

UNIVERSIDAD SAN PEDRO
FACULTAD DE INGENIERÍA
PROGRAMA DE ESTUDIOS DE INGENIERÍA
INFORMÁTICA Y DE SISTEMAS



**Aplicación web de gestión del alquiler de equipos de
construcción civil para la empresa Roosveth Servicios
Generales E.I.R.L, Cajamarca, 2022**

Tesis para optar el Título Profesional de Ingeniero en Informática
y de Sistemas

Autor

Jiménez Hoyos Esmilton

ORCID: 0000-0002-6513-6676

Asesor

Villarreal Torres, Henry Oswaldo

ORCID: 0000-0002-5989-4534

Cajamarca – Perú

2023

Índice General

Índice General.....	i
Índice de Tablas	ii
Índice da Figuras.....	iv
Palabras claves	vi
Constancia de Originalidad.....	vii
Título.....	viii
Resumen.....	ix
Abstract.....	x
Introducción.....	1
Metodología.....	19
Resultados.....	21
Análisis y Discusión	66
Conclusiones.....	68
Recomendaciones	69
Referencias Bibliográficas.....	70
Anexos y Apéndice.....	74

Índice de Tablas

Tabla 1 Técnicas e instrumentos empleados en el estudio	19
Tabla 2. Roles SCRUM	22
Tabla 3. Detallado del historial de usuarios.....	24
Tabla 4. Pila de productos (Product Backlog)	28
Tabla 5. Detallado del Sprint 01	29
Tabla 6. Detallado del Sprint 02	30
Tabla 7. Detalle del Sprint 03	31
Tabla 8. Detalle del Sprint 04	32
Tabla 9. Detalle del Sprint 05	33
Tabla 10. Detallado del Sprint 06	34
Tabla 11. Denominación de las tablas de negocio	36
Tabla 12. Descripción de la Tabla Alquiler	36
Tabla 13. Descripción de la Tabla AlquilerDetalle	37
Tabla 14. Descripción de la tabla Equipo	38
Tabla 15. Descripción de las Tablas de Auditoria	39
Tabla 16. Descripción de la tabla AspNetUsers	40
Tabla 17. Descripción de la Tabla AspNetRoles	41
Tabla 18. Descripción de la tabla AspNetUserRoles.....	42
Tabla 19. Descripción de la tabla AspNetUserLogins.....	42
Tabla 20. Descripción del caso de prueba 1	61
Tabla 21. Descripción del caso de prueba 2	62
Tabla 22. Descripción del caso de prueba 4	63

Tabla 23. Descripción del caso de prueba 6	64
Tabla 24. Descripción del caso de prueba 7	65
Tabla 25. Descripción del caso de prueba 8	65

Índice da Figuras

Figura 1 Diagrama de Base de Datos Relacional	35
Figura 2 Arquitectura común de aplicaciones web .NET 6	43
Figura 3. Estructura de Aplicación Web Roosveth.....	44
Figura 4.Pantallazo de la configuración de base de datos	48
Figura 5. Pantallazo de configuración para la seguridad y el registro de usuarios....	48
Figura 6. Pantallazo de configuración de cookies, inyección de servicios e interfaces.	49
Figura 7. Pantallazo de la clase controlador de la aplicación	49
Figura 8. Pantallazo de la clase lógica de la aplicación: EquipoBL.cs.....	50
Figura 9.pantallazo de acceso a datos: GenericEntityRepository.cs	51
Figura 10. Pantallazo de la Clase entity: Equipo.cs.....	52
Figura 11. Pantallazo de la Clase view model - CreatedAlquilerViewModel.cs.....	53
Figura 12. Diagrama de Arquitectura Azure de Aplicación en la Nube.....	54
Figura 13. Diagrama de Creación y Despliegue en Azure App Service.....	54
Figura 14. Pantallazo de registro de clientes	55
Figura 15. Pantallazo de login de cliente	56
Figura 16. Pantallazo de mantenimiento de datos personales del cliente	56
Figura 17. Pantallazo de mantenimiento de usuarios	57
Figura 18. Pantallazo de mantenimiento de roles de usuario	57
Figura 19. Pantallazo de pantalla principal - equipos disponibles.....	57
Figura 20. Pantallazo de registro de información de equipo nuevo.....	58
Figura 21. Pantallazo de mantenimiento de equipo	58

Figura 22. Pantallazo de administración de alquileres	59
Figura 23. Pantallazo de creación de nuevo alquiler	59
Figura 24. Pantallazo de resumen de alquiler crea	60
Figura 25. Pantallazo de Gestión de Alquiler	60

Palabras claves

Tema	: Aplicación web
Especialidad	: Ingeniería de Software

Keyword

Theme	: Web Application
Specialty	: Software Engineering

Línea de Investigación

Línea de investigación	: Ingeniería de software
Área	: Ingeniería, Tecnología
Subárea	: Ingeniería eléctrica, electrónica e informática
Disciplina	: Ingeniería de sistemas y comunicaciones

Constancia de Originalidad



CONSTANCIA DE ORIGINALIDAD

El que suscribe, Vicerrector de Investigación de la Universidad San Pedro:

HACE CONSTAR

Que, de la revisión del trabajo titulado "Aplicación web de gestión del alquiler de equipos de construcción civil para la empresa Roosevelt Servicios Generales E.I.R.L, Cajamarca, 2022" del (a) estudiante: JIMENEZ HOYOS ESMILTON, identificado(a) con Código N° 2820220034, se ha verificado un porcentaje de similitud del 23%, el cual se encuentra dentro del parámetro establecido por la Universidad San Pedro mediante resolución de Consejo Universitario N° 5037-2019-USP/CU para la obtención de grados y títulos académicos de pre y posgrado, así como proyectos de investigación anual Docente.

Se expide la presente constancia para los fines pertinentes.

Chimbote, 07 de septiembre de 2023

UNIVERSIDAD SAN PEDRO
VICERECTORADO DE INVESTIGACIÓN

Dr. JAVIER MARTÍNEZ CARRIÓN
VICERECTOR



NOTA: Este documento carece de valor si no tiene adjunta el reporte del Software TURNITIN.

Título

Aplicación web de gestión del alquiler de equipos de construcción civil para la empresa Roosveth Servicios Generales E.I.R.L, Cajamarca 2022.

Resumen

La investigación aborda el desarrollo de una aplicación mediante la utilización de tecnología web para gestionar en una empresa el alquiler de equipos de construcción civil, cuyo objetivo principal fue desarrollar una aplicación web para mejorar la gestión de alquileres de los equipos de construcción civil con los que cuenta la empresa. El software se adapta a lo largo del proyecto y debe soportar cambios constantes, se implementó utilizando los servicios de Microsoft Azure Cloud, ya que estos servicios en la nube reducen los costos y tiempos de desarrollo y optimiza la entrega continua del servicio como software al cliente.

Finalmente, esta investigación tuvo como resultados positivos entre el proceso de negocio de alquiler de equipos y la aplicación web que fue el desarrollo de un software de calidad para los Stakeholders (clientes, trabajadores, proveedores), además de un alto impacto en la operatividad de consulta y organización de la información de los equipos, alquileres y clientes de la empresa.

Abstract

The research deals with the development of an application through the use of web technology to manage the rental of civil construction equipment in a company, whose main objective was to develop a web application to improve the rental management of civil construction equipment with which the company account. The software adapts throughout the project and must support constant changes, it was implemented using Microsoft Azure Cloud services, since these cloud services reduce costs and development times and optimize the continuous delivery of the service as software to the client. Finally, this research resulted in positive indications between the equipment rental business process and the web application that was the development of quality software for Stakeholders (customers, workers, suppliers), in addition to a high impact on the operation consultation and organization of the information of the equipment, rentals and clients of the company.

Introducción

Para el estudio se realizó una indagación de todas aquellas investigaciones realizados a las variables de estudio, posterior se seleccionaron las más adecuadas para considerarlas como antecedentes clasificados según su origen. Para lo cual, se presentan las investigaciones en el ámbito internacional:

Díaz (2021), en Colombia realizó una investigación donde elaboró el diseño e implementación de un sistema de gestión para alquiler y soporte de equipos para la empresa SINAPPTIC SAS, la metodología fue experimental, los resultados evidencian que, para gestionar de forma eficiente la gestión del alquiler es necesario una aplicación web, en donde se utilice la arquitectura Modelo Vista Controlador, UML y CodeIgniter, además de las tecnologías PHP, HTML5 y MySql. Concluye que, el sistema web para la empresa Sinapptic SAS automatiza en su totalidad el servicio de alquiler de equipos, en la implementación se puede observar módulos de registro, visualización de equipos, clientes, reportes de entrega, devolución de equipos y reporte general de equipos.

Chávez (2021) las empresas de esta industria deben utilizar equipo especializado, ya sea propio o de proveedores externos, para la realización de Proyectos de Construcción. Este informe de experiencia profesional detalla una iniciativa para agilizar el procedimiento de alquiler de equipos de una empresa constructora. El problema que encontró la constructora en 2010 fue que no daba prioridad a los equipos propios sobre los equipos externos al ingresar a los proyectos y que la contratación de equipos externos no era el mejor curso de acción, lo que generaba costos que impactaban en los proyectos. ' rendimiento financiero. Se sugirió una instalación centralizada de alquiler de equipos como solución a este problema, ya que garantiza que los equipos propios se unan primero a los proyectos y que se priorice la contratación de equipos de terceros.

Continuando, Cuvi y Rosero (2020), realizaron una investigación teniendo como objetivo principal realizar la implementación de una herramienta web para el control de las actividades de alquiler de maquinaria pesada de la empresa ANDESNEGSA; la metodología utilizada fue no experimental. Los resultados muestran que, el sistema web fue desarrollado bajo la metodología Scrum, con las herramientas Python y PostgreSQL, el sistema permitió automatizar los procesos de la gestión de la información de los equipos. Se concluye que, el sistema web permitió automatizar los procedimientos de alquiler de maquinaria, se utilizaron los módulos de gestión de maquinaria/ equipos, usuarios, alquiler, proveedores, clientes e informes; el servidor asignado fue Apache con Putty, para automatizar los query.

Vera (2019), en su estudio realizó el desarrollo e implementación de un sistema web para el control de inventario y alquiler de maquinarias de la empresa Megarent S.A, la metodología fue no experimental. Los resultados evidencian que, fue desarrollado e implementado PHP7, con base de datos de MariaDB, con JQuery y el framework Bootstrap. La arquitectura del aplicativo fue desarrollada en Codeigniter con el patrón M.V.C. Se concluye que, con el desarrollo e implementación de un sistema web se ha optimizado los procesos de alquiler de equipos, gestión de maquinaria, entre otros, se obtiene la información de manera más oportuna, en donde los empleados obtienen mejor productividad en la empresa.

Respecto al ámbito local, tenemos a Ramírez e Ynga (2019), en Cajamarca, investigaron acerca de la Implementación de una aplicación web/móvil para la gestión de pedidos y cotizaciones de la empresa Eurotubo S.A.C. Los resultados reportan que, se implementó una aplicación web/móvil para la gestión de pedidos, con la metodología RUP, con la plataforma J2me, con Java y PHP, con base de datos MySQL, con el propósito de efectuar de mejor manera la administración de pedidos y cotizaciones, así como también la gestión de pedidos, se ha optimizado los procesos, se ha disminuido el tiempo de registro. Se concluye que, se logró demostrar que la aplicación web móvil mejora la gestión de pedidos en la empresa, es viable.

Asimismo, en el ámbito nacional mencionar las investigaciones como de Salhuana (2019), en Lima, realizó una Aplicación web para la gestión de equipos en Technical Training Center. Los resultados refieren que, se realizó una aplicación web para la gestión de equipos, se ha utilizado el Proceso Racional Unificado para la calidad de la producción del software, la base de datos fue MySQL, el servidor de aplicaciones fue la tecnología Backend por PHP/Laravel; con el sistema se ha optimizado el proceso de compras, control de stock de productos, detalle de ventas, clientes y productos, gestión de equipos para brindar un mejor servicio y atención al cliente, gestión de direcciones de envío y gestión del personal administrativo. Se concluye que, la aplicación web para la gestión de equipos fue un éxito, influye satisfactoriamente en el tiempo de atención al cliente, se ha cumplido con los requerimientos por parte del usuario, con el fin de la satisfacción del cliente.

Gallego (2019) en Piura, realizó una implementación de un sistema para mejorar la gestión de inventarios y la programación de mantenimiento de los equipos en la empresa Pomahuaca – Jaén. Los resultados muestran que, la aplicación web tuvo una metodología del Rational Unified Process, el análisis fue realizado con UML, la codificación se realizó con NetBeans IDE 8.0.1, la base de datos se realizó con SQL Server 2016, se utilizó el servidor web de Glassfishy Server, con Framework Ext JS – Javascript. La aplicación web permitió gestionar el inventario de gestión de equipos, mantenimiento, control, así mismo se permite realizar reportes para la toma de decisiones de la alta gerencia. Se concluye que, se desarrolló e implemento con satisfacción la aplicación web, cubriendo todos los requisitos del usuario, reduciendo tiempo en la solicitud y asignación de equipos.

Jimeno y Visitación (2019), en Lima, investigaron el diseño e implementación de un sistema web para la gestión del flujo de información en el taller automotriz Autoservicios Aguilar. Los resultados evidencian un diseño e implementación de un sistema web, se utilizó como base de datos MySQL, como arquitectura se utilizó el Modelo Vista Controlador, en el cual se registra personal, reservación de citas, control de inventarios de equipos; con el sistema web se permite el flujo de la información y se optimiza los servicios y procesos. Se concluye que, con el sistema web se logra

optimizar los tiempos de atención al cliente, el usuario tiene una plataforma web de fácil acceso y entendimiento que gestiona de manera eficiente la información.

Huamán y Huayanca (2017), en Lima, indagaron acerca del Desarrollo e implementación de un sistema de información para mejorar los procesos de gestión de equipos en la empresa Humaju. Los resultados muestran que, se desarrolló un sistema web, fue implementado con la metodología del Proceso Unificado Ágil, se fue desarrollado en Visual Studio 210 y la base de datos fue SQL 2012, el sistema de información permitió facilitar los procesos de compras y ventas, y optimizó el tiempo de atención a los clientes en la gestión de equipos. Se concluye que, con el sistema de información se cumplen los objetivos de la empresa en la gestión de equipos, con la implementación del sistema e evidencia se reduce y automatiza los tiempos en la atención del cliente, procesos, sin perder la información relevante de la empresa.

Takamura (2021) en su estudios desarrollada para proponer la Implementación de un Sistema Web de Gestión Administrativa de la Empresa Constructora Kamage Contratistas Generales S.R.L - Piura; 2021, con el fin de mejorar la optimización de los procesos y servicios. Para lo cual aplico una investigación cuantitativo, nivel descriptivo, tipo no experimental y corte transversal. Asimismo, conto con una población de 22 trabajadores vinculados a las áreas administrativas. El resultado obtenido sobre el primer nivel de satisfacción respectó al sistema actual es de 62% de los trabajadores encuestados NO se sienten satisfechos con el actual manejo de los procesos internos en cuanto a documentación, en el segundo nivel de satisfacción sobre la gestión administrativa actual, el 98% de los trabajadores SI aceptan la propuesta de implementación de un sistema web de gestión administrativa. Finalmente, la investigación queda debidamente justificada en la necesidad de una Propuesta de Implementación de un Sistema Web de Gestión Administrativa para la Empresa Constructora Kamage Contratistas Generales S.R.L - Piura; 2021.

La fundamentación del presente estudio se basa en los principios y teorías para el desarrollo, mantenimiento y pruebas de los sistemas informáticos; contempla la teoría de sistemas, la teoría de la información, la teoría de la computación, la teoría de la toma de decisiones, la teoría de la organización y la teoría de redes, los que ha sido

utilizados para desarrollar modelos y métodos para el diseño, desarrollo, prueba y despliegue de sistemas informáticos. También se utilizan para el diseño de arquitecturas de software, la selección de herramientas de software, el diseño de interfaces de usuario y la seguridad de los sistemas; además, se puede utilizar para la gestión de proyectos de software, la gestión de recursos de software y el aprendizaje automático.:

Una teoría formal de la totalidad y la complejidad de la variedad holística, la teoría general de sistemas (abreviada TGS) intenta investigar las características universales de cualquier tipo de sistema. Aproximadamente diez años antes de la cibernética de Wiener, Ludwig Von Bertalanffy (1901–1972) desarrolló las ideas centrales de esta teoría. Establece los conceptos de sistema e isomorfismo como objetos primarios de investigación señalando primero las deficiencias de las concepciones "clásicas" de la ciencia, que se basan en los paradigmas mecanicista y positivista. Además de estar fuertemente influido por las ideas de una filosofía organísmica, que reconoce las limitaciones de los modelos causales y deterministas y busca definir principios generales aplicables a todo tipo de sistema, sea cual sea el contexto, nació con una clara vocación de interdisciplinariedad.

Aunque el TGS se propuso por primera vez en la década de 1920, no fue hasta mediados de la década de 1960 que comenzó a ganar relevancia. Bertalanffy publicó Teoría general de los sistemas en 1968, y Mesarovic y Schwarz publicaron sus trabajos fundacionales entre 1965 y 1971. Otros autores vinculados a la teoría de los sistemas universales incluyen a W. Boguslaw (*The New Utopians*, 1965), Anatol Rapoport (quien la usa en estudios sociológicos). research), y W. Churchman (*Philosophy and Science of Systems*, 1968). Posteriormente, Erwin Laszlo ha sido desde entonces el partidario más ferviente de la TGS, particularmente entre los de la Academia de Viena, un centro para el estudio de sistemas). De hecho, varios autores creían que el TGS era un subproducto de la cibernética (Wiener, de la década de 1930) y de la teoría de la información (Shannon y Weaver, 1949), a pesar de que el TGS ya había sido descubierto. La difusión de teorías sistémicas parciales, como la teoría de redes, la teoría de compartimentos, la teoría de autómatas (Turing, 1936), la teoría de juegos

(von Neumann y Morgenstern, 1947), o la propia teoría, según Bertalanffy Teoría de colas, teoría de la decisión, cibernética, y la teoría de la información son todos subconjuntos de una sola teoría genérica de sistemas. Por ejemplo, cree que los sistemas cibernéticos son un tipo específico de sistema autoorganizado y que la cibernética (la teoría de los mecanismos de control basada en las nociones de información y retroalimentación) es solo un subconjunto de TGS.

