuitiva y su enfoque en la eficiencia del flujo de trabajo del desarrollador. Es multiplataforma y está disponible para Windows, macOS y Linux, lo que facilita su adopción en diferentes entornos de desarrollo. En resumen, Visual Studio Code proporciona un entorno de desarrollo altamente funcional y adaptable, adecuado para proyectos de diferentes tamaños y complejidades.

## FRAMEWORK

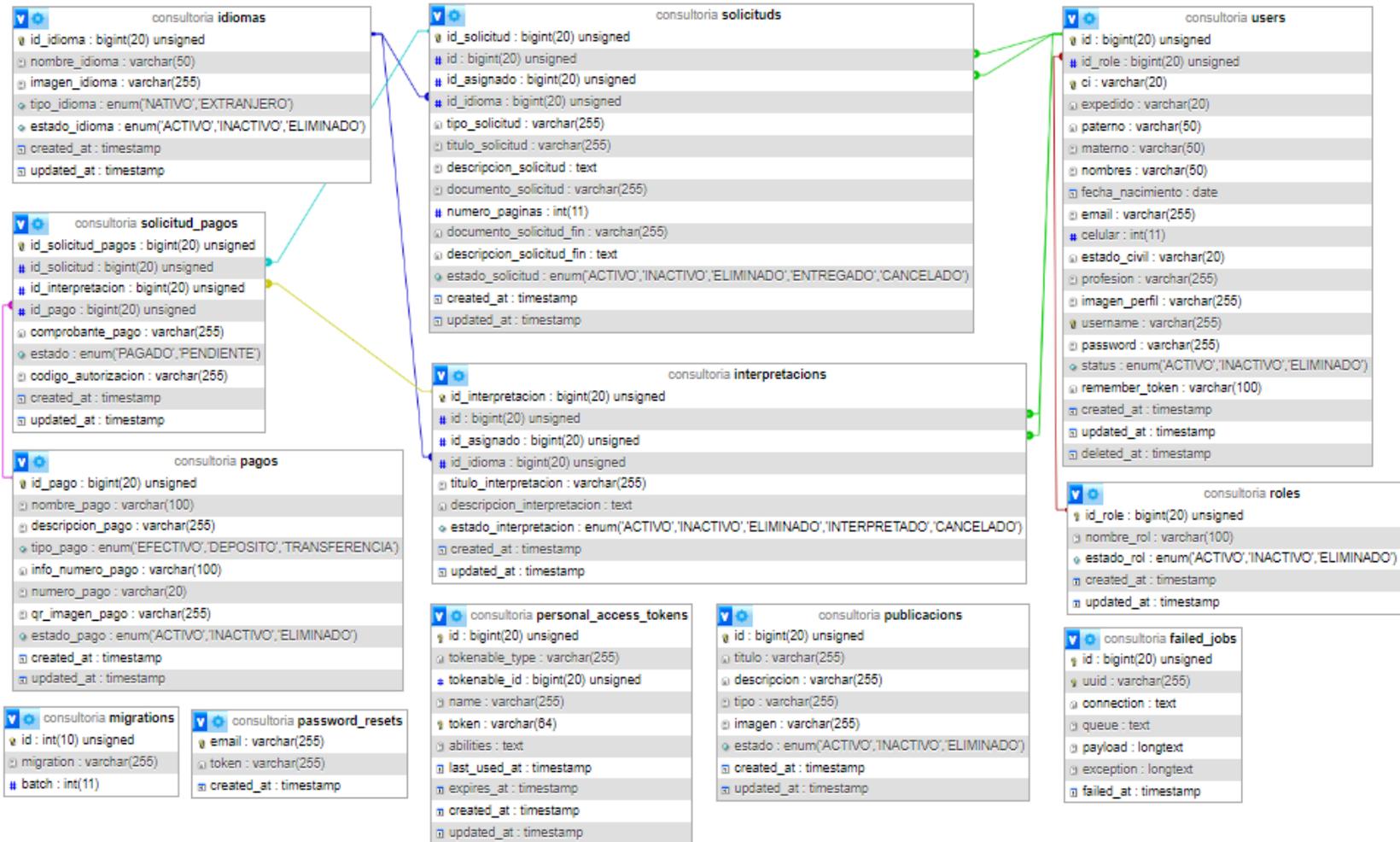
Laravel es un moderno y robusto framework de desarrollo web en PHP que sigue el patrón de arquitectura MVC (Modelo-Vista-Controlador). Diseñado para facilitar y agilizar el proceso de desarrollo de aplicaciones web, Laravel ofrece una sintaxis elegante y expresiva, junto con una variedad de características poderosas. Entre ellas se incluyen un sistema de enrutamiento claro, un ORM llamado Eloquent para el manejo de bases de datos, plantillas Blade para la gestión de vistas, y una amplia gama de utilidades para tareas comunes.

