

UNIVERSIDAD NACIONAL DEL ALTIPLANO

FACULTAD DE INGENIERÍA ESTADÍSTICA E INFORMÁTICA

ESCUELA PROFESIONAL DE INGENIERÍA ESTADÍSTICA E INFORMÁTICA



**IMPLEMENTACIÓN DE SISTEMA DE ADMINISTRACIÓN DE
CONTENIDOS WEB USANDO TECNOLOGÍAS INTERNET
INFORMATION SERVER - IIS Y ASP.NET PARA LA UNIDAD DE
GESTIÓN EDUCATIVA LOCAL CANAS CUSCO – UGEL CANAS
313**

TESIS

PRESENTADA POR:

BACH. DELIA APAZA CAMPOS

**PARA OPTAR EL TÍTULO PROFESIONAL DE:
INGENIERO ESTADÍSTICO E INFORMÁTICO**

PUNO – PERÚ

2019

UNIVERSIDAD NACIONAL DEL ALTIPLANO
FACULTAD DE INGENIERÍA ESTADÍSTICA E INFORMÁTICA
ESCUELA PROFESIONAL DE INGENIERÍA ESTADÍSTICA E INFORMÁTICA

IMPLEMENTACIÓN DE SISTEMA DE ADMINISTRACIÓN DE CONTENIDOS WEB USANDO TECNOLOGÍAS INTERNET INFORMATION SERVER - IIS Y ASP.NET PARA LA UNIDAD DE GESTIÓN EDUCATIVA LOCAL CANAS CUSCO – UGEL CANAS 313

TESIS PRESENTADA POR:

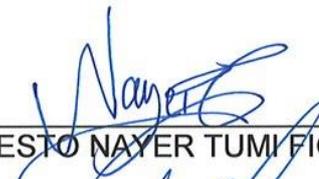
Bach. DELIA APAZA CAMPOS

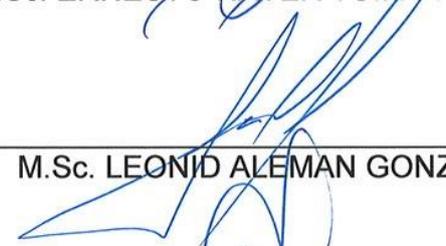


PARA OPTAR EL TÍTULO PROFESIONAL DE:

INGENIERO ESTADÍSTICO E INFORMÁTICO

APROBADO POR EL JURADO REVISOR CONFORMADO POR:

PRESIDENTE : 
 M.Sc. ERNESTO NAYER TUMI FIGUEROA

PRIMER MIEMBRO : 
 M.Sc. LEONID ALEMAN GONZALES

SEGUNDO MIEMBRO : 
 M.Sc. ANGEL JAVIER QUISPE CARITA

DIRECTOR DE TESIS : 
 M.Sc. ELQUI YEYE PARI CONDORI

Área: sistemas de Información
Tema: Sistemas de administracion

Fecha de sustentación: 16 de octubre del 2019

DEDICATORIA

El presente tesis está dedicado primeramente a DIOS por su infinita bondad y amor, por ser mi amparo y fortaleza por hacer posible que hoy este aquí y poder cumplir una de mis retos.

Se lo dedico a mi papitos BENITO y ESTELITA por su amor incondicional, por haberme dado un hogar en el que siempre prevaleció el amor y el respeto por ser modelo de esfuerzo y lucha constante .Sin ellos no hubiese sido posible cumplir mis sueños. Abusivo

Se los dedico a mis hermanos FLAVIO, CARMEN Y MARIBEL quienes con sus palabras y sus consejos me impulsan a seguir adelante, tener en cuenta que nada es imposible, mientras exista una esperanza.