El trabajo realizado por Boulding (1966) sobre la teoría de sistemas aplicada a la economía tuvo un gran impacto en el desarrollo de una teoría general de sistemas. Boulding desarrolló un modelo que permitía analizar la economía y los sistemas sociales como un sistema general, integrando los elementos económicos, sociales, políticos y psicológicos. Su trabajo también se centró en la importancia de la comunicación, la interdependencia y la retroalimentación en los sistemas. Además, también desarrolló una teoría de la jerarquía para explicar la forma en que los sistemas pueden estar organizados, contribuyendo significativamente a la evolución y comprensión de la teoría de sistemas.

Para Rapoport (1909) fue uno de los primeros en aplicar la teoría de sistemas a la biología, contribuyó de forma relevante al desarrollo de la teoría de sistemas y, en particular, a la comprensión de los procesos de interacción entre los organismos y su entorno, desarrolló la teoría general de sistemas. En el caso de Shannon (1948), sostuvo la teoría de la información, que define como un medio para reducir la incertidumbre, también introduce el concepto de entropía, que mide el grado de incertidumbre de un sistema. Weaver (1949) discute la relación entre la información y la complejidad, resaltando la importancia de la información en la comprensión de los sistemas complejos; complementado por Wiener (1948) quien presenta la teoría de la información y la cibernética, estableciendo una relación entre la información, el control y la comunicación, y propone que los sistemas informáticos pueden ser utilizados para controlar los sistemas físicos.

Así mismo, Turing (1936) propone la denominada máquina de Turing, un equipo abstracto que se puede usarse para ejecutar programas, esta máquina es la base de la tecnología moderna de computación, dando lugar a Von Neumann (1945) quien

propone la arquitectura de computadora basada en la máquina de Turing, esta arquitectura es la base de la mayoría de las computadoras modernas. Tanto, las investigaciones de Cook (1971) y Valiant (1979) demuestran que hay algunos problemas que son demasiado difíciles para ser resueltos por una computadora, esta demostración es la base de la teoría de la complejidad computacional. Finalmente, Baran (1954) propone una arquitectura de redes que permita la comunicación entre computadoras a través de una red de computadoras, esta arquitectura fue la base de la tecnología moderna de redes; seguido posteriormente por Davies (1965) proponiendo una arquitectura de red basada en paquetes para permitir la comunicación entre las computadoras, complementándose con el estudio realizado por Kleinrock (1976) para proponer un modelo para estudiar los sistemas de colas y su comportamiento.

Para Boehm (1981) la ingeniería del software es una disciplina que se dedica al estudio, diseño, desarrollo, mantenimiento y mejora de software, se fundamenta en una variedad de disciplinas científicas, como la matemática, la lógica, la ciencia de la computación, la psicología y la ingeniería, proporcionan una base sólida para abordar los problemas complejos implicados en el desarrollo de un producto de software.

Según DeMarco (1979) se centra en el análisis estructurado y la especificación de sistemas, y ofrece una guía útil para desarrollar software de alta calidad. El libro explica cómo crear un sistema de análisis estructurado, cómo diseñar una especificación de sistema y cómo desarrollar un modelo de análisis y especificación. Reinertsen (1997) la obra se consolida en la productividad y los principios de desarrollo de productos y ofrece una guía útil para mejorar la eficiencia en el desarrollo de software; explica cómo identificar los problemas de productividad, cómo desarrollar un flujo de trabajo óptimo y cómo mejorar la eficiencia del desarrollo de productos. Fowler (1999) aborda los conceptos clave de la programación orientada a objetos y ofrece una guía útil para el desarrollo de software, explica cómo identificar y refactorizar el código existente, cómo mejorar la estructura del código y cómo crear código limpio y fácil de mantener. Fowler (2002) aborda los patrones de arquitectura de aplicaciones de empresa y ofrece una guía útil para el desarrollo de software, explica cómo diseñar una arquitectura de aplicación de empresa, cómo organizar el código

para lograr una mayor escalabilidad y cómo integrar múltiples tecnologías en una solución única.

Sobre los lenguajes de programación se tiene teorías se tiene: Teoría de la sintaxis: se centra en la estructura y la semántica de los lenguajes de programación, esta teoría se aplica para diseñar lenguajes de programación que sean fáciles de entender y usar. Teoría de la computabilidad: Esta teoría se centra en el análisis de los lenguajes de programación para determinar qué problemas pueden ser resueltos por una computadora. Teoría de la eficiencia: definida en el análisis de los lenguajes de programación para determinar el tiempo y la memoria requeridos para ejecutar un programa. Teoría de la programación: se centra en el análisis de los lenguajes de programación para determinar la forma más eficiente de diseñar y codificar un programa (Dijkstra, 1976).

Las teorías de bases de datos se fundamentan en el estudio de bases de datos relacionales, los principios incluyen el uso de modelos de datos lógicos para representar la información y los datos, el uso de lenguajes de consulta para recuperar y manipular los datos, la administración de transacciones y el diseño de bases de datos. Estos principios de la teoría de bases de datos se han desarrollado y mejorado a lo largo de los años, permitiendo el diseño de bases de datos que pueden proporcionar un acceso rápido y seguro a los datos, han permitido el desarrollo de sistemas de bases de datos modernos que pueden manipular grandes cantidades de datos de manera eficiente y segura; consta de 12 principios de la teoría de bases de datos relacionales, que se han convertido en la mayoría de los sistemas de bases de datos modernos; estos principios incluyen el uso de un modelo de datos lógico para representar la información, el uso de lenguajes de consulta para recuperar y manipular los datos, la administración de transacciones y el diseño de bases de datos, constituyendo el aporte más significativo (Codd, 1970). Asimismo, la abstracción de los datos, ha permitido crear sistemas más flexibles y escalables (Date, 1975).

La arquitectura de software se basa en varias teorías y principios fundamentales, entre estos principios se encuentran el modularidad, la abstracción, la reutilización, la seguridad, la escalabilidad y la robustez. Estos principios están basados en la teoría de diseño, la cual se centra en la creación de una estructura estable y escalable para el software; se ha desarrollado a lo largo de los años y se ha vuelto más refinada con el tiempo. Explora los principios y conceptos básicos de la arquitectura de software, con un enfoque en la modulación, la abstracción, la reutilización y el diseño orientado a objetos. (Martin, 2017). La mejora de la calidad del código existente mediante el uso de diversos patrones y técnicas, proporcionando una guía paso a paso para la refactorización de código, lo que ayuda a los desarrolladores a crear un software de calidad (Fowler, 1999).

El uso de patrones de diseño para crear software reutilizable y escalable. otorgando una colección de patrones y técnicas utilizados para el diseño de software orientado a objetos (Gamma et al., 1994). Una visión general de la arquitectura orientada a objetos, junto con una guía paso a paso para el análisis y el diseño de software, también se incluyen ejemplos y herramientas para ayudar a los desarrolladores a comprender mejor la arquitectura orientada a objetos incluyendo los principios y conceptos básicos de la misma; además de ofrecer una guía para la creación de arquitecturas de software robustas y escalables. (Booch, Rumbaugh & Jacobson, 2007; Clements, Kazman & Klein, 1998).

Además, se tiene la Teoría General de la Administración donde independientemente de nuestro nivel de conocimientos en la escuela, la administración es una actividad humana que todos realizamos. Se diferencia de las matemáticas, la biología o la antropología en que se puede vivir bien sin ser un estudiante asiduo de estas u otras ciencias, que no es la caso con la administración porque siempre debemos estar gestionando todo. Desde que nos despertamos hasta que nos vamos a dormir, debemos estar maximizando nuestros recursos, empezando por el tiempo. Hay muchos ejemplos de doctores en administración que se destacan por administrar mal incluso a su propia persona, así como ejemplos proverbiales de doctores en administración o profesionales de la administración que brillan por su conocimiento administrativo que los hace destacar. Cada una de las personas es administradora con resultados favorables o desfavorables de su ejercicio administrativo.

Sin embargo, cabe señalar que la regla es que los que estudian administración profesionalmente son administradores excepcionales. En el otro extremo están las personas que nunca han estudiado administración, pero superan con creces las calificaciones de un buen administrador. Hagamos el siguiente caso: Distinguidos hombres de negocios, o buenos administradores, como Tom Watson (IBM), Ray Kroc (McDonald's), Robert Woodruff (Coca-Cola), Henry Ford (Ford Motor Company), los hermanos Warner (Time Warner), o Bill Gates (Microsoft), sin haber recibido una educación universitaria, crearon verdaderos imperios empresariales debido a sus habilidades administrativas inherentes.

La mayoría de ellos ya se han extinguido, y los negocios que construyeron con el tiempo han progresado más que cuando aún estaban vivos. Solo eche un vistazo a las capacidades de estas organizaciones, sus grupos de gestión y los métodos de gestión que emplean a diario, incluida la investigación de operaciones, los diseños de estructuras organizativas, el ejercicio del poder, la toma de decisiones, la teoría del liderazgo, etc. Varios Las teorías que representan avances genuinos en el desarrollo de la teoría general de la administración incluyen la teoría del conocimiento o la teoría del aprendizaje.

Cuando los estudiantes de administración ingresan inicialmente a este campo, se les presenta la amplia teoría de la administración. Se han incluido dichos célebres que hacen referencia al tema, parábolas y anécdotas con una lección, el marco histórico en el que se desarrolló una determinada teoría y cápsulas del tiempo culturales o técnicas que recuerdan información importante. También se han introducido estudios de casos simples y una recomendación de lectura que viene con un aprendizaje completo para ayudar al lector a comprender lo que se ha presentado.

Sobre los Sistemas de Información, los autores Pantaleo y Rinaudo (2015) refieren que, los sistemas de información han sido una de las áreas de estudio más populares en la disciplina de organización empresarial en los últimos años. El entorno en el que las empresas hacen crecer sus operaciones se vuelve más complejo. La información se ha convertido en un componente crucial tanto para la gestión como para la supervivencia y expansión de la organización empresarial como consecuencia de la creciente globalización, el proceso de internacionalización de la empresa, el aumento de la competencia en los mercados de bienes y servicios, el rápido desarrollo de tecnologías de la información, el aumento de la incertidumbre ambiental y el acortamiento de los ciclos de vida de los productos. Si la tierra, el trabajo y el capital eran los recursos primarios examinados anteriormente, el conocimiento ahora parece ser otro contribuyente crucial para el valor en los negocios.

Lapiedra, Devece y Guiral (2011) mencionan que, también existe un proceso de retroalimentación, o “feedback”, en el que se debe determinar si la información obtenida es adecuada a lo esperado. Todo sistema de información utiliza como materia prima los datos, los cuales almacena, procesa y transforma para obtener como resultado final la información, la cual será suministrada a los distintos usuarios del sistema.

Por otra parte, Oz (2008), indica que, los desarrolladores de sistemas de información deben estar familiarizados con las tecnologías de la información disponibles y utilizables dentro de la organización, así como con la forma de organizarlas. Para ello, deberán comprender la estrategia y el tipo de organización para determinar las demandas de información y adquirir las herramientas necesarias para la creación del sistema de información.

Respecto a la Ingeniería de Software, Pressman (2010) refiere que el hardware cada vez más potente, software de última generación (lenguajes orientados a objetos, lenguajes deductivos, arquitecturas interpretadas, intermedias, multiplataforma, orientadas a servicios), desarrollo de modelos (CMMi, ITIL) y metodologías ágiles (XP, Scrum) son los que dan forma al escenario del software de desarrollo que se estableció en los albores del nuevo siglo (2000). En estos días, es fundamental garantizar la alta calidad de los procesos de desarrollo de software y productos terminados.

Gómez, Cervantes y González (2019), menciona que, la evolución del desarrollo de software comenzó con la creación de programas especializados utilizando herramientas exclusivas y un pequeño grupo de programadores, lo que condujo al uso de tecnologías estándar por parte de los programadores que trabajan en equipos con varios miembros para crear programas que utilizan empresas y/o comunidades. de personas en varios campos hoy. Esta evolución resultó de la sistematización del conocimiento en este campo tecnológico, que luego fluyó sobre sí mismo.

Se justifica la investigación en la práctica porque se evidencia la necesidad de evaluar una aplicación web de gestión, para el alquiler de equipos en una empresa cajamarquina.

La justificación teórica para el desarrollo de un sistema informático web implica entender la tecnología web y los fundamentos de la informática; significa entender los conceptos básicos de programación web, bases de datos, diseño web y protocolos de red; también se requiere conocimiento de lenguajes de programación web como HTML, CSS, JavaScript y otros. Mediante la creación de soluciones de software personalizadas, las empresas pueden adaptar su infraestructura tecnológica para satisfacer necesidades específicas y, al mismo tiempo, garantizar una mayor seguridad que la que podrían proporcionar los productos estándar.

La justificación metodológica, el desarrollo de un sistema informático web implica un proceso estructurado para asegurar el cumplimiento de los requisitos del sistema, donde se utiliza una metodología para el desarrollo de software, como el ciclo de vida en cascada o el desarrollo ágil. Estas metodologías permiten identificar y priorizar los requisitos del sistema, diseñar una arquitectura de software adecuada, desarrollando una interfaz de usuario intuitiva y controlar la calidad del software.

La justificación social, el desarrollo de un sistema informático web implica entender la naturaleza de la web y los usuarios de la aplicación; teniendo en cuenta la privacidad, seguridad, accesibilidad y usabilidad del sistema donde el desarrollador debe entender las necesidades y expectativas de los usuarios y crear una aplicación que sea fácil de usar y cumpla con los estándares de la industria estará basada en la motivación hacia el cliente interno, pues es un factor de éxito para toda empresa. Así mismo, brindará información a todo el sector empresarial que desee conocer acerca una aplicación web de gestión de alquiler.

Desde un punto de vista práctico, el desarrollo de un sistema informático web implica tener un conocimiento profundo de herramientas de desarrollo web, como servidores web, entornos de desarrollo, frameworks de desarrollo web y herramientas de pruebas. Esto permite al desarrollador crear una aplicación web robusta y escalable logrando mayor eficiencia y productividad; mejor servicio al cliente; medidas de

seguridad mejoradas; mejores capacidades de gestión de datos; y ahorro de costos en términos de costos de mano de obra asociados con los procesos manuales.

La problemática sobre el desarrollo de software a nivel mundial presenta diversos desafíos y problemas, uno de los más relevantes es la complejidad debido al constante cambio en los procesos de negocios y la necesidad de mantenerlos actualizados y seguros, lo que se ve afectado por la falta de una comunicación efectiva entre los integrantes del equipo de desarrollo, generando retrasos y errores en el proyecto; otros de los problemas, es la presión por desplegar productos de software a producción rápidamente que puede llevar a la adopción de prácticas de desarrollo apresuradas y descuidadas, resultando en productos de baja calidad y con vulnerabilidades de seguridad. El desarrollo de software a nivel mundial requiere una atención en los aspectos de planificación, calidad y seguridad para garantizar el éxito a largo plazo de los proyectos de software.

En América Latina, el desarrollo de software para la gestión de alquiler de equipos de construcción civil presenta una serie de desafíos como la falta de estandarización de los procesos para la gestión de alquiler de equipos, lo que dificulta la creación de soluciones de software que sean aplicables a todo el mercado conllevando a soluciones informáticas personalizadas; razones por las que muchas empresas todavía dependen de procesos manuales para realizar sus operaciones, lo que puede resultar en errores y retrasos. Otro de los desafíos es la seguridad de los datos, considerado como un riesgo importante en la gestión de alquiler de equipos de construcción civil debido a la información sensible que se gestiona sobre los equipos, debiendo ser protegida contra robos y amenazas.

En el Perú, el desarrollo de software a la medida enfrenta diversos problemas que obstaculizan su crecimiento y eficiencia; como son los deficientes marcos regulatorios y la falta de recursos humanos nos lleva a una brecha importante entre la demanda de desarrolladores de software y la oferta de profesionales con habilidades y capacidades en tecnologías de información y comunicación. Por consiguiente, es un proceso iterativo, lo que significa que los requisitos del software pueden cambiar a medida que el proyecto avanza, haciéndose difícil gestionar el proyecto y asegurarse

que el software se entregue a tiempo y dentro del presupuesto. Además, los sistemas deben de estar actualizados y mantenerse de manera regular incrementado los recursos iniciales.

Ante la problemática expuesta, el estudio planteó como problema: ¿De qué manera el desarrollo de una aplicación web gestiona el alquiler de equipos de construcción civil para la empresa Roosveth Servicios Generales EIRL, Cajamarca, 2022?

En ese sentido, para desarrollar la aplicación web, se emplearon herramientas de desarrollo web, tomándose bases teóricas para conceptualización y operacionalización de las variables de estudios:

Sistema informático web, Barba (2014) señala que el desarrollo de un sitio web es la programación requerida para su creación. La parte del cliente y la parte del servidor están separadas en dos secciones que pueden o no estar conectadas. En la parte del cliente, discutiríamos JavaScript y DOM para la interacción del usuario, así como HTML y CSS, los códigos fundamentales para crear páginas web. Trabaja con código más complicado en el lado del servidor, como PHP, ASP.NET, JSP, etc. El back-end del sitio web, que el usuario no puede ver, se construye utilizando este código. Diseñar bases de datos y garantizar la seguridad online son sus dos objetivos principales. La programación cliente-servidor se utiliza cuando ambos lados se comunican. A través de esta comunicación, los usuarios pueden interactuar con el material de la base de datos y registrarse para los servicios.