Este framework fomenta la escritura de código limpio y la adopción de prácticas de desarrollo modernas, como la inyección de dependencias y el uso de Composer para la gestión de paquetes. Laravel también proporciona una serie de herramientas para la automatización de tareas y la gestión de pruebas, lo que contribuye a la creación de aplicaciones web eficientes y de alto rendimiento. Con una activa comunidad de desarrolladores y una documentación exhaustiva, Laravel se ha convertido en una opción popular para la construcción de aplicaciones web modernas y escalables.

## CONFIGURAR BASES DE DATOS

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=consultoria
DB_USERNAME=root
```

# MODELO ENTIDAD RELACIÓN



## DICCIONARIO DE DATOS DEL MODELO ENTIDAD RELACIÓN

A continuación, se presentará toda la información que requiere cada dato dentro de las tablas y sus respectivas descripciones, campos y características.

Nombre	Tipo	Cotejamiento
1	<b>id</b>	bigint(20)
2	<b>id_role</b>	bigint(20)
3	<b>ci</b>	varchar(20)
4	<b>expedido</b>	varchar(20)
5	<b>paterno</b>	varchar(50)
6	<b>materno</b>	varchar(50)
7	<b>nombres</b>	varchar(50)
8	<b>fecha_nacimiento</b>	Date
9	<b>email</b>	varchar(255)
10	<b>celular</b>	int(11)
11	<b>estado_civil</b>	varchar(20)
12	<b>profesion</b>	varchar(255)
13	<b>imagen_perfil</b>	varchar(255)
14	<b>username</b>	varchar(255)
15	<b>password</b>	varchar(255)
16	<b>status</b>	enum('ACTIVO', 'INACTIVO', 'ELIMINADO')
17	<b>remember_token</b>	varchar(100)
18	<b>created_at</b>	Timestamp
19	<b>updated_at</b>	Timestamp
20	<b>deleted_at</b>	Timestamp

## TABLA- USUARIOS

id	id_rol	ci	expedido	paterno	materno	nombres	fecha_nacimiento	email	celular	estado_civil	profesion	imagen_perfil
1	1	11111111	LP	PATERN O	MATERN O	ADMIN	01/12/2001	admin@gmail.com	1111111 1	CASADO/A	DEV	1698034737.png
2	2	22222222	LP	PATERN O	MATERN O	COORDINADO R	01/12/2001	coordinador@gmail.com	2222222 2	CASADO/A	Lic en Idiomas	
3	3	33333333	LP	PATERN O	MATERN O	TRADUCTOR	01/12/2001	traductor@gmail.com	3333333 3	CASADO/A	Lic en Idiomas	
4	4	44444444	LP	PATERN O	MATERN O	INTERPRETE	01/12/2001	interprete@gmail.com	4444444 4	CASADO/A	Lic en Idiomas	
5	5	55555555	LP	PATERN O	MATERN O	USUARIO	01/12/2001	usuario@gmail.com	5555555 5	CASADO/A	Lic en Idiomas	
6	5	12345678	LP	PATERN O	MATERN O	PRUEBA PRUEBA	01/01/2000	prueba@gmail.com	1234567 8	SOLTERO/ A	PRUEBA	NULL
7	2	11223344	LP	PATERN O	MATERN O	PRUEBA DOS	10/10/2000	prueba2@gmail.com	1122334 4	SOLTERO/ A	PRUEBA	NULL
8	4	11122233 3	PA	PATERN O	MATERN O	PRUEBA TRES	01/01/1999	prueba3@gmail.com	1112223 3	SOLTERO/ A	INGENIERI O DE SISTEMAS	NULL