Se les dedico a mis sobrinos MARIBEL, RODRIGO YAMIRA Y HEYDI y mi compañero de vida BRANDON que con solo verlos me impulsan a seguir adelante, son por ellos que debo seguir cumpliendo mis metas para que ellos cumplan los suyos, al igual que yo lo estoy haciendo el día de hoy, ellos son y serán mi motor en mi vida.

AGRADECIMIENTO

Agradezco a la UNIVERSIDAD NACIONAL DEL ALTIPLANO – PUNO y la FACULTAD DE INGENIERÍA ESTADÍSTICA E INFORMÁTICA por haberme aceptado ser parte de ella .Para así podernos preparar para un futuro competitivo y formándonos como personas de bien.

Agradezco a mi asesor M.Sc. ELQUI YEYE PARI CONDORI por haberme brindado la oportunidad de recurrir a su capacidad, conocimiento y experiencia y tenido toda la paciencia del mundo para guiarme durante todo el desarrollo de mi tesis.

Agradezco al Prof. Mario Zavaleta Quispe DIRECTOR de la UNIDAD DE GESTIÓN EDUCATIVA LOCAL DE CANAS también al ING: William Rene Quispe Aquino RESPONSABLE DE LA OFICINA DE ESTADÍSTICA por haberme apoyado incondicionalmente y brindarme todo las facilidades requeridas de mi persona estoy muy agradecida con los resultados que se ha obtenido.

Al jurado calificador de tesis; M.Sc ERNESTO NAYER TUMI FIGUEROA, M.Sc LEONID ALEMAN GONZALES, M.Sc ANGEL JAVIER QUISPE CARITA M.Sc. ELQUI YEYE PARI CONDORI, por su comprensión, consejos, tiempo y sus correcciones hechas a lo largo de estos meses.

ÍNDICE GENERAL

	Pág.
ÍNDICE DE FIGURAS	
ÍNDICE DE TABLAS	
ÍNDICE DE ACRÓNIMOS	
RESUMEN	12
ABSTRACT	13

CAPÍTULO I**INTRODUCCIÓN**

1.1. PLANTEAMIENTO DEL PROBLEMA	14
1.2. JUSTIFICACIÓN	15
1.3. OBJETIVOS	16
1.3.1. Objetivos Generales.....	16
1.3.2. Objetivos Específicos.....	16
1.4. HIPÓTESIS.....	16

CAPÍTULO II**REVISIÓN DE LITERATURA**

2.1. ANTECEDENTES	17
2.2. MARCO TEÓRICO	19
2.2.1. Desarrollo de Sistemas.....	19
2.2.2. Sistemas de Información	20
2.2.3. WEB	21
2.2.4. Los lenguajes de programación para la WEB.....	23
2.2.5. Metodología de desarrollo de software	25

CAPÍTULO III**MATERIALES Y MÉTODOS**

3.1. UBICACIÓN DE LA POBLACIÓN	36
3.2. POBLACIÓN	36
3.3. MUESTRA	36

3.4.	MÉTODO DE RECOLECCIÓN DE DATOS	36
3.5.	DISEÑO ESTADÍSTICO A UTILIZAR	37
3.6.	METODOLOGÍA DE DESARROLLO	37

CAPÍTULO IV

RESULTADOS Y DISCUSIÓN

4.1.	ANÁLISIS DEL SISTEMA	41
4.1.1.	Análisis de requisitos del sistema	41
4.1.2.	Funcionalidades, requisitos de tipo de usuario	42
4.1.3.	Casos de uso para el administrador y/o usuario	43
4.1.4.	Funcionalidades, del estadístico y director	45
4.1.5.	Casos de uso para el usuario y/o estadístico.....	45
4.2.	DISEÑO	47
4.3.	DESARROLLO	49
4.4.	PRUEBA	50
4.5.	PROCESAMIENTO DE DATOS SEGÚN LAS ENCUESTAS	52
4.6.	MÉTRICAS DE LA FUNCIÓN	56
CONCLUSIONES		60
RECOMENDACIONES		60
REFERENCIAS BOBLOGRÁFICAS.....		62
ANEXOS.....		65
ANEXO A: ENCUESTA AL PERSONAL.....		66
ANEXO B: FICHA DE EVALUACION DEL ISO 9126		68
ANEXO C: CODIGO FUENTE: ASP WEB CONFIG.....		72
ANEXO D: CODIGO FUENTE: CONTROLADOR WEB		73
ANEXO E: CODIGO FUENTE: CONTROLADOR NOTICIAS		76
ANEXO F: CODIGO FUENTE: CONTROLADOR ADMIN		78
ANEXO G: CODIGO FUENTE: CONTROLADOR SESIONES.....		85