.NET Framework, según la documentación oficial de Microsoft (2022) las aplicaciones que se dirigen a .NET Framework son administradas por el entorno de ejecución en tiempo de ejecución conocido como .NET Framework. Incluye una rica biblioteca de clases que permite a los programadores aprovechar un código sólido y confiable en todas las áreas clave del desarrollo de aplicaciones, así como Common Language Runtime, que ofrece administración de memoria y otras funciones del sistema.

.NET 6, según Microsoft (2022) al igual que su predecesor, .NET Core 3.1, .NET 6 es una versión LTS (soporte a largo plazo), lo que significa que el soporte continuará durante tres años. Si bien .NET 5 ya no será compatible a partir de mayo de 2022, esta versión 3.1 estará finalizada en diciembre de 2022. El lenguaje de programación orientado a objetos C# 10 actualizado se incluye con .NET 6. El uso de directivas de uso global, mejoras en estructuras tipos, un controlador para cadenas interpoladas y otros cambios se encuentran entre las adiciones que ofrece C# 10.

Razor Pages, según Microsoft (2022) la propuesta de ASP.NET Core MVC, conocida como "Razor Pages", tiene como objetivo facilitar la creación de contenido web que esté más en línea con el mundo moderno de Back al permitir la programación en C# para parte de la lógica y HTML, CSS, JavaScript, y notaciones personalizadas para la apariencia de las vistas.

Microsoft SQL Server 18, según Microsoft (2017) Transact-SQL (TSQL), una implementación del estándar ANSI del lenguaje SQL, es el lenguaje de desarrollo utilizado (a través de la línea de comandos o utilizando la interfaz gráfica de Management Studio), y se utiliza para editar y recuperar datos (DML), crear tablas, y establecer relaciones entre ellas (DDL). Se puede configurar para usar varias instancias en el mismo servidor físico, con el nombre del servidor apareciendo a menudo en la instalación inicial y diferentes nombres en las instalaciones posteriores (con una barra invertida entre el nombre del servidor y el nombre de la instalación).

Azure App Service, según la documentación oficial de Microsoft (2020), con Azure App Service, puede crear aplicaciones móviles y web listas para la empresa de forma más rápida y sencilla para cualquier plataforma o dispositivo, y luego implementarlas en una infraestructura de nube escalable y sin problemas. confiable. Es un servicio completamente administrado con capacidades integradas de escalado, parches de seguridad y mantenimiento.

Azure SQL Database, según la documentación oficial de Microsoft (2020), Creado para la nube, Azure SQL Database es un servicio de base de datos relacional completamente administrado y actualizado continuamente. Con la facilidad de uso y la adaptabilidad de una base de datos multimodelo que se adapta a la demanda, cree

su próxima aplicación. El motor de base de datos completamente administrado automatiza el aprovisionamiento, las actualizaciones y las copias de seguridad para que pueda concentrarse en desarrollar aplicaciones.

Cloud Computing, según Barba (2014) describe que la Cloud Computing o computación en la nube se refiere a las herramientas que ofrecen algunas empresas, públicas o comerciales, para la administración, gestión y uso compartido de datos en línea. Además, enfatizan que la virtualización es una de las bases para los inicios de la computación en la nube. Es un sistema que permite realizar virtualmente algunas tareas de computación e informática, como la simulación de computadoras virtuales dentro de computadoras reales, la simulación de procesos de negocios, la simulación de procedimientos médicos, etc.

Metodología Scrum, Para Schwaber y Sutherland (2020), Scrum es un proceso de desarrollo bastante sencillo que exige esfuerzo porque no se basa en adherirse rígidamente a un plan, sino en una adaptación constante a las especificaciones del desarrollo del proyecto. Aunque actualmente no existe un enfoque estándar, es crucial desde una perspectiva de software proporcionar los mecanismos adecuados para que la implementación de este tipo de aplicaciones satisfaga las necesidades tanto de los usuarios como de los clientes que contratan el desarrollo de estas aplicaciones. mejores prácticas reconocidas para la creación de aplicaciones web.

Para Schwaber y Sutherland (2017), la metodología está basada en la idea de que los proyectos deben ser flexibles y capaces de adaptarse a cambios inesperados. Scrum tiene varias fases, que se explican a continuación:

- Planificación del producto (Product Backlog Refinement): en esta fase, el propietario del producto (Product Owner) crea una lista de tareas que se deben completar; este listado, conocido como "Product Backlog", se prioriza en función de las necesidades del negocio.
- Planificación del Sprint (Sprint Planning): el equipo Scrum se reúne para planificar el siguiente sprint, que es un período de tiempo fijo, generalmente de 2 a 4 semanas, durante el cual se debe completar un conjunto de tareas; el equipo decide qué tareas del Backlog del Producto

se van a tratar en el siguiente sprint y cómo se van a realizar.

- Sprint: en esta fase, el equipo trabaja en las tareas seleccionadas, no se deben agregar nuevas tareas a la lista de tareas del sprint una vez que ha comenzado, para mantener el enfoque y la productividad.
- Reunión diaria de Scrum (Daily Scrum): esta es una breve reunión de 15 minutos en la que el equipo se informa mutuamente sobre el progreso y discute cualquier problema que pueda haber encontrado; el propósito es permitir una comunicación rápida y eficaz dentro del equipo.
- Revisión del Sprint (Sprint Review): al final del sprint, el equipo se reúne para revisar su trabajo y demostrar lo que ha logrado. Esto permite la inspección y adaptación del producto en desarrollo.
- Retrospectiva del Sprint (Sprint Retrospective): esta es una reunión en la que el equipo reflexiona sobre el sprint anterior y discute formas de mejorar su rendimiento en el futuro; se trata de un ejercicio de mejora continua.

La investigación tuvo como hipótesis la siguiente premisa: La implementación de una aplicación web mejora la gestión de alquiler de equipos de construcción civil en la empresa Roosveth Servicios Generales E.I.R.L.

Así mismo, se formuló el objetivo general para el estudio el cual es: Desarrollar una aplicación web de calidad para la gestión de alquiler de equipos de construcción civil asegurando la entrega continua del producto. Al mismo tiempo, se establecieron los siguientes objetivos específicos: Analizar el proceso actual de gestión de alquiler de equipos de la empresa, diseñar los módulos para el sistema de gestión de alquileres de equipos de construcción civil y construir una aplicación web de calidad utilizando los servicios Cloud Database y App Service de Microsoft Azure Cloud.

Metodología

El nivel de la investigación fue descriptivo, ya que permite obtener datos mediante instrumentos y técnicas de recolección de datos para luego describir la situación en la que se encuentra la realidad, con tipo de investigación aplicada porque busca la aplicación de conocimientos adquiridos durante el proceso de investigación. Así mismo, considerando como fueron tomados los datos y procesados para el desarrollo de la aplicación corresponde a un diseño no experimental.

La población estuvo definida por la cantidad de personas que tengan el acceso a la aplicación considerando los potenciales clientes de la empresa, así como la cantidad de trabajadores administrativos que laboran en la empresa.

La muestra estuvo constituida por un conjunto de potenciales clientes y trabajadores administrativos que accedan a la aplicación.

El estudio empleó las siguientes técnicas e instrumentos de investigación:

Tabla 1

Técnicas e instrumentos empleados en el estudio

Técnica	Instrumentos	Unidad de estudio (involucrado)
Entrevista	Ficha de Percepción de la Calidad del software.	- Personal de la empresa Roosveth Servicios Generales E.I.R. L
Observación	-Bloc de Apuntes. -Checklist.	- Potenciales clientes de la empresa Roosveth.

Respecto al procesamiento y análisis de la información desde estudio, esta será recolectada en tablas simples de Microsoft Excel posteriormente se procesó la información para comprender y entender los requerimientos funcionales y no funcionales del sistema.

Al haber finalizado con la recopilación de las bases teóricas que rigen la investigación, se comenzó por obtener y mapear los procesos comerciales de la empresa, mediante reuniones con los interesados (Anexo N°1).

Para el desarrollo del sistema informático se utilizó la metodología Scrum, es un marco popular utilizado en el desarrollo de software Agile, proporcionando un enfoque iterativo e incremental para la gestión de proyectos complejos, lo que permite a los equipos responder rápidamente a los requisitos cambiantes de los clientes y, al mismo tiempo, ofrecer productos de alta calidad. El proceso Scrum consta de cinco fases distintas: inicio, planificación y estimación, implementación, revisión y retrospectiva, lanzamiento.

Resultados

Se empleó la metodología ágil SCRUM para la creación del sistema informático basado en la web, ya que satisface las demandas tanto de los trabajadores como de las partes interesadas.

SCRUM (Desarrollo de productos) crea un marco para la creación del producto y sus entregables destacados porque se adapta al método de trabajo sugerido con requisitos cambiantes según la aprobación del instructor. También nos permite concentrarnos en las relaciones y el desempeño de los participantes sin estar obligados a realizar una documentación detallada del desarrollo, buscando así una mayor calidad en cada implementación de un requisito.

Eventos en SCRUM

Reunión para planificar los Sprint

Permite esclarecer los objetivos de cada sprint y además establecer que actividades serán necesarias para cumplir con dichos objetivos.

SCRUM diario

Son reuniones cortas que va a ayudar al equipo preguntarse las siguientes interrogantes:

- ¿Qué trabajo se realizó el día anterior?
- ¿Qué se tiene previsto realizar?
- ¿Qué se necesita? o ¿Qué cosa produce un impedimento para realizar una actividad?
- Revisión de Sprint

Se analiza y se verifica el incremento que se ha generado en un sprint y en el caso de sea necesario se debe organizar la pila del producto.

Retrospectiva del Sprint

Se cuestiona el: ¿qué ha ocurrido durante un sprint? para poder establecer un plan de mejora al siguiente sprint.

Tabla 2

Roles SCRUM

Personaje	Rol
- Manuel de la Cruz Jiménez Rojas	- Gerente General de Roosveth Servicios Generales E.I.R. L, interesado en el producto de software.
- Friedel Jiménez Hoyos	- Administrador de empresas, encargado de la especificación de los requerimientos del software.
- Esmilton Jiménez Hoyos	- Ingeniero de Sistemas, captura los requerimientos en historias de usuarios.
- Esmilton Jiménez Hoyos	- Scrum Master
- Esmilton Jiménez Hoyos	- Equipo de Desarrollo
- Personal de Roosveth Servicios Generales E.I.R. L	- Stakeholders principales
- Potenciales Clientes	- Stakeholders secundarios

Artefactos

Historias de Usuario

Las historias de usuario son una herramienta Scrum que explica una característica de un producto o pieza de software que el usuario encontrará valiosa. Describen la funcionalidad que se creará, pero no el proceso. Las notas post-it generalmente se usan para capturarlas y distribuirlas en todo el equipo.

Las historias de usuarios son algo más amplio que cabe en un post-it; no van tan lejos como para revelar la complejidad de una definición de necesidad. Estos son

los resultados de las discusiones iniciales entre los usuarios y los desarrolladores de la funcionalidad y sirven como base para estimar el tiempo necesario para implementar una nueva característica.

Tabla 3

Detallado del historial de usuarios

Enunciado de la historia					Criterios de Aceptación			
Identificador de la historia (ID)	Rol	Característica/ Funcionalidad	Resultado	# de escenario	Criterios de Aceptación (Título)	Contexto	Evento	Comportamiento esperado
1	Usuario/ Cliente potencial	Se necesita una interfaz para que el usuario pueda registrarse en la aplicación.	Amigable y fácil de entender.	1	Visualizar el formulario en orden	En caso de que el usuario necesite ayuda	Cuando el usuario necesita ayuda	Se mostrarán unas sugerencias que indicarán que debe ingresar en el formulario.
2	Usuario/ Cliente Potencial	Se necesita una interfaz para que el usuario pueda ingresar a la aplicación.	Amigable y fácil de entender	1	Visualizar el formulario en orden	En caso de que el usuario necesite ayuda	Cuando el usuario necesita ayuda	Se mostrarán unas sugerencias que indicarán que debe ingresar en el formulario.
3	Administrador	Se necesita una interfaz para que el administrador pueda administrar los usuarios de la aplicación	Una tabla de administración de usuarios, con 3 opciones	1	Administrar Roles	En caso de querer cambiar el rol de usuario.	Cuando necesite asignar roles al usuario	Asignar o quitar roles del usuario en cuestión.
				2	Administrar Usuario	En caso de actualizar datos de usuario	Cuando necesite actualizar datos	Actualizar datos del usuario correctamente
				3	Cambiar Contraseña	En caso de cambiar la clave del usuario.	Cuando necesite cambiar la clave	Cambiar la clave de acceso a la aplicación de un usuario.

Enunciado de la historia				Criterios de Aceptación				
Identificador de la historia (ID)	Rol	Característica/ Funcionalidad	Resultado	# de escenario	Criterios de Aceptación (Título)	Contexto	Evento	Comportamiento esperado
5	Administrador	Se necesita una interfaz para que el administrador pueda administrar los equipos de construcción civil.	Una interfaz con opciones para ver, crear, actualizar y eliminar Equipos.	1	Ver Equipos	En caso de querer Visualizar Equipos		Ver una lista de los equipos.
				2	Crear Equipo	En caso de querer Crear Equipo		Equipo nuevo creado
				3	Actualizar Equipo	En caso de querer Actualizar Equipo		Equipo Actualizado
				4	Eliminar Equipo	En caso de querer Eliminar Equipo		Equipo Eliminado
6	Administrador	Se necesita una interfaz para visualizar los alquileres de los equipos de construcción civil.	Una interfaz con opciones para ver, crear, actualizar y anular alquileres.	1	Ver Alquileres	En caso de querer Visualizar Alquileres		Ver una lista de los Alquileres.
				2	Crear Alquiler	En caso de querer Crear Alquiler		Alquiler nuevo creado

Enunciado de la historia				Criterios de Aceptación				
Identificador de la historia (ID)	Rol	Característica/ Funcionalidad	Resultado	# de escenario	Criterios de Aceptación (Título)	Contexto	Evento	Comportamiento esperado
7	Administrador	Se necesita una interfaz para visualizar el detalle de los alquileres de construcción civil	Una interfaz que permita ver el detalle de cada alquiler.	3	Actualizar Alquiler	En caso de querer Actualizar Alquiler		Alquiler Actualizado
				4	Anular Alquiler	En caso de querer Anular Alquiler		Alquiler Anulado.
				1	Añadir Equipo al Alquiler	En caso de querer añadir más equipos al alquiler		Equipo añadido al alquiler
				2	Actualizar Equipo del Alquiler	En caso de querer editar la información de un Equipo en alquiler		Equipo del alquiler actualizado
				3	Quitar Equipo del Alquiler	En caso de querer quitar un equipo del alquiler.		Equipo quitado del Alquiler.

Enunciado de la historia					Criterios de Aceptación			
Identificador de la historia (ID)	Rol	Característica/ Funcionalidad	Resultado	# de escenario	Criterios de Aceptación (Título)	Contexto	Evento	Comportamiento esperado
8.	Administrador	Se necesita una interfaz para obtener una guía de un alquiler.	Una opción para poder obtener la guía del alquiler en cuestión.	1	Obtener la guía del alquiler.	En caso de necesitar una guía en formato PDF de un determinado alquiler.		Archivo PDF con el detalle del alquiler realizado.
9.	Administrador	Se necesita una opción para enviar una notificación de finalización de alquiler.	Se necesita una opción para poder enviar una notificación al cliente de que su alquiler está próximo a finalizar.	1	Enviar notificación de vencimiento de Alquiler.	En caso de necesitar notificar al cliente que su alquiler estará próximo a finalizar.		Mensaje de WhatsApp y de correo electrónico enviado satisfactoriamente con los detalles del alquiler.

Pila del Producto (Product Backlog)

A continuación, se muestran los requerimientos desde la perspectiva del usuario final (Personal INC y Visitante) e interesados, los cuales encarnan la expectativa que tienen acerca del sistema informático web en elaboración.

Tabla 4

Pila de productos (Product Backlog)

ID	FUNCIONALIDAD	IMPORTANCIA
01	Desarrollar una interfaz para que el usuario pueda registrarse en la aplicación	Alta
02	Desarrollar una interfaz para que el usuario pueda ingresar a la aplicación.	Alta
03	Desarrollar una interfaz para que el administrador pueda administrar los usuarios de la aplicación	Alta
04	Desarrollar una interfaz para que el administrador pueda administrar los equipos de construcción civil.	Alta
05	Desarrollar una interfaz para visualizar los alquileres de los equipos de construcción civil.	Alta
06	Desarrollar una interfaz para visualizar el detalle de los alquileres de construcción civil	Alta
07	Desarrollar una interfaz para obtener una guía de un alquiler.	Alta
08	Desarrollar una opción para enviar una notificación de finalización de alquiler.	Alta

Sprints del Producto

Tabla 5

Detallado del Sprint 01

Sprint	Inicio	duración	Objetivo			
01	Jue 01/09/2022	30h	Construir el entorno de la aplicación web.			
		Horas pendientes	192			
		Tareas pendientes	27			
BACKLOG ID: 01 – Desarrollar una interfaz para que el usuario pueda registrarse en la aplicación.						
Tarea	Categoría	Estimado en Hrs	Estado	Responsable	Esfuerzo 0-10	
Conseguir información de los trabajadores de Roosveth Servicios Generales y potenciales clientes.	Análisis	10	Terminado	Esmilton Jiménez	2	
Crear el proyecto de la aplicación web	Implementación	5	Terminado	Esmilton Jiménez	3	
Realizar configuraciones básicas de la aplicación web.	Implementación	2	Terminado	Esmilton Jiménez	5	
Crear prototipos de ingreso y registro de usuarios.	Diseño	10	Terminado	Esmilton Jiménez	5	
Crear tablas en base de datos para los usuarios	Implementación	3	Terminado	Esmilton Jiménez	4	

Tabla 6*Detallado del Sprint 02*

Sprint	Inicio	Duración	Objetivo			
02	Mie 07/09/2022	31h	Construir el entorno de administración de usuarios en la aplicación.			
		Horas pendientes	162			
		Tareas pendientes	22			
BACKLOG ID: 02 y 03.						
Tarea	Categoría	Estimado en Hrs	Estado	Responsable	Esfuerzo 0-10	
Programar lógica para registro e ingreso de usuarios.	Implementación	7	Terminado	Esmilton Jiménez	6	
Programar lógica de administración de Roles de Usuario.	Implementación	10	Terminado	Esmilton Jiménez	7	
Programar lógica de administración de datos de Usuario.	Implementación	7	Terminado	Esmilton Jiménez	6	
Programar lógica de cambio de contraseña de Usuario.	Implementación	7	Terminado	Esmilton Jiménez	6	

Tabla 7*Detalle del Sprint 03*

Sprint	Inicio	Duración	Objetivo		
03	Mie 19/09/2022	29h	Construir el entorno de administración de los equipos para alquilar.		
		Horas pendientes	131		
		Tareas pendientes	18		
BACKLOG ID: 04 – Desarrollar una interfaz para que el administrador pueda administrar las categorías de los equipos de construcción civil.					
Tarea	Categoría	Estimado en Hrs	Estado	Responsable	Esfuerzo 0-10
Crear interfaz de administración de los Equipos.	Diseño	5	Terminado	Esmilton Jiménez	5
Programar lógica para crear un Equipo.	Implementación	6	Terminado	Esmilton Jiménez	7
Programar lógica para ver la lista de Equipos.	Implementación	5	Terminado	Esmilton Jiménez	6
Programar lógica para actualizar los datos de un Equipo.	Implementación	3	Terminado	Esmilton Jiménez	6
Programar lógica para eliminar Equipos.	Implementación	10	Terminado	Esmilton Jiménez	9

Tabla 8*Detalle del Sprint 04*

Sprint	Inicio	Duración	Objetivo			
04	Lun 03/10/2022	32h	Construir el entorno de administración de los alquileres realizados.			
		Horas pendientes	102			
		Tareas pendientes	13			
BACKLOG ID: 05 – Desarrollar una interfaz para que el administrador pueda administrar los equipos de construcción civil.						
Tarea	Categoría	Estimado en Hrs	Estado	Responsable	Esfuerzo 0-10	
Crear interfaz de administración del detalle de los alquileres realizados.	Diseño	8	Terminado	Esmilton Jiménez	8	
Programar lógica para Crear un Alquiler.	Implementación	7	Terminado	Esmilton Jiménez	7	
Programar lógica para ver la lista de Alquileres.	Implementación	8	Terminado	Esmilton Jiménez	6	
Programar lógica para actualizar los datos de un Alquiler.	Implementación	5	Terminado	Esmilton Jiménez	7	
Programar lógica para anular Alquileres.	Implementación	4	Terminado	Esmilton Jiménez	5	

Tabla 9*Detalle del Sprint 05*

Sprint	Inicio	Duración	Objetivo		
05	Jue 13/10/2022	35h	Construir el entorno de administración del detalle de cada alquiler realizado.		
		Horas pendientes	70		
		Tareas pendientes	8		
BACKLOG ID: 06 – Desarrollar una interfaz para visualizar los alquileres de los equipos de construcción civil.					
Tarea	Categoría	Estimado en Hrs	Estado	Responsable	Esfuerzo 0-10
Crear interfaz de administración de los alquileres.	Diseño	10	Terminado	Esmilton Jiménez	9
Programar lógica para añadir un equipo al alquiler.	Implementación	12	Terminado	Esmilton Jiménez	9
Programar lógica para actualizar datos de un equipo en el alquiler.	Implementación	7	Terminado	Esmilton Jiménez	6
Programar lógica para quitar un equipo del alquiler.	Implementación	6	Terminado	Esmilton Jiménez	7

Tabla 10*Detallado del Sprint 06*

Sprint	Inicio	Duración	Objetivo			
06	Vier 21/10/2022	35h	Construir el entorno para la obtención de una guía de alquiler realizado y enviar notificaciones a los clientes.			
		Horas pendientes	35			
		Tareas pendientes	4			
BACKLOG ID: 07 – Desarrollar una interfaz para visualizar el detalle de los alquileres de construcción civil.						
Tarea	Categoría	Estimado en Hrs	Estado	Responsable	Esfuerzo 0-10	
Crear interfaz de obtención de guías de los alquileres.	Diseño	8	Terminado	Esmilton Jiménez	8	
Programar lógica para obtener la guía del alquiler.	Implementación	10	Terminado	Esmilton Jiménez	9	
Crear interfaz de envío de notificaciones al cliente.	Implementación	7	Terminado	Esmilton Jiménez	6	
Programar lógica para enviar notificaciones al cliente vía correo y WhatsApp.	Implementación	10	Terminado	Esmilton Jiménez	8	

Diagrama de base de datos relacional

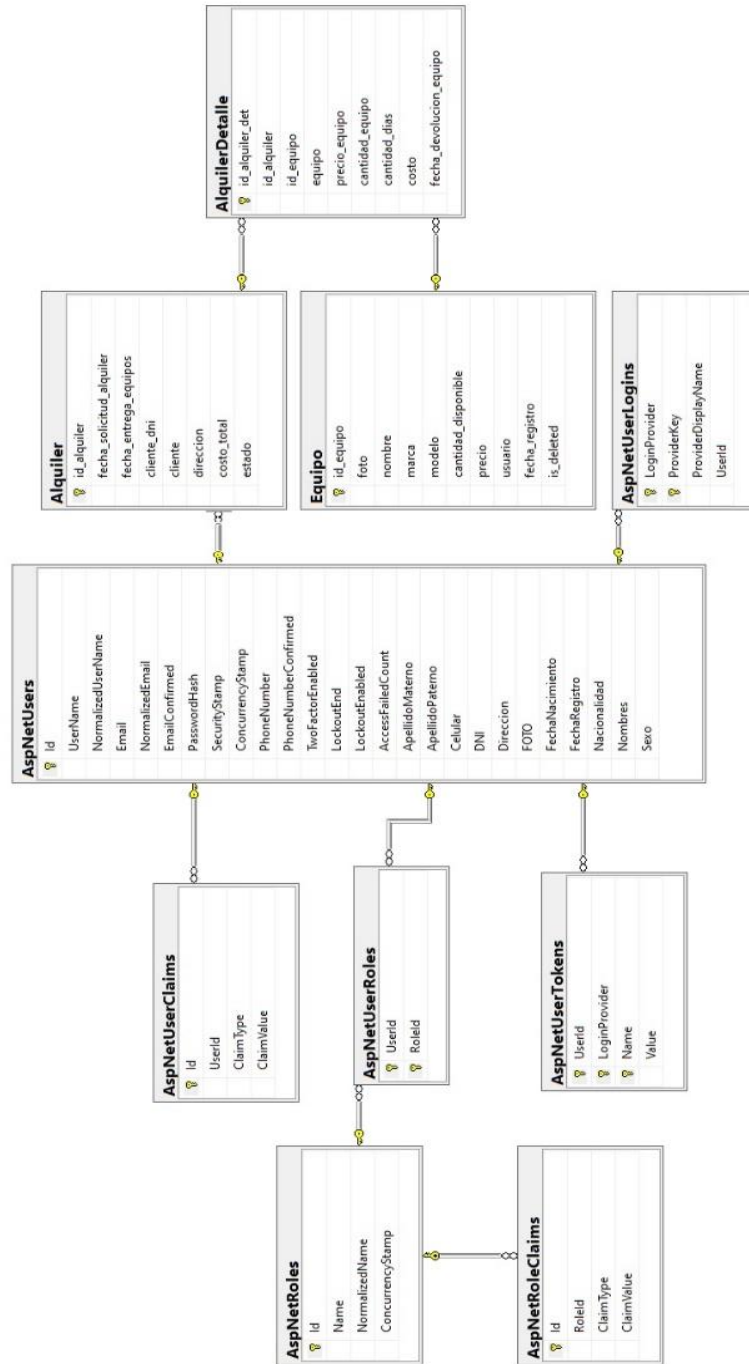


Figura 1 Diagrama de Base de Datos Relacional

Tablas de negocio

Tabla 11

Denominación de las tablas de negocio

Nombre de Tabla	Descripción
Alquiler	Tabla donde se acumulará la información de cabecera correspondiente a los alquileres realizados por la empresa.
AlquilerDetalle	Tabla donde se desglosa la información de un alquiler y guardará la información del detalle de un alquiler realizado.
Equipo	Tabla donde se acumulará la información de los Equipos para alquiler con los que cuenta la empresa.

Definiciones de tablas

Alquiler

Tabla 12

Descripción de la Tabla Alquiler

Columna	Tipo de Dato	Es nulo	Comentario
id_alquiler	INT (PK)	NOT NULL	Campo identificador de la tabla
fecha_solicitud_alquiler	DATETIME	NULL	Fecha que indica cuando se ha realizado la petición de alquiler por parte un cliente.
fecha_entrega_equipos	DATETIME	NULL	Fecha que indica cuando sería la entrega de los equipos al cliente.
Cliente_dni	VARCHAR(10)	NULL	Número de DNI del cliente.
Cliente	VARCHAR(100)	NULL	Campo que almacenará nombre completo y apellidos del cliente.

Dirección	VARCHAR (200)	NULL	Campo que almacenará la dirección donde estarán trabajando los equipos.
Costo_total	DECIMAL (10,2)	NULL	Campo que almacenará el costo total del alquiler expresado en soles.
Estado	VARCHAR (10)	NULL	Campo que indicará el estado general del alquiler ya sea PENDIENTE, ENTREGADO o FINALIZADO.

AlquilerDetalle

Tabla 13

Descripción de la Tabla AlquilerDetalle

Columna	Tipo de Dato	Es nulo	Comentario
id_alquiler_detalle	INT (PK)	NOT NULL	Campo identificador de la tabla.
Id_alquiler	INT (FK)	NOT NULL	Clave foránea a tabla Alquiler.
Id_equipo	INT (FK)	NOT NULL	Clave foránea a tabla Equipo.
Equipo	VARCHAR(50)	NULL	Nombre del equipo que se alquila.
Precio_equipo	DECIMAL(10,2)	NOT NULL	Precio del equipo que se alquila.
Cantidad_equipo	INT	NOT NULL	Cantidad de los equipos que se alquilarán.

Cantidad_dias	INT	NOT NULL	Cantidad de días por los que se alquilarán los equipos.
Costo	DECIMAL(10,2)	NOT NULL	Costo calculado multiplicando la cantidad de equipos, por el precio por equipo y la cantidad de días.
Fecha_devolución_equipos	DATETIME	NULL	Fecha que se calculará desde la fecha de ceder los equipos más la suma de la cantidad de días de los equipos.

Tabla 14

Descripción de la tabla Equipo

Columna	Tipo de Dato	Es Nulo	Comentario
Id_equipo	INT (PK)	NOT NULL	Campo identificador de la tabla
foto	IMAGE	NULL	Foto del equipo en bytes, codificada en base 64.
Nombre	VARCHAR(100)	NULL	Nombre del equipo
Marca	VARCHAR(50)	NOT NULL	Marca del equipo
Cantidad_disponible	INT	NOT NULL	Cantidad de equipos con los que cuenta la empresa, de un equipo determinado.

Precio	DECIMAL(10,2)	NOT NULL	Precio de alquiler del equipo.
Usuario	VARCHAR(50)	NOT NULL	Persona que registro el equipo.
Fecha_registro	DATETIME	NOT NULL	Fecha en la que se realizó el registro del equipo.
Is_deleted	BIT	NULL	Dato que sirve para determinar si el(los) equipo(s) puede(n) estar o no disponible, así haya una cantidad mayor a 1 del mismo.

Tabla 15

Descripción de las Tablas de Auditoria

Nombre de Tabla	Descripción
AspNetUsers	Tabla donde se acumulará la información de los usuarios que tengan ingreso a la aplicación.
AspNetRoles	Tabla donde almacenará la descripción de los roles de los usuarios.
AspNetUserRoles	Tabla donde se almacenará los roles con sus respectivos usuarios, esta servirá para manejar permisos según el rol del usuario.
AspNetUserLogins	Tabla para monitorear los ingresos de los usuarios a la aplicación.

Definiciones de tablas.

Tabla 16

Descripción de la tabla AspNetUsers

Columna	Tipo de Dato	Es nulo	Comentario
Id	NVARCHAR(450) (PK)	NOT NULL	Campo identificador de la tabla
UserName	NVARCHAR(256)	NULL	Nombre de usuario con el que el cliente ingresará a la aplicación.
Email	NVARCHAR(256)	NULL	Email con el que el cliente se registra para tener acceso a la aplicación.
EmailConfirmed	BIT	NULL	Indica si el correo electrónico proporcionado por el usuario se ha confirmado y es verídico.
PasswordHash	NVARCHAR(MAX)	NULL	Contraseña cifrada para ingresar a la aplicación.
PhoneNumber	NVARCHAR(MAX)	NULL	Teléfono del cliente.
LockoutEnabled	BIT	NOT NULL	Campo que indica si la cuenta de usuario se ha bloqueado o no por exceso de fallas al momento de ingresar.
AccessFailedCount	INT	NOT NULL	Contador de intentos fallidos de acceso a la aplicación.
Nombres	NVARCHAR(MAX)	NOT NULL	Nombres del cliente.
ApellidoPaterno	NVARCHAR(MAX)	NOT NULL	Apellido paterno del cliente.

ApellidoMaterno	NVARCHAR(MAX)	NOT NULL	Apellido materno del cliente.
Celular	NVARCHAR(MAX)	NOT NULL	Celular del cliente.
DNI	NVARCHAR(MAX)	NOT NULL	Documento nacional de identidad del cliente.
Direccion	NVARCHAR(MAX)	NOT NULL	Dirección del cliente.
FOTO	IMAGE	NULL	Foto del cliente.
FechaNacimiento	DATETIME2	NOT NULL	Fecha de nacimiento del cliente.
FechaRegistro	DATETIME2	NOT NULL	Fecha de registro del cliente en la aplicación.
Nacionalidad	NVARCHAR(MAX)	NOT NULL	Nacionalidad del cliente.
Sexo	NVARCHAR(MAX)	NOT NULL	Sexo del cliente.

Tabla 17

Descripción de la Tabla AspNetRoles

Columna	Tipo de Dato	Es Nulo	Comentario
Id	NVARCHAR(450)(PK)	NOT NULL	Campo identificador de la tabla
Name	NVARCHAR(256)	NULL	Nombre identificador del rol
NormalizedName	NVARCHAR(256)	NULL	Nombre identificador del rol en mayúsculas.
ConcurrencyStamp	VARCHAR(50)	NOT NULL	Token del rol.

Tabla 18*Descripción de la tabla AspNetUserRoles*

Columna	Tipo de Dato	Es Nulo	Comentario
UserId	NVARCHAR(PK)(FK)	NOT NULL	Campo identificador de la tabla y a la vez foránea a la tabla de usuarios.
RoleId	NVARCHAR(PK)(FK)	NOT NULL	Clave foránea a la tabla de Roles.

Tabla 19. Descripción de la tabla AspNetUserLogins

Columna	Tipo de Dato	Es Nulo	Comentario
LoginProvider	NVARCHAR (PK)	NOT NULL	Id del proveedor del login.
ProviderKey	NVARCHAR (PK)	NOT NULL	Clave Token para ingresar.
ProviderDisplayName	NVARCHAR (MAX)	NULL	Nombre del proveedor del Login.
UserId	NVARCHAR (FK) (450)	NOT NULL	Id Foránea a la tabla de usuarios.

Arquitectura común de aplicaciones web .net

ASP.NET Core Architecture

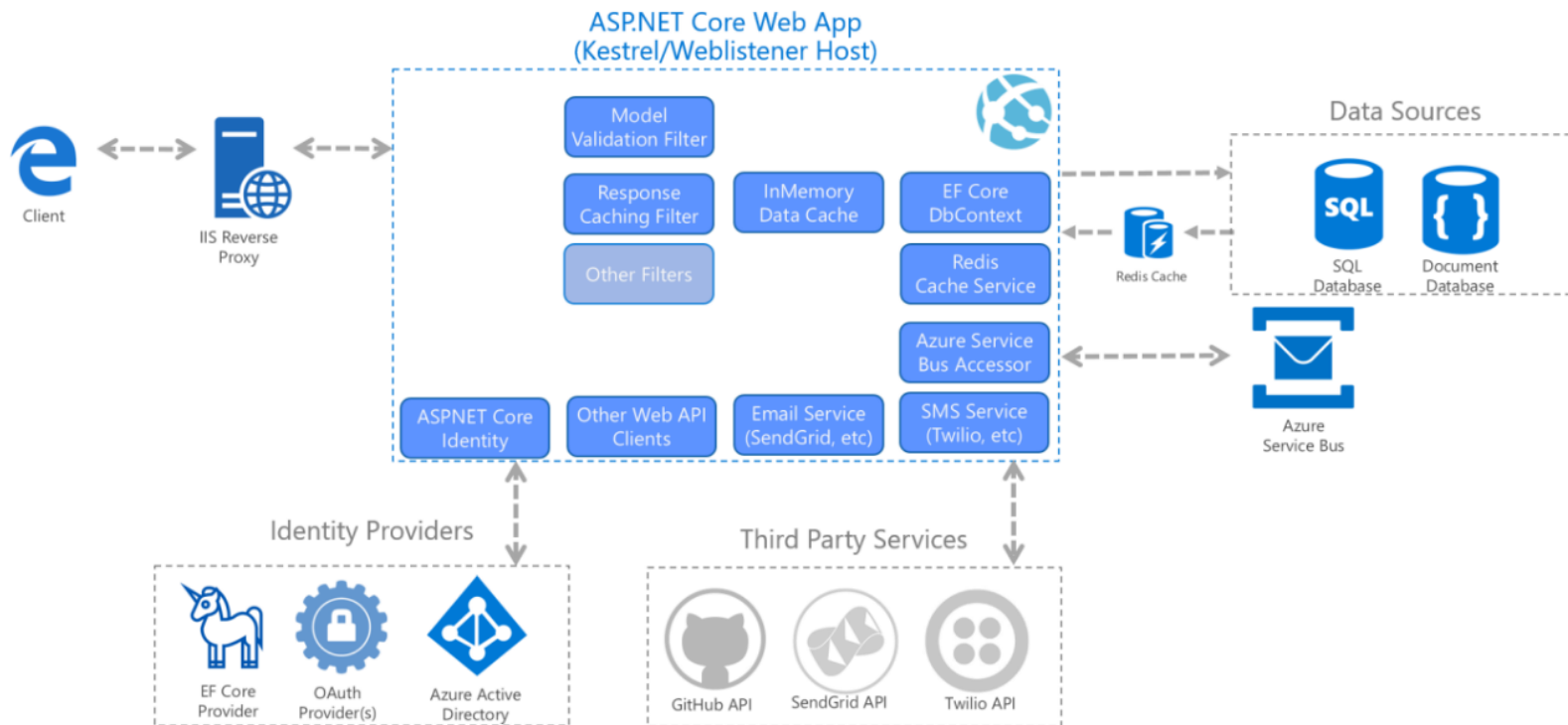


Figura 2 Arquitectura común de aplicaciones web .NET 6

Fuente: Microsoft Corporation

Estructura del proyecto de la aplicación web

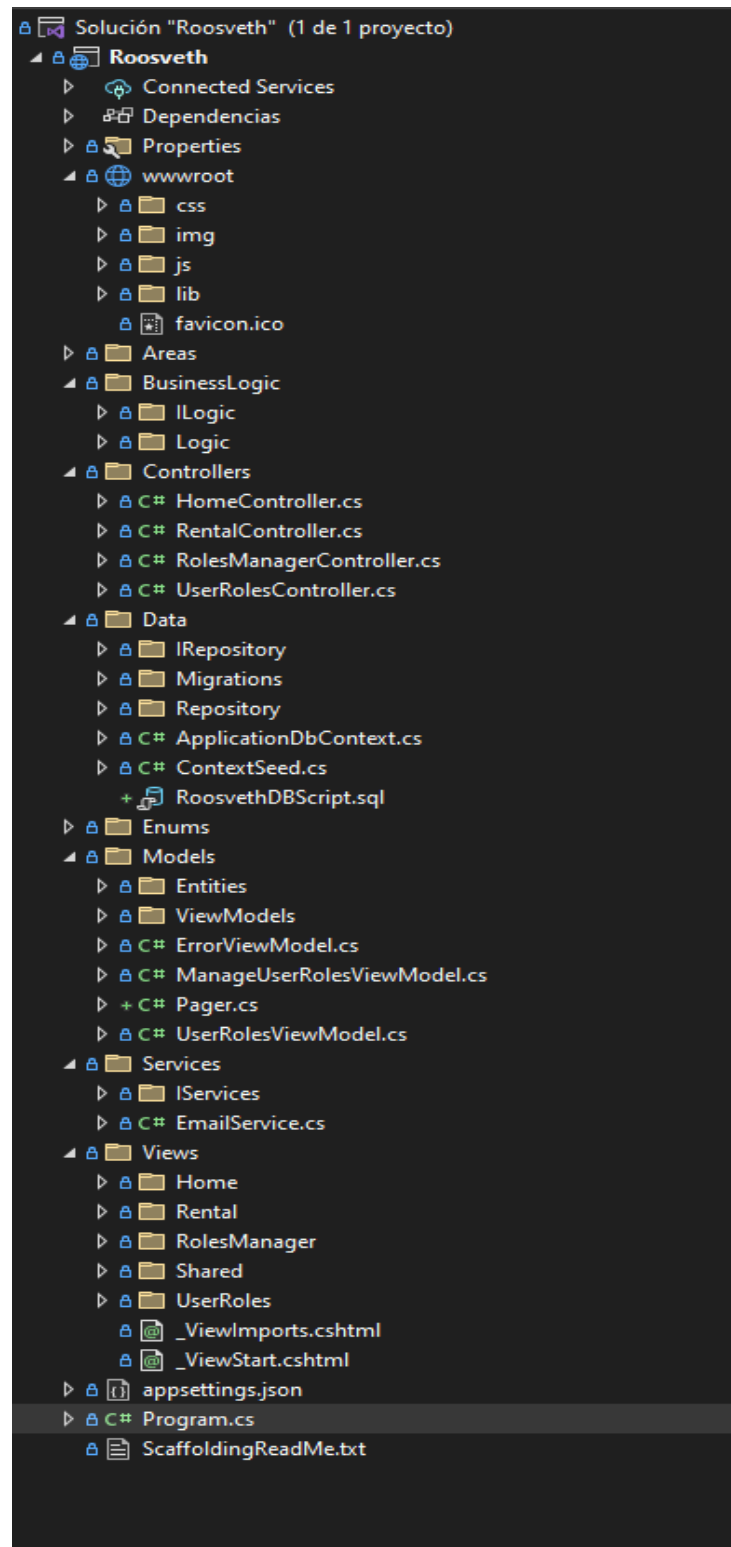


Figura 3. Estructura de Aplicación Web Roosveth.

Carpeta “wwwroot”

Esta carpeta contiene los recursos necesarios que se descargarán en el navegador web del cliente al momento en que el usuario/cliente escriba la dirección (URL) de la aplicación en su navegador, tales como hojas de estilo CSS para embellecer las páginas HTML de las interfaces, algunas imágenes como el logo de la empresa, archivos JavaScript (JS) que sirven para manipular los documentos HTML de las interfaces y realizar interacciones agradables con el usuario, algunas librerías de JavaScript y el ícono de la aplicación.

Carpeta “BusinessLogic”

Esta carpeta contiene 2 subcarpetas ILogic y Logic, en la carpeta Logic se encuentran las implementaciones de toda la programación que abordarán la lógica del negocio, ya sean cálculos de montos, cantidades, validaciones de stock de equipos, etc. En la carpeta ILogic contienen las interfaces que servirán para acceder a las implementaciones mencionadas desde los controladores de la aplicación.

Carpeta “Controllers”

Esta carpeta contiene los famosos endpoints de la aplicación, es decir estos mapean las rutas a la que usuario/cliente finalmente accede a través de su navegador web y técnicamente estas clases de tipo “controlador” de la aplicación son los que hacen uso de las distintas inyecciones de otras partes de la misma tales como el método del negocio, la parte de conexión a base de datos, la parte de servicios conectados y finalmente las interfaces (vistas).

Carpeta “Data”

Esta carpeta contiene clases para conectar a base de datos, así como en la parte de los parametros de la aplicación vista previamente, que se conduce de la manera: IMPLEMENTACIÓN – INTERFAZ, del mismo modo se ha trabajado esta parte de la aplicación, donde tenemos interfaces

para acceder a las clases de tipo “Repository” (repositorio en español) las cuales implementan el código para ingresar a la información de las tablas de base de datos y ejecutar operaciones sobre la misma haciendo uso de la tecnología LinQ de la biblioteca de clases de EntityFramework, la cual nos facilita la ejecución de transacciones a realizarse a nivel de base de datos. Finalmente se tiene una clase “ApplicationDbContext.cs” del contexto de datos, la cual sirve para mapear las tablas de nuestra base de datos.

Carpeta “Enums”

Esta carpeta contiene las clases para definir los Enumeradores que básicamente son como valores constantes que pueden haber dentro de la aplicación y sirven en modo de reutilización, en este caso solamente se consideró una clase Enumerador para los Roles del Usuario que son: Administrador y Cliente.

Carpeta “Models”

Esta carpeta contiene las subcarpetas Entities y ViewModels, la primera contiene el mapeo de las tablas de base de datos a entidades, es decir la definición de una tabla de base de datos como una clase en código C# con su definición del objeto de base de datos, la segunda contiene las definiciones de los objetos que el usuario verá en su navegador cuando realice interacciones, esto debido a que no es recomendable que el usuario sepa en su totalidad cuales son las entidades de base de datos y los campos de las mismas, esto por motivos de seguridad y de simplificación de objetos para mejorar el rendimiento de la aplicación, entre otros.

Carpeta “Services”

Dentro de esta carpeta están las clases que harán la implementación de los servicios requeridos para la aplicación, como muestra la remisión de emails. Se maneja de la misma forma que las otras partes de la aplicación, es decir interfaz e implementación.

Carpeta “Views”

Esta carpeta contiene las vistas o interfaces de usuario, las cuales se descargarán en el navegador del usuario que ingrese a la aplicación, cabe mencionar que son vistas “RAZOR” una tecnología de .NET que permiten a los documentos tener la extensión CSHTML debido a que son documentos HTML que se les puede inyectar código C# (C-Sharp) directamente haciendo uso del símbolo arroba, esto mejora el rendimiento de la aplicación y asegura la robustez ya que como el lenguaje C# es fuertemente tipado, es menos vulnerable a distintos tipos de Bugs y problemas de seguridad, etc.

Archivos “Program.cs” y “appsettings.json”

El primer archivo Program.cs es una clase global que permite gestionar y realizar toda la configuración del programa para que arranque, conexión a base de datos, seguridad, Cookies, Mapeo de interfaces y clases, mapeo de servicios conectados, configuración de ambientes de prueba o de producción, etc. Y el archivo appsettings.json contiene definición de rutas, credenciales, valores constantes, etc.

Configuración de base de datos

El sucesivo bloque de código se emplea la inyección de todos los paquetes que servirán para construir. en la clase principal Program.cs de la aplicación web, además obtiene la serie de contacto a la base de datos del documento de configuración AppSettings.json y se añade el contexto de datos con la clase ApplicationDbContext.cs

```
Program.cs - X
Roosveth
1 using Microsoft.AspNetCore.Identity;
2 using Microsoft.EntityFrameworkCore;
3 using Roosveth.Data;
4 using Roosveth.Models.Entities;
5 using Roosveth.Services.IServices;
6 using Roosveth.Services;
7 using Roosveth.Data.IRepository;
8 using Roosveth.Data.Repository;
9 using Roosveth.BusinessLogic.Logic;
10 using Roosveth.BusinessLogic.ILogic;
11
12 var builder = WebApplication.CreateBuilder(args);
13
14 // Add services to the container.
15 var connectionString = builder.Configuration.GetConnectionString("SQLServerConnection");
16 builder.Services.AddDbContext<ApplicationDbContext>(options =>
17     options.UseSqlServer(connectionString));
```

Figura 4. Pantallazo de la configuración de base de datos

Configuración de clase base para la seguridad y el registro de usuarios.

En el siguiente bloque de código de la clase Program.cs se define cual será el objeto – entidad que manejará la aplicación para acceder a ella, así como los parámetros de ingreso/registro tales como si se le especificará que cada vez que un usuario se registre en la aplicación este confirme obligatoriamente su correo electrónico, etc.

```
19 builder.Services.AddIdentity<ApplicationUser, IdentityRole>(options =>
20 {
21     options.SignIn.RequireConfirmedAccount = true;
22     options.Password.RequireDigit = false;
23     options.Password.RequireLowercase = false;
24     options.Password.RequireUppercase = false;
25     options.Password.RequireNonAlphanumeric = false;
26     options.Password.RequiredLength = 6;
27 })
28 .AddEntityFrameworkStores<ApplicationDbContext>()
29 .AddDefaultTokenProviders(); //para enviar el Token por default de confirmación de email
```

Figura 5. Pantallazo de configuración para la seguridad y el registro de usuarios

Configuración de cookies, inyección de servicios e interfaces.

En este bloque de código de añaden los servicios básicos de la librería de clases de .NET Core para aplicaciones web, tales como el agregador de controladores con vistas, etc. Además, aquí especificamos cuales serán nuestras interfaces y cuáles serán las clases que las implementan para poder ser utilizadas.

```

30 builder.Services.ConfigureApplicationCookie(options =>
31 {
32     options.LoginPath = "/Identity/Account/Login";
33     options.AccessDeniedPath = "/Identity/Account/AccessDenied";
34 });
35
36 builder.Services.AddDatabaseDeveloperPageExceptionFilter();
37 builder.Services.AddControllersWithViews();
38 builder.Services.AddRazorPages();
39 builder.Services.AddMvc().AddRazorRuntimeCompilation(); //Microsoft.AspNetCore.Mvc.Razor.RuntimeCompilation
40
41 builder.Services.AddTransient<IEmailService, EmailService>();
42 builder.Services.AddTransient<IGenericEntityRepository<object>, GenericEntityRepository>();
43 builder.Services.AddTransient<IALquilerRepository, AlquilerRepository>();
44 builder.Services.AddTransient<IEquipoBL, EquipoBL>();
45 builder.Services.AddTransient<IALquilerBL, AlquilerBL>();
46
47 var app = builder.Build();

```

Figura 6. Pantallazo de configuración de cookies, inyección de servicios e interfaces.

Clase controlador de la aplicación

En la siguiente clase HomeController.cs como su nombre indica, son las que manejan básicamente toda la aplicación, en estas clases inyectamos el usuario logeado, la parte del ingreso a los datos, la parte lógica y retornamos al cliente / usuario una vista con datos.

```

HomeController.cs
Roosveth Roosveth.Controllers.HomeController
1 using Microsoft.AspNetCore.Authorization;
2 using Microsoft.AspNetCore.Identity;
3 using Microsoft.AspNetCore.Mvc;
4 using Roosveth.BusinessLogic.ILogic;
5 using Roosveth.Data.IRepository;
6 using Roosveth.Models;
7 using Roosveth.Models.Entities;
8 using System.Diagnostics;
9
10 namespace Roosveth.Controllers
11 {
12     public class HomeController : Controller
13     {
14         private readonly UserManager<ApplicationUser> _userManager;
15         private readonly IGenericEntityRepository<object> _genericEntityOpsInterface;
16         private readonly IEquipoBL _equipmentBL;
17
18         public HomeController(UserManager<ApplicationUser> userManager,
19             IGenericEntityRepository<object> genericEntityOpsInterface,
20             IEquipoBL equipoBL)
21         {
22             _userManager = userManager;
23             _genericEntityOpsInterface = genericEntityOpsInterface;
24             _equipmentBL = equipoBL;
25         }
26
27         public async Task<IActionResult> Index()
28         {
29             IEnumerable<Equipo>? equipos = null;
30             try
31             {
32                 equipos = await _genericEntityOpsInterface.GetAllEntity<Equipo>();
33                 return View(equipos);
34             }
35             catch (Exception e)
36             {
37                 ViewBag.result = $"ERROR: {e.InnerException?.Message}";
38                 return View(equipos);
39             }
40         }
41     }

```

Figura 7. Pantallazo de la clase controlador de la aplicación

Clase lógica de la aplicación: EquipoBL.cs

En esta clase colocamos de ejemplo como se maneja la lógica la aplicación antes de devolver o ingresar datos desde o hasta la interfaz del usuario, en este ejemplo hacemos unas validaciones para que el precio del equipo que se va a registrar no sea igual o menor a cero o que la cantidad disponible de los equipos tampoco sea menor a cero, entre otras.

```
4 namespace Roosveth.BusinessLogic.Logic
5 {
6     public class EquipoBL : IEquipoBL
7     {
8         public Equipo OnPostAddOrUpdateEquipment(ApplicationUser user, Equipo equipo, Equipo? eqdb)
9         {
10             try
11             {
12                 var userName = user.UsuarioNombreCompleto;
13
14                 if (equipo.precio == 0 || equipo.precio < 0)
15                 {
16                     throw new Exception("ERROR: el precio debe ser Mayor a 0");
17                 }
18
19                 if (equipo.cantidad_disponible < 0)
20                 {
21                     throw new Exception("ERROR: la cantidad de equipos disponibles no puede ser menor a 0");
22                 }
23
24                 decimal p = decimal.Parse(equipo.precio.ToString());
25
26                 if (equipo.id_equipo > 0)
27                 {
28                     eqdb.nombre = equipo.nombre;
29                     eqdb.marca = equipo.marca;
30                     eqdb.modelo = equipo.modelo;
31                     eqdb.cantidad_disponible = equipo.cantidad_disponible;
32                     eqdb.precio = p;
33                     eqdb.usuario = userName;
34                     eqdb.fecha_registro = DateTime.Now;
35                     eqdb.is_deleted = equipo.is_deleted;
36
37                     return (eqdb);
38                 }
39                 else
40                 {
41                     return new Equipo()
42                     {
43                         nombre = equipo.nombre,
44                         marca = equipo.marca,
45                         modelo = equipo.modelo,
46                         cantidad_disponible = equipo.cantidad_disponible,
47                         precio = p,
48                         usuario = userName,
49                         fecha_registro = DateTime.Now,
50                         is_deleted = equipo.is_deleted
51                     };
52                 }
53             }
54             catch (Exception e)
55             {
56                 throw new Exception(e.Message);
57             }
58         }
59     }
60 }
```

Figura 8. Pantallazo de la clase lógica de la aplicación: EquipoBL.cs

Clase de acceso a datos de la aplicación: GenericEntityRepository.cs

En esta clase haciendo uso de métodos genéricos reutilizables, de nuestra clase de contexto de datos ApplicationDbContext.cs y de LinQ para examinar la información a la base de datos colocamos dos instrucciones (GetAllEntity y GetAllEntityByConditions), una para traer todos los registros de una tabla y la otra para obtener solo algunos registros basados en una condición.

```
6 namespace Roosveth.Data.Repository
7 {
8     public class GenericEntityRepository : IGenericEntityRepository<object>
9     {
10         private readonly ApplicationDbContext _appDbContext;
11
12         public GenericEntityRepository(ApplicationDbContext applicationDbContext)
13         {
14             _appDbContext = applicationDbContext;
15         }
16
17         public async Task<IEnumerable<T>> GetAllEntity<T>() where T : class
18         {
19             var objects = await _appDbContext.Set<T>().ToListAsync();
20             return objects;
21         }
22
23         public async Task<IEnumerable<T>> GetAllEntitiesByConditions<T>(Expression<Func<T, bool>> predicate) where T : class
24         {
25             IEnumerable<T> result;
26             result = await _appDbContext.Set<T>().AsQueryable().Where(predicate).ToListAsync();
27             return result;
28         }
29     }
30 }
```

Figura 9.pantallazo de acceso a datos: GenericEntityRepository.cs

Clase entity: Equipo.cs

En esta clase se define la entidad de base de datos, con sus respectivos tipos de datos, además se realizan algunas anotaciones para mostrar los nombres personalizados de los campos, algunas validaciones que nos provee .NET Core como que un dato es requerido o una fecha debe tener cierto formato personalizado, etc.

```

5 namespace Roosveth.Models.Entities
6 {
7     public class Equipo
8     {
9         [Key]
10        [Display(Name = "ID")]
11        [Required(ErrorMessage = "Debe tener un ID")]
12        public int id_equipo { get; set; }
13
14        [Display(Name = "Imagen Referencial del Equipo")]
15        [Column(TypeName = "image")]
16        public byte[]? foto { get; set; }
17
18        [Required(ErrorMessage = "Debe ingresar un nombre para el Equipo")]
19        [Display(Name = "Nombre")]
20        public string nombre { get; set; } = String.Empty;
21
22        [Required(ErrorMessage = "Debe ingresar la marca del Equipo")]
23        [Display(Name = "Marca")]
24        public string marca { get; set; } = String.Empty;
25
26        [Display(Name = "Marca")]
27        public string? modelo { get; set; } = String.Empty;
28
29        [Required(ErrorMessage = "Ingrese la cantidad de Equipos Disponibles")]
30        [Display(Name = "Cantidad Disponible")]
31        public int cantidad_disponible { get; set; }
32
33        [Required(ErrorMessage = "Ingrese el precio de Alquiler")]
34        [RegularExpression(@"^d+(\.\d{1,2})?$", ErrorMessage = "Precio inválido")]
35        [Range(typeof(decimal), "0", "79228162514264337593543950335", ErrorMessage = "El valor no es decimal")]
36        [Display(Name = "Precio por Día")]
37        public decimal precio { get; set; }
38
39        [Display(Name = "Usuario")]
40        public string usuario { get; set; } = String.Empty;
41
42        [DisplayFormat(DataFormatString = "{0:dd-MM-yyyy hh-mm-ss}")]
43        public DateTime fecha_registro { get; set; }
44
45        [Display(Name = "Activo")]
46        [Required(ErrorMessage = "Por favor indique si está disponible")]
47        public bool is_deleted { get; set; }
48    }
49 }

```

Figura 10. Pantallazo de la Clase entity: Equipo.cs

Clase view model: CreatedAlquilerViewModel.cs

En esta clase de modelo de vista o también conocido como DTO (Data Transfer Object) u Objeto de Transferencia de Datos en español, definimos los datos que queremos que un usuario vea en la interfaz de nuestra aplicación, no necesariamente todos, combinando propiedades de una o más entidades o tablas.

```
3 namespace Roosveth.Models.ViewModels
4 {
5     public class CreatedAlquilerViewModel
6     {
7         [Display(Name = "Código de Alquiler")]
8         [Required(ErrorMessage = "El código del alquiler es requerido")]
9         public string codigo { get; set; } = string.Empty;
10
11         public string fecha_solicitud { get; set; } = string.Empty;
12
13         public string fecha_entrega { get; set; } = string.Empty;
14
15         public string dni { get; set; } = string.Empty;
16
17         public string cliente { get; set; } = string.Empty;
18
19         public string direccion { get; set; } = string.Empty;
20
21         public string costo { get; set; } = string.Empty;
22
23         public string estado { get; set; } = string.Empty;
24
25         public List<Detail>? details { get; set; }
26
27         public class Detail
28         {
29             [Required(ErrorMessage = "El codigo del equipo es requerido")]
30             public string id_equipo { get; set; } = string.Empty;
31
32             public string equipo { get; set; } = string.Empty;
33
34             public string precio { get; set; } = string.Empty;
35
36             public string cantidad_e { get; set; } = string.Empty;
37
38             public string cantidad_d { get; set; } = string.Empty;
39
40             public string costo { get; set; } = string.Empty;
41
42             public string? fecha_devolucion { get; set; }
43         }
44     }
45 }
```

Figura 11. Pantallazo de la Clase view model - CreatedAlquilerViewModel.cs

Arquitectura en la nube Azure de la aplicación.

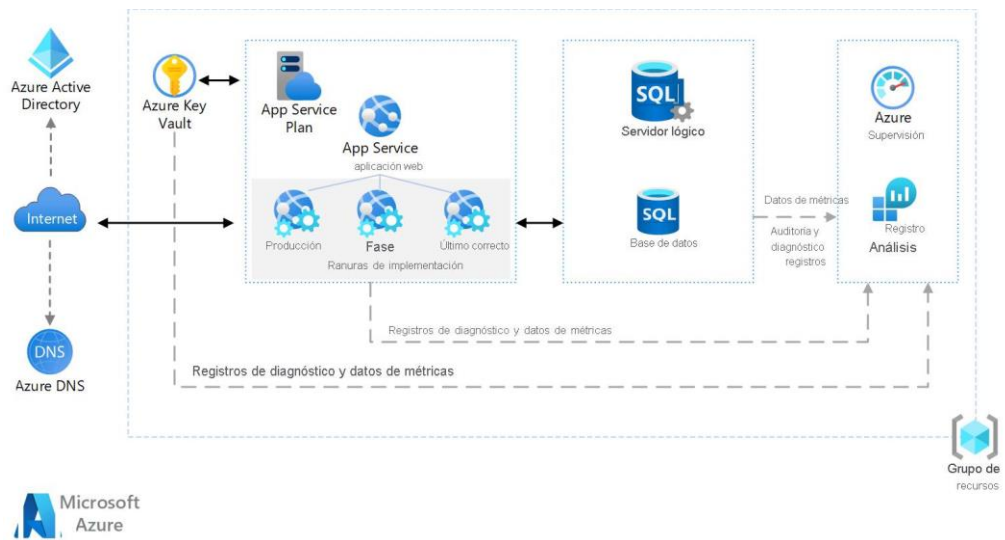


Figura 12. Diagrama de Arquitectura Azure de Aplicación en la Nube

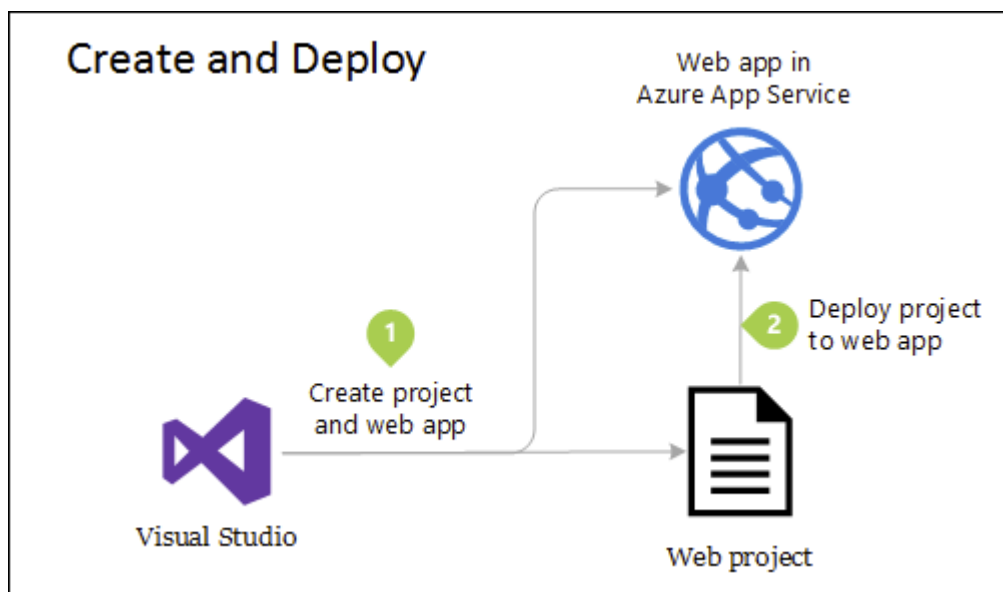


Figura 13. Diagrama de Creación y Despliegue en Azure App Service



Crear una nueva cuenta en Roosveth S.G E.I.R.L

DNI

Nombres

Apellido Paterno

Apellido Materno

Correo electrónico


Contraseña

Confirmar Contraseña

Registrar

¿Ya estás registrado? [Entrar a la App](#)

Figura 14. Pantallazo de registro de clientes



ROOSVETH
SERVICIOS GENERALES E.I.R.L

Ingresar en Roosveth SG E.I.R.L

Correo electrónico

Ingrese su Email


Contraseña

[¿Olvidaste tu contraseña?](#)
[¿No has confirmado tu email para Entrar?](#)

Entrar

[¿No tienes una cuenta? Registrarme](#)

Figura 15. Pantallazo de login de cliente



Esmilton Jiménez Hoyos
esmitorgimenez@gmail.com

Equipos Alquileres

Mi Perfil


DNI 00000000	Apellido Paterno Jiménez	Foto de Perfil  Seleccionar archivo Ninguno archivo selec.
Nombres Esmilton	Apellido Materno Hoyos	
Sexo MASCULINO	Fecha de Nacimiento 28/03/1992 00:00	
Telefono de Familiar 76603165	Celular Personal 958461520	
<p style="text-align: left;">Cancelar</p> <p style="text-align: right;">Guardar</p>		

Figura 16. Pantallazo de mantenimiento de datos personales del cliente

Ver Usuarios y sus Roles

Show 10 entries

#	DNI	Nombres	Apellido Paterno	Apellido Materno	Sexo	Email	Roles	Actualizar	Cambiar Pass	Admin. Roles
1	00000000	Esmilton	Jiménez	Hoyos	MASCULINO	esmltonjimenez@gmail.com	Administrador	Actualizar	Cambiar Pass	Admin. Roles
2	71889573	DIDIER GERSON	YEPEZ	CABANILLAS	MASCULINO	didieryepezca@gmail.com	Cliente	Actualizar	Cambiar Pass	Admin. Roles

Showing 1 to 2 of 2 entries

Copyright © 2022 Roosveth Servicios Generales E.I.R.L. - Todos los Derechos Reservados - v1.0

Figura 17. Pantallazo de mantenimiento de usuarios

Actualizar Roles del Usuario (esmltonjimenez@gmail.com) - (Esmilton Jiménez Hoyos)

Administrador
 Cliente

Volver Actualizar

Copyright © 2022 Roosveth Servicios Generales E.I.R.L. - Todos los Derechos Reservados - v1.0

Figura 18. Pantallazo de mantenimiento de roles de usuario

Nuestros Equipos Disponibles

ROOSVETH SERVICIOS GENERALES E.I.R.L. RUC: 20570678451
 JR. LORETO 420 - BR. LA ESPERANZA CAJAMARCA
 CEL: 912307185 - 988953097

ALQUILER DE MARTILLO ELÉCTRICO

MARTILLO ELÉCTRICO
 Marca: BOSH
 Modelo: modelo 1
 Disponible para los Clientes: SI

1/163.36 x día
 # 2 disponible(s)

ROOSVETH SERVICIOS GENERALES E.I.R.L. RUC: 20570678451
 JR. LORETO 420 - BR. LA ESPERANZA CAJAMARCA
 CEL: 912307185 - 988953097

ALQUILER DE APISONADOR TIPO CANGURO

APISONADOR TIPO CANGURO
 Marca: BOSH
 Modelo:
 Disponible para los Clientes: SI

1/80.00 x día
 # 4 disponible(s)


ROOSVETH SERVICIOS GENERALES E.I.R.L. RUC: 20570678451
 JR. LORETO 420 - BR. LA ESPERANZA CAJAMARCA
 CEL: 912307185 - 988953097

ALQUILER DE COMPRESORA

COMPRESORA
 Marca: DREAMSTIME
 Modelo:
 Disponible para los Clientes: SI

1/120.00 x día
 # 1 disponible(s)

Figura 19. Pantallazo de pantalla principal - equipos disponibles



Esmilton Jiménez Hoyos
esmlitonjimenez@gmail.com

Equipos Alquileres

Agregar Equipo

Registrar información del Equipo


Nombre	Cantidad Disponible	Imagen Referencial del Equipo
Marca	Precio por día	
Modelo	¿No está disponible?	
	<input type="checkbox"/>	

Seleccionar archivo | Ninguno archivo selec.

Cancelar Guardar

Copyright © 2022 Roosveth Servicios Generales E.I.R.L. - Todos los Derechos Reservados - v1.0 Ing. Esmilton Jiménez

Figura 20. Pantallazo de registro de información de equipo nuevo



Esmilton Jiménez Hoyos
esmlitonjimenez@gmail.com

Equipos Alquileres

Actualizar Equipo

Actualizar información del Equipo


Nombre	Cantidad Disponible	Imagen Referencial del Equipo
Marca	Precio por día	
Modelo	¿No está disponible?	
	<input type="checkbox"/>	

Seleccionar archivo | Ninguno archivo selec.

Cancelar Actualizar

Copyright © 2022 Roosveth Servicios Generales E.I.R.L. - Todos los Derechos Reservados - v1.0 Ing. Esmilton Jiménez

Figura 21. Pantallazo de mantenimiento de equipo



Esmilton Jiménez Hoyos
esmltorjimenez@gmail.com

Equipos Alquileres

Alquileres Nuevo Alquiler

Administración de Alquileres


Buscar Código de Alquiler:

#	CODIGO	SOICITUD DE ALQUILER	ENTREGA DE EQUIPOS	CLIENTE	ESTADO		
1	43	15/12/2022 16:36:42		DIDIER GERSON YEPEZ CABANILLAS	PENDIENTE	Ver	Gestionar
2	42	14/12/2022 18:06:22		Esmilton Jiménez Hoyos	PENDIENTE	Ver	Gestionar
3	41	14/12/2022 17:55:21	14/12/2022 17:59:03	DIDIER GERSON YEPEZ CABANILLAS	FINALIZADO	Ver	Gestionar
4	40	4/11/2022 09:48:24		Esmilton Jiménez Hoyos	PENDIENTE	Ver	Gestionar
5	39	4/11/2022 09:45:39		DIDIER GERSON YEPEZ CABANILLAS	PENDIENTE	Ver	Gestionar
6	38	4/11/2022 09:41:23		Esmilton Jiménez Hoyos	PENDIENTE	Ver	Gestionar
7	37	4/11/2022 09:21:01		DIDIER GERSON YEPEZ CABANILLAS	PENDIENTE	Ver	Gestionar
8	36	3/11/2022 22:24:13	3/11/2022 22:26:22	Esmilton Jiménez Hoyos	FINALIZADO	Ver	Gestionar
9	35	3/11/2022 21:41:23	3/11/2022 21:43:20	Esmilton Jiménez Hoyos	FINALIZADO	Ver	Gestionar
10	34	3/11/2022 21:33:29		Esmilton Jiménez Hoyos	FINALIZADO	Ver	Gestionar

Mostrando 1 a 10 Registros en 1 Página(s) 1

Copyright © 2022 Roosveth Servicios Generales E.I.R.L - Todos los Derechos Reservados - v1.0 Ing. Esmilton Jiménez

Figura 22. Pantallazo de administración de alquileres



DIDIER GERSON YEPEZ CABANILLAS
didieryepeza@gmail.com

Equipos Mis Alquileres

Solicitud de Alquiler de Equipos

Información del Alquiler

Fecha de Solicitud de Alquiler <input type="text" value="02/01/2023"/>	Cliente DNI <input type="text" value="71889573"/>	Cliente <input type="text" value="DIDIER GERSON YEPEZ CABANILLAS"/>
Fecha de Entrega de Equipos <input type="text" value="Al momento de entregar los equipos"/>	Cliente Celular <input type="text" value="941176618"/>	Dirección <input type="text" value="Escriba la dirección exacta donde trabajará el Equipo"/>

Selección de la Lista de Equipos Disponibles

Show entries Search:

NOMBRE	DISPONIBLES	NECESITAS	PRECIO	DIAS
MARTILLO ELECTRICO	<input type="text" value="2"/>	<input type="text" value="1"/>	163.36	<input type="text" value="1"/>
APISADOR TIPO CANGURO	<input type="text" value="4"/>	<input type="text" value="1"/>	80.00	<input type="text" value="1"/>
COMPRESORA	<input type="text" value="1"/>	<input type="text" value="0"/>	120.00	<input type="text" value="0"/>
AMOLADORA	<input type="text" value="6"/>	<input type="text" value="0"/>	95.50	<input type="text" value="0"/>
CORTADORA DE CONCRETO	<input type="text" value="3"/>	<input type="text" value="0"/>	120.50	<input type="text" value="0"/>
TROMPO	<input type="text" value="1"/>	<input type="text" value="0"/>	50.50	<input type="text" value="0"/>

Showing 1 to 6 of 6 entries Previous Next

Tus Equipos a Alquilar

Costo Total de tu Alquiler \$y. 243.36

1 MARTILLO ELECTRICO	1 Dias
1 APISADOR TIPO CANGURO	1 Dias

Volver al inicio Solicitar

Copyright © 2022 Roosveth Servicios Generales E.I.R.L - Todos los Derechos Reservados - v1.0 Ing. Esmilton Jiménez

Figura 23. Pantallazo de creación de nuevo alquiler

Solicitud de Alquiler Imprimir Solicitud

Código de Solicitud: 44
 Cliente: DIDIER GERSON YEPEZ CABANILLAS
 DNI: 71889573

Datos Importantes
 Fecha de Registro de Solicitud: 2/01/2023 18:42
 Fecha de Entrega de Equipos: Se actualizará al momento de la entrega de los Equipos Solicitados
 Lugar de Trabajo de Equipos: Av. Via de Evitamiento #1864

Estado: PENDIENTE

EQUIPO	PRECIO X DÍA	CANTIDAD	# DÍAS TRABAJO	FECHA DEVOLUCIÓN EQUIPO	COSTO S/.
MARTILLO ELECTRICO	163.36	1	1		163.36
APISADOR TIPO CANGURO	80.00	1	1		80.00
TOTAL S/.					243.36

Muchas Gracias por hacer Negocios con Nosotros. ¡ Esperamos Volver a Trabajar Con Usted !

Copyright © 2022 Roosveth Servicios Generales E.I.R.L. - Todos los Derechos Reservados - v1.0 Ing. Esmilton Jiménez

Figura 24. Pantallazo de resumen de alquiler crea

Gestión de Alquiler

Información básica

- Código de Alquiler: 44
- Estado de Solicitud de Alquiler: **PENDIENTE**
- Fecha de Solicitud: 2/01/2023 18:42
- Fecha de Entrega de Equipos: Se actualizará al momento de la entrega de los Equipos Solicitados
- Costo Total s/. 243.36
- DNI: 71889573
- Cliente: DIDIER GERSON YEPEZ CABANILLAS
- Dirección de Trabajo: Av. Via de Evitamiento #1864

EQUIPO	PRECIO X DIA	# EQUIPOS	# DIAS	COSTO	FECHA DEVOLUCIÓN DE EQUIPO
MARTILLO ELECTRICO	163.36	1	1	163.36	
APISADOR TIPO CANGURO	80.00	1	1	80.00	

[Volver](#)
No Alquilar
Alquilar

Copyright © 2022 Roosveth Servicios Generales E.I.R.L. - Todos los Derechos Reservados - v1.0 Ing. Esmilton Jiménez

Figura 25. Pantallazo de Gestión de Alquiler

CASOS DE PRUEBA DE LA APLICACIÓN WEB SEGÚN TABLA DE HISTORIAS DE USUARIO

Tabla 20

Descripción del caso de prueba 1

CASO DE PRUEBA # 1	
ID DE HISTORIA: 3	ESCENARIO #2
HISTORIA DE USUARIO: Administrar Usuario	
CONDICIONES DE EJECUCION: Tener habilitado el rol de administrador del sistema.	
ENTRADA/PASOS DE EJECUCIÓN:	
<ol style="list-style-type: none">1. Ingresar al sistema web.2. Posicionar el cursor sobre el nombre de usuario autenticado en la aplicación.3. Seleccionar la opción “Mi Perfil” del menú desplegable.4. Modificar los datos que sean necesarios y/o pertinentes5. Hacer Clic en el botón “Guardar”	
RESULTADO ESPERADO: Se actualizan y guardan los datos del usuario de la aplicación autenticado.	
EVALUACION DE LA PRUEBA: La prueba se concluyó satisfactoriamente.	

Tabla 21

Descripción del caso de prueba 2

CASO DE PRUEBA # 2	
ID DE HISTORIA: 3	ESCENARIO #1
HISTORIA DE USUARIO: Administrar Roles de Usuario	
CONDICIONES DE EJECUCION: Tener habilitado el rol de administrador del sistema.	
ENTRADA/PASOS DE EJECUCIÓN:	
<ol style="list-style-type: none">1. Ingresar al sistema web.2. Posicionar el cursor sobre el nombre de usuario autenticado en la aplicación.3. Seleccionar la opción “Administrar Usuarios” del menú desplegable.4. Hacer Clic en el botón “Admin Roles” de un Usuario de la tabla de todos los usuarios registrados en la aplicación.5. Marcar o desmarcar los roles que se requieran para el Usuario en cuestión6. Hacer Clic en el botón “Actualizar”	
RESULTADO ESPERADO: Se actualizan y guardan los roles asignados al usuario seleccionado.	
EVALUACION DE LA PRUEBA: La prueba se concluyó satisfactoriamente.	

Tabla 22

Descripción del caso de prueba 4

CASO DE PRUEBA # 4	
ID DE HISTORIA: 6	ESCENARIO #2
HISTORIA DE USUARIO: Crear Alquiler	
CONDICIONES DE EJECUCION: Haberse autenticado en la aplicación con usuario y clave.	
ENTRADA/PASOS DE EJECUCIÓN:	
<ol style="list-style-type: none">1. Ingresar al sistema web.2. Hacer Clic en la foto de un Equipo de la pantalla de Bienvenida o Hacer Clic en la opción “Mis Alquileres”3. En caso de haber hecho Clic en la opción “Mis Alquileres” se deberá volver hacer Clic en el botón “Nuevo Alquiler”4. Ingresar la Dirección exacta donde trabajarán el o los equipos en la caja de texto correspondiente.5. Seleccionar la cantidad requerida de cada equipo de la lista de equipos disponibles.6. Seleccionar la cantidad de días requeridos por cada equipo.7. El sistema mostrará el resumen previo de los equipos a Alquilar.8. Hacer Clic en el botón “Solicitar”.9. El sistema mostrará el resumen final del alquiler Solicitado.	
RESULTADO ESPERADO: Se crea un alquiler nuevo que será mostrado en la pantalla de Alquileres del Sistema con estado “PENDIENTE”.	
EVALUACION DE LA PRUEBA: La prueba se concluyó satisfactoriamente.	

Tabla 23

Descripción del caso de prueba 6

CASO DE PRUEBA # 6	
ID DE HISTORIA: 6	ESCENARIO #3
HISTORIA DE USUARIO: Gestionar/ADMINISTRAR Alquiler	
CONDICIONES DE EJECUCION: Haberse autenticado en la aplicación con usuario y clave y tener el rol de Administrador.	
ENTRADA/PASOS DE EJECUCIÓN:	
<ol style="list-style-type: none">1. Ingresar al sistema web.2. Hacer Clic en la opción “Alquileres” del menú de navegación de la App.3. Hacer Clic en el botón “Gestionar” del alquiler deseado con estado “EN CURSO”.4. Se tendrá las opciones “Volver”, “Guardar” y “Finalizar”, cuando el alquiler esté en estado “EN CURSO”5. Se ingresa nueva fecha de devolución para cada equipo de la lista de equipos alquilados, en caso se requiera extender el tiempo de alquiler de dichos equipos.6. Hacer Clic en el botón “Guardar”.	
RESULTADO ESPERADO: El alquiler seleccionado mantiene su estado “EN CURSO”, y las fechas de devolución de cada equipo del alquiler se han extendido como nuevos plazos de devolución.	
EVALUACION DE LA PRUEBA: La prueba se concluyó satisfactoriamente.	

Tabla 24

Descripción del caso de prueba 7

CASO DE PRUEBA # 7	
ID DE HISTORIA: 6	ESCENARIO #3
HISTORIA DE USUARIO: Gestionar/FINALIZAR Alquiler	
CONDICIONES DE EJECUCION: Haberse autenticado en la aplicación con usuario y clave y tener el rol de Administrador.	
ENTRADA/PASOS DE EJECUCIÓN:	
<ol style="list-style-type: none">1. Ingresar al sistema web.2. Hacer Clic en la opción “Alquileres” del menú de navegación de la App.3. Hacer Clic en el botón “Gestionar” del alquiler deseado con estado “EN CURSO”.4. Se tendrá las opciones “Volver”, “Guardar” y “Finalizar”, cuando el alquiler esté en estado “EN CURSO”5. Hacer Clic en el botón “Finalizar”.	
RESULTADO ESPERADO: El alquiler seleccionado cambia de estado a “FINALIZADO”, indicando así que el alquiler de los equipos ha terminado.	
EVALUACION DE LA PRUEBA: La prueba se concluyó satisfactoriamente.	

Tabla 25

Descripción del caso de prueba 8

CASO DE PRUEBA # 8	
ID DE HISTORIA: 6	ESCENARIO #3
HISTORIA DE USUARIO: Gestionar/NO REALIZAR Alquiler	
CONDICIONES DE EJECUCION: Haberse autenticado en la aplicación con usuario y clave y tener el rol de Administrador.	
ENTRADA/PASOS DE EJECUCIÓN:	
<ol style="list-style-type: none">1. Ingresar al sistema web.2. Hacer Clic en la opción “Alquileres” del menú de navegación de la App.3. Hacer Clic en el botón “Gestionar” del alquiler deseado.4. Se tendrá las opciones “Volver”, “No Alquilar” y “Alquilar”, cuando el alquiler esté en estado “PENDIENTE”5. Hacer Clic en el botón “No Alquilar”.	
RESULTADO ESPERADO: El alquiler seleccionado cambia de estado a “FINALIZADO” y no tendrá Fecha de Entrega de Equipos, dado que no se realizó el Alquiler.	
EVALUACION DE LA PRUEBA: La prueba se concluyó satisfactoriamente.	

Análisis y Discusión

De los resultados alcanzados, se puede apreciar que el desarrollo de la solución para la gestión de equipos en alquiler utilizando la metodología SCRUM permitió al equipo de desarrollo entregar los productos de calidad a un ritmo ágil, obteniendo una plataforma escalable y segura para almacenar los datos en alta disponibilidad, lo que permite que los usuarios accedan a ella de forma continua; teniendo como principales características, almacenamiento escalable y seguro donde los datos almacenados en la nube se mantienen seguros y escalan fácilmente para satisfacer las necesidades de cualquier proyecto; seguridad de la información, Microsoft Azure ofrece una plataforma de seguridad sólida que protege la aplicación contra amenazas externas; costo reducido: Microsoft Azure es competitiva, lo que disminuye los costos operativos y de mantenimiento; disponibilidad, tiene alta disponibilidad.

Se expone el debate de resultados alcanzados en la investigación realizada, aquí se analiza la relevancia y el significado de los hallazgos, se hace una comparación con otros estudios; asimismo, se discuten los límites y alcances de esta investigación, así como su contribución a la literatura científica. La investigación realizada concuerda con los resultados alcanzados por Díaz (2021), el sistema web desarrollado para la empresa Sinapptic SAS automatiza en su totalidad el servicio de alquiler de equipos, en la implementación se puede observar módulos de registro, visualización de equipos, clientes, reportes de entrega, devolución de equipos y reporte general de equipos debido que han empleado metodologías y buenas prácticas en el desarrollo del software. De igual modo, Cuvi y Rosero (2020) el sistema web permitió automatizar los procedimientos de alquiler de maquinaria, se utilizaron los bloques de gestión de maquinaria/equipos, alquiler, usuarios, clientes, proveedores e informes.

Por otro lado, el estudio realizado sus resultados coinciden con Vera (2019) sobre la elaboración y puesta en funcionamiento de un sistema web para optimizar los procesos de alquiler de equipos, gestión de maquinaria, entre otros, se obtiene la información de manera más oportuna, en ambos casos los empleados obtienen mejor productividad en sus actividades diarias; en la misma línea, sucede lo mismo con Salhuana (2019) y Gallego (2019), la aplicación web para la gestión de equipos fue un éxito, influye

satisfactoriamente en el tiempo de atención al cliente, se ha cumplido con los requerimientos por parte del usuario; reduciendo del tiempo en la solicitud y asignación de equipos.

Las evidencias de la investigación concuerdan con el trabajo realizado por Huamán y Huayanca (2017) quienes sostienen que el sistema de información cumple los objetivos de la organización en la gestión de equipos, con la implementación del sistema se reduce y automatiza los tiempos en la atención del cliente, procesos, sin perder la información relevante de la empresa. Se corrobora con los hallazgos del estudio realizado por Jimeno y Visitación (2019) donde el sistema web logra optimizar los tiempos de atención al cliente, el usuario tiene una plataforma web de fácil acceso y entendimiento que gestiona de manera eficiente la información.

Así mismo, se evidencia que la aplicación web y móvil permiten mejorar los procesos de pedidos y cotizaciones en la empresa EuroTubo (Ramírez e Inga, 2019), en concordancia con la investigación realizada sobre el desarrollo del sistema informático web para la gestión del proceso de alquiler de equipos de construcción civil brinda un número significativo de beneficios a los usuarios finales. Por otro lado, Chávez (2021) sostiene que la planificación, implementación y buenas practicas permiten gestionar adecuadamente la implementación de los proyectos; en tal sentido, el estudio coincide con el trabajo realizado debido que se utilizó una metodología para el desarrollo del sistema informático web obteniendo un producto que satisface a los usuarios; dicha información es corroborada por Takamura (2021) quien deduce un alto nivel de insatisfacción de los empleados con respecto a la herramienta informática percibiendo una actualización del mismo para beneficios directos, ahorro de tiempo y recursos.

Toda implementación del sistema informático web, es una tarea compleja y conlleva muchas habilidades y capacidades; como todo proyecto de software se tuvieron algunas limitaciones técnicas, tales como el establecimiento y comprensión de los requerimientos, las provisiones de hardware y software vacantes en la nube, los requisitos de seguridad y la capacidad de desarrollar una interfaz de usuario intuitiva; además, los plazos establecidos para el desarrollo y los costos asociados influyeron en el desarrollo del proyecto.

Conclusiones

En esta sección se propone una reflexión crítica sobre los resultados presentados y se evalúan los posibles efectos de las conclusiones de esta investigación

- Se alcanzó examinar el proceso de gestión de alquiler de equipos de la empresa y su posterior implementación en un sistema web.
- Sé evidencia el diseño de los módulos para el sistema de gestión de alquileres de equipos de construcción civil, aplicando la metodología SCRUM
- Sé alcanzo construir los componentes de una aplicación web utilizando los servicios Cloud Database y App Service de Microsoft Azure Cloud.

Recomendaciones

El trabajo de investigación ha logrado comprender mejor el desarrollo de un sistema web para el alquiler de equipos de construcción civil empleando servicios de Microsoft Azure; los resultados obtenidos permiten ofrecer una visión más profunda y completa del mismo. Sin embargo, para aprovechar al máximo los hallazgos del estudio, se hace necesario realizar algunas recomendaciones. A continuación, se presentan cinco recomendaciones finales para maximizar el beneficio de los resultados obtenidos.

- Desarrollar más estudios sobre el tema para una mejor comprensión del tema propuesto con la utilización de otras tecnologías y proveedores de información con la finalidad de buscar la reproducibilidad y validez de la investigación.
- Mediante la metodología SCRUM se puede gestionar proyectos de elaboración de software brindando el control de las acciones y tareas.
- A los propietarios, se recomienda el uso de sistemas de informáticos para gestionar eficientemente los procesos negocio conllevando a una mayor productividad de los empleados e imagen institucional para los clientes.

Referencias Bibliográficas

- Baran, P. (1964). *On distributed communications networks*. Santa Monica, CA: Rand Corporation.
- Barba J. (2014). *Diseño y desarrollo web. Análisis de casos*. Valencia – España: Universitat Politècnica d Valencia.
- Boehm, B. (1981). *Software Engineering Economics*. Upper Saddle River, NJ: Prentice-Hall.
- Booch, G., Rumbaugh, J., & Jacobson, I. (2007). *Object-Oriented Analysis and Design with Applications*. Redwood City, CA: Addison-Wesley.
- Boulding, K. (1966). *The Economics of the Coming Spaceship Earth*. H. Jarrett (Ed.). New York, NY: Harper & Row.
- Capterra (12 diciembre 2022). Software de alquiler de equipos. Recuperado: <https://www.capterra.com.ar/directory/31717/equipment-rental/software>.
- Chávez, H. S. (2021). Implementación, control y gestión del área centralizada de alquiler de equipos de la Empresa Gym S.A - Grupo Graña y Montero. (tesis de pregrado, Universidad Ricardo Palma). Recuperado de <https://repositorio.urp.edu.pe/handle/20.500.14138/3982>
- Clements, P., Kazman, R., & Klein, M. (1998). *Software Architecture in Practice*. Reading, MA: Addison-Wesley Professional.
- Codd, E. F. (1970). *A Relational Model of Data for Large Shared Data Banks*. *Communications of the ACM*, 13(6), 377-387.
- Cook, S. A. (1971). *The complexity of theorem-proving procedures*. *Proceedings of the Third Annual ACM Symposium on Theory of Computing* (pp. 151–158). New York, NY: ACM.
- Cuvi, S., & Rosero Castro, P. (2020). *Implementación de una herramienta web para el control de las actividades de alquiler de maquinaria pesada de la empresa ANDESNEGSA*. Milagro – Ecuador: Universidad Agraria del Ecuador.

- Date, C. (1975). *An Introduction to Database Systems*. Reading, MA: Addison-Wesley.
- Davies, D. (1965). *A general service in packet-switched networks*. *Proceedings of the IEE*, 113(4), 535–542.
- DeMarco, T. (1979). *Structured Analysis and System Specification*. Englewood Cliffs, NJ: Yourdon Press.
- Díaz Rodríguez, J. (2021). *Diseño e implementación de un sistema de gestión para alquiler y soporte de cómputo distribuido para la empresa SINAPPTIC SAS*. Bogotá – Colombia: Universidad Distrital Francisco José De Caldas.
- Dijkstra, E. W. (1976). *A Discipline of Programming*. Englewood Cliffs, NJ: Prentice Hall.
- Fowler, M. (1999). *Refactoring: Improving the Design of Existing Code*. Reading, MA: Addison-Wesley.
- Fowler, M. (1999). *Refactoring: Improving the Design of Existing Code*. Boston, MA: Addison-Wesley Professional.
- Fowler, M. (2002). *Patterns of Enterprise Application Architecture*. Reading, MA: Addison-Wesley.
- Gallego Bocanegra, R. (2019). *Implementacion de un sistema para mejorar la gestión de inventarios y la programacion de mantenimiento de los equipos en la empresa Pomahuaca - Jaén*. Piura – Perú: Universidad Nacional de Piura.
- Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). *Design Patterns: Elements of Reusable Object-Oriented Software*. Reading, MA: Addison-Wesley Professional.
- Gómez Fuentes, M., Cervantes Ojeda, J., & González Pérez, P. (2019). *Fundamentos de Ingeniería de Software*. México: Universidad Autónoma Metropolitana.
- Goudie Domínguez, A. (2015). *Evaluación del modelo de gestión de maquinairas y equipos de Astaldi sucursal Chile*. Santiago – Chile: Universidad de Chile.

- Huamán Varas, O., & Huayanca Quispe, C. (2017). *Desarrollo e implementación de un sistema de información para mejorar los procesos de gestión de equipos en la empresa Humaju*. Lima – Perú: Universidad Autónoma del Perú.
- Jimeno Flores, J., & Visitación Castillo, R. (2019). *Diseño e implementación de un sistema web para la gestión del flujo de información en el taller automotriz Autoservicios Aguilar*. Lima – Perú: Universidad Tecnológica del Perú.
- Kleinrock, L. (1976). *Queueing Systems*. New York, NY: Wiley.
- Lapiedra Alcamí, R., Devece Carañana, C., & Guiral Herrando, J. (2011). *Introducción a la gestión de sistemas de información en la empresa*. España: Universitat Jaume.
- Martin, R. C. (2017). *Clean Architecture: A Craftsman's Guide to Software Structure and Design*. Upper Saddle River, NJ: Prentice Hall.
- Microsoft. (abril de 2017). *Documentación de Microsoft SQL*.
- Microsoft. (abril de 2022). *Introducción a .NET Framework*.
- Microsoft. (enero de 2020). *Microsoft Azure Portal*.
- Oz, E. (2008). *Administración de los sistemas de información* (Quinta ed.). México: Cengage Learning Editores.
- Pantaleo, G., & Rinaudo, L. (2015). *Ingeniería de software*. Buenos Aires - Argentina: Alfaomega.
- Pressman, R. (2010). *Ingeniería del software. Un enfoque práctico* (Séptima ed.). México: McGraw-Hill Interamericana.
- Ramírez Haro, A., & Ynga Peña, A. (2019). *Implementación de una aplicación web/móvil para la gestión de pedidos y cotizaciones de la empresa Eurotubo S.A.C*. Cajamarca – Perú: Universidad Privada del Norte.
- Rapoport, A. (1909). *Frequencies of Occurrence and Distributions in Ecological Systems*. Chicago, IL: University of Chicago Press.

- Reinertsen, D. (1997). *The Principles of Product Development Flow*. Redmond, WA: Celeritas Publishing.
- Salhuana Mendoza, A. (2019). *Aplicación web para la gestión de equipos en Technical Training Center*. Lima – Perú: Universidad Inca Garcilaso de la Vega.
- Schwaber, K., & Sutherland, J. (2017). *The Scrum Guide*. Scrum.org.
- Schwaber, K., & Sutherland, J. (2020). *La Guía Scrum. La Guía Definitiva de Scrum: Las Reglas del Juego*. USA: Creative Commons.
- Takamura, Y. P. (2021). Propuesta de implementación de un sistema web de gestión administrativa de la empresa constructora Kamage contratistas generales S.R.L. - Piura; 2021. (Universidad Católica Los Ángeles de Chimbote). Recuperado <https://repositorio.uladech.edu.pe/handle/20.500.13032/25157>
- Turing, A. M. (1936). *On computable numbers, with an application to the Entscheidungsproblem*. *Proceedings of the London Mathematical Society*, 2(42), 230–265.
- Valiant, L. G. (1979). *The complexity of enumeration and reliability problems*. *SIAM Journal on Computing*, 8(3), 410–421.
- Vera Yáñez, C. (2019). *Desarrollo e implementación de un sistema web para el control de inventario y alquiler de maquinarias de la empresa Megarent S.A.* Guayaquil – Ecuador: Universidad Politécnica Salesiana.
- Von Neumann, J. (1945). *First Draft of a Report on the EDVAC*. En C. E. Shannon y N. Wiener (eds.), *Cybernetics* (pp. 63–80). Cambridge, MA: MIT Press.
- Wiener, N. (1948). *Cybernetics*. Cambridge, MA: MIT Press.

Anexo 2: Matriz de Consistencia

Aplicación web de gestión del alquiler de equipos de construcción civil para la empresa Roosveth Servicios Generales E.I.R.L.

PROBLEMA	OBJETIVOS	HIPÓTESIS
Problema Principal: ¿Cómo desarrollar una aplicación web de calidad para la gestión de alquiler de equipos de construcción civil que garanticen la entrega continua de la misma?	Objetivo General Desarrollar una aplicación web de calidad para la gestión de alquiler de equipos de construcción civil asegurando la entrega continua del producto Objetivos Específicos <ul style="list-style-type: none">● Analizar el proceso actual de gestión de alquiler de equipos de la empresa.● Diseñar los módulos para el sistema de gestión de alquileres de equipos de construcción civil.● Construir una aplicación web de calidad utilizando los servicios Cloud Database y App Service de Microsoft Azure Cloud.● Realizar las pruebas de la aplicación web de gestión de alquileres de equipos de construcción civil.	Hipótesis General La implementación de una aplicación web mejora la gestión de alquiler de equipos de construcción civil en la empresa Roosveth Servicios Generales E.I.R.L.

Anexo 3 :

ENCUESTA DE SATISFACCION DEL USUARIO DE ROOSVETH SERVICIOS GENERALES.

Con el propósito de determinar la importancia y calidad de la aplicación web de alquiler de equipos “ROOSVETH”, deseamos conocer su punto de vista. El éxito de dicho proceso depende de su objetividad y colaboración.

Solicitamos tener en cuenta la siguiente escala establecida para su valoración.

- | | |
|-----------------------------|--------------------------|
| 1. Muy en desacuerdo | 4. De acuerdo |
| 2. En desacuerdo | 5. Muy de acuerdo |
| 3. No estoy seguro | |

Cargo que ocupa en la empresa:

Años de experiencia en el cargo:

Para responder señale con X la opción o la casilla elegida por usted.

1	La aplicación se encuentra siempre disponible para realizar el alquiler de equipos.	1	2	3	4	5
2	La información que me brinda la aplicación web es confiable.	1	2	3	4	5
3	Las consultas y reportes que me brinda la aplicación son exactas y no se presentan inconsistencias.	1	2	3	4	5
4	La información que se gestiona en la aplicación es de suma importancia para mi trabajo diario; sin el apoyo de este sistema no podría realizar mis labores.	1	2	3	4	5
5	Muchas veces el cumplimiento de mis responsabilidades se ven afectados porque la aplicación no me proporciona información confiable y oportuna.	1	2	3	4	5
6	Las funcionalidades ofrecidas por la aplicación apoyan de manera completa los procesos propios de la Oficina de Administración y Tesorería.	1	2	3	4	5
7	La aplicación contiene funciones que abarcan las necesidades particulares de la empresa Roosveth Servicios Generales que la Oficina de Administración y Tesorería debe llevar a cabo	1	2	3	4	5
8	La navegación en la aplicación es fácil.	1	2	3	4	5
9	La búsqueda de información y selección de los elementos en la aplicación es sencilla.	1	2	3	4	5
10	El Registro o modificación de la información de un alquiler se realiza de manera sencilla.	1	2	3	4	5
11	La apariencia de la aplicación en general es estética y agradable, facilitando el trabajo cotidiano.	1	2	3	4	5
12	Para operar la aplicación se requiere hacer una capacitación extensa y un continuo acompañamiento del desarrollador de la aplicación.	1	2	3	4	5
13	La manera como se comunica la aplicación conmigo en la medida que trabajo con él (mensajes, advertencias, etc.) es entendible.	1	2	3	4	5
14	La documentación de ayuda que tiene la aplicación es la apropiada.	1	2	3	4	5
15	La aplicación presenta errores continuamente mientras se opera en ella.	1	2	3	4	5
16	Cuando se solicita información a la aplicación, ésta despliega dicha información en el tiempo esperado.	1	2	3	4	5

.						
17	Considero que la aplicación es un activo para la Corporación Universitaria Adventista.	1	2	3	4	5
18	Desde el inicio de mis labores con la aplicación, ha tenido una evolución continua y de mejora progresiva	1	2	3	4	5
19	En general me encuentro satisfecho con la aplicación.	1	2	3	4	5
20	El desarrollador de la aplicación da una respuesta oportuna y acertada a las sugerencias de cambio.	1	2	3	4	5

REPOSITORIO INSTITUCIONAL DIGITAL

FORMULARIO DE AUTORIZACIÓN PARA LA PUBLICACIÓN DE DOCUMENTOS DE INVESTIGACIÓN

1. Información del Autor				
JIMENEZ Hoyos ESMILTON		42265372	2820220034@usp-san-pedro.edu.pe	
Apellidos y Nombre		DNI	Correo Electrónico	
2. Tipo de Documento de Investigación				
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Trabajo	Trabajo de Difusión Profesional	Trabajo Académico	Trabajo de Investigación	
3. Grado Académico o Título Profesional *				
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Bachiller	Título Profesional	Título Superior Especialidad	Maestría	Doctorado
4. Título del Documento de Investigación				
Aplicación Web de gestión de alquiler de equipos de construcción civil para la empresa ROOSEVELT Servicios Generales E.I.R.L., Cajamarca, 2022.				
5. Programa Académico				
Programa de Ingeniería Informática y de Sistemas.				
6. Tipo de Acceso al Documento				
<input checked="" type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>	
Abierto o Público * (debe ser respaldado por el Autor)	Acceso restringido * (solo en repositorio institucional de la Universidad)			
(*) En caso de restringir cualquier medio				

A. Originalidad del Archivo Digital

Por el presente dejo constancia que el archivo digital que entrego a la Universidad, es la versión final del trabajo de investigación sustentado y aprobado por el Jurado Evaluador y forma parte del proceso que conduce a obtener el grado académico o título profesional.

B. Otorgamiento de una licencia CREATIVE COMMONS *

El autor, por medio de este documento, autoriza a la Universidad, publicar su trabajo de investigación en formato digital en el Repositorio Institucional Digital, al cual se podrá acceder, preservar y difundir de forma libre y gratuita, de manera íntegra a todo el documento. *

Lugar: Chimbote Día: 08 Mes: 11 Año: 2023

Firma Digital  

Referencias

1. Dirección General de Gestión de Recursos Humanos, (2019). Reglamento del Registro Nacional de Trabajos de Investigación en el Perú. Lima: Universidad San Pedro.
2. Ley N° 27302 que aprueba el Reglamento del Repositorio Institucional de la Universidad San Pedro. Lima: Universidad San Pedro.
3. WIKI. (2022). El tipo de acceso abierto. Consultado el 08 de noviembre de 2022. Disponible en: https://www.wiki.org/wiki/El_tipo_de_acceso_abierto
4. INEEL. (2022). Metodología de Evaluación de la Calidad de los Trabajos de Investigación. Lima: Universidad San Pedro.
5. Ley N° 27302 que aprueba el Reglamento del Repositorio Institucional de la Universidad San Pedro. Lima: Universidad San Pedro.
6. Reglamento del Repositorio Institucional de la Universidad San Pedro. Lima: Universidad San Pedro.

IMP - 01/2023 de Modificación de Datos de Información de la Universidad San Pedro en 22 de noviembre de 2023

REPORTE DE SIMILITUD

Aplicación web de gestión del alquiler de equipos de construcción civil para la empresa Roosveth Servicios Generales E.I.R.L, Cajamarca, 2022

INFORME DE ORIGINALIDAD



FUENTES PRIMARIAS

1	repositorio.usanpedro.edu.pe Fuente de Internet	4%
2	hdl.handle.net Fuente de Internet	1%
3	elcajondewatson.blogspot.mx Fuente de Internet	1%
4	repositorio.uladech.edu.pe Fuente de Internet	1%
5	docero.mx Fuente de Internet	1%
6	dokumen.pub Fuente de Internet	1%
7	repositorio.utc.edu.ec Fuente de Internet	1%
8	www.coursehero.com Fuente de Internet	1%

9	Submitted to Universidad Privada San Pedro Trabajo del estudiante	1 %
10	Submitted to Corporación Universitaria Minuto de Dios, UNIMINUTO Trabajo del estudiante	1 %
11	repositorio.urp.edu.pe Fuente de Internet	<1 %
12	docplayer.es Fuente de Internet	<1 %
13	ribuni.uni.edu.ni Fuente de Internet	<1 %
14	redi.unjbg.edu.pe Fuente de Internet	<1 %
15	repositorio.unp.edu.pe Fuente de Internet	<1 %
16	qdoc.tips Fuente de Internet	<1 %
17	repositorio.ulasamericas.edu.pe Fuente de Internet	<1 %
18	Submitted to Pontificia Universidad Católica del Perú Trabajo del estudiante	<1 %
19	Submitted to Universidad del Istmo de Panamá Trabajo del estudiante	<1 %

20	inba.info Fuente de Internet	<1 %
21	repositorio.unan.edu.ni Fuente de Internet	<1 %
22	Submitted to Institución Universitaria Digital de Antioquia Trabajo del estudiante	<1 %
23	www.researchgate.net Fuente de Internet	<1 %
24	repositorio.upn.edu.pe Fuente de Internet	<1 %
25	1library.co Fuente de Internet	<1 %
26	Submitted to City University of Hong Kong Trabajo del estudiante	<1 %
27	sriagral.uabc.mx Fuente de Internet	<1 %
28	repositorio.ucv.edu.pe Fuente de Internet	<1 %
29	baixardoc.com Fuente de Internet	<1 %
30	repositorio.upp.edu.pe Fuente de Internet	<1 %
31	Submitted to Universidad Andina del Cusco Trabajo del estudiante	

		<1 %
32	dspace.udla.edu.ec Fuente de Internet	<1 %
33	Submitted to Universidad Nacional Abierta y a Distancia, UNAD,UNAD Trabajo del estudiante	<1 %
34	www.slideshare.net Fuente de Internet	<1 %
35	es.slideshare.net Fuente de Internet	<1 %
36	expeditiorepositorio.utadeo.edu.co Fuente de Internet	<1 %
37	repositorio.unsm.edu.pe Fuente de Internet	<1 %
38	www.lead-innovation.com Fuente de Internet	<1 %
39	Submitted to Universidad Catolica Los Angeles de Chimbote Trabajo del estudiante	<1 %
40	pt.slideshare.net Fuente de Internet	<1 %
41	www.dspace.uce.edu.ec:8080 Fuente de Internet	<1 %

42	www.itu.int Fuente de Internet	<1 %
43	doku.pub Fuente de Internet	<1 %
44	empiezoinformatica.wordpress.com Fuente de Internet	<1 %
45	paxful.zendesk.com Fuente de Internet	<1 %
46	Submitted to Universidad Cesar Vallejo Trabajo del estudiante	<1 %
47	github.com Fuente de Internet	<1 %
48	(Carlinda Leite and Miguel Zabalza). "Ensino superior: inovação e qualidade na docência", Repositório Aberto da Universidade do Porto, 2012. Publicación	<1 %
49	alicia.concytec.gob.pe Fuente de Internet	<1 %
50	www.cnblogs.com Fuente de Internet	<1 %
51	documentop.com Fuente de Internet	<1 %

52	Submitted to Universidad Peruana Austral del Cusco Trabajo del estudiante	<1 %
53	Submitted to Universidad de Cundinamarca Trabajo del estudiante	<1 %
54	repositorio.continental.edu.pe Fuente de Internet	<1 %
55	repositorio.utelesup.edu.pe Fuente de Internet	<1 %
56	gitlab.citius.usc.es Fuente de Internet	<1 %
57	renati.sunedu.gob.pe Fuente de Internet	<1 %
58	repositorio.unapiquitos.edu.pe Fuente de Internet	<1 %
59	repository.eafit.edu.co Fuente de Internet	<1 %
60	www.atlassian.com Fuente de Internet	<1 %
61	www.gestiopolis.com Fuente de Internet	<1 %
62	www.issa.int Fuente de Internet	<1 %
63	Submitted to Santa Clara University	

Trabajo del estudiante

<1 %

64 de.slideshare.net

Fuente de Internet

<1 %

65 ebin.pub

Fuente de Internet

<1 %

66 es.unionpedia.org

Fuente de Internet

<1 %

67 portal.astic.es

Fuente de Internet

<1 %

68 publicaciones.usanpedro.edu.pe

Fuente de Internet

<1 %

69 repositorio.ug.edu.ec

Fuente de Internet

<1 %

70 repositorio.usil.edu.pe

Fuente de Internet

<1 %

71 reunir.unir.net

Fuente de Internet

<1 %

72 www.dspace.uce.edu.ec

Fuente de Internet

<1 %

73 www.gazafatonarioit.com

Fuente de Internet

<1 %

Excluir citas

Apagado

Excluir coincidencias < 10 words

Excluir bibliografía

Activo