ANEXO H: CODIGO FUENTE: CONTROLADOR HTTP REQUESTS	87
ANEXO I: MANUAL DE USUARIO	90

ÍNDICE DE FIGURAS

Figura 1. Etapas en el Desarrollo de Sistemas de Información.....	19
Figura 2. Modelo General de un Sistema.....	20
Figura 3. Diagrama de Casos de Uso del Administrador de usuario.....	43
Figura 4. Diagrama de Secuencia del Administrador de Usuario.....	44
Figura 5. Diagrama de colaboración del administrador de usuario	44
Figura 6. Diagrama de Casos del Sistema de Administración	45
Figura 7. Diagrama de Secuencia del Sistema de Administración.....	46
Figura 8. Diagrama de Colaboración del Sistema de Administración	46
Figura 9. Diagrama de Clases del Sistema de Administración de Contenidos Web.....	47
Figura 10. Acceso a la Base de Datos del Sistema de Administración de Contenidos Web.....	48
Figura 11. Tablas con que se Trabajó la Base de Datos Base de Datos, del Sistema de Administración de Contenidos Web.....	48
Figura 12. Interfaz del Sistema de Administración de Contenidos Web. ugelcanas.gob.pe.....	49
Figura 13. Pantalla Principal del Sistema de Administración de Contenidos web Canas – Cusco 2019.....	49
Figura 14. Noticias y Enlaces Importantes vinculados a Educación en la Ugel Canas – Cusco 2019.....	50
Figura 15. Distribución del Personal Según la Calificación del Interfaz Ergonómico y Adecuado, en la Ugel Canas 2019.....	53
Figura 16. Distribución del Personal Según si Ayudara en Informarse de la Institución, en la Ugel Canas 2019.....	54

Figura 17. Distribución del Personal Según como Obtenían Información antes,
en la Ugel Canas 2019..... 55

Figura 18. Distribución del Personal según como Obtienen Información ahora,
en la Ugel Canas 2019..... 56

ÍNDICE DE TABLAS

Tabla 1. Metodologías Agiles y Metodologías Tradicionales	29
Tabla 2. Distribución del Personal según la Calificación del Interfaz Ergonómico y Adecuado, en la Ugel Canas 2019	52
Tabla 3. Distribución del Personal Según si Ayudara en Informarse de la Institución, en la Ugel Canas 2019.....	53
Tabla 4. Distribución del Personal Según como Obtenían Información antes, en la Ugel Canas 2019.....	54
Tabla 5. Distribución del Personal según como Obtienen Información ahora, en la Ugel Canas 2019.....	55
Tabla 6. Parámetros de medición	57
Tabla 7. Valores de ajuste de la complejidad del sistema.....	58
Tabla 8. Escala Valorativa de la Calificación del Sistema de.....	59

ÍNDICE DE ACRÓNIMOS

ASP.NET	: Open-source web framework for .NET
ISO 9126	: estándar internacional para la evaluación de la calidad del Software
MVC	: Modelo Vista Controlador
SQL	: Structured Query Language - Lenguaje de Consulta Estructurada
UGEL	: Unidad de Gestión Educativa Local
UML	: Lenguaje Unificado de Modelamiento
XP	: Metodología extrema

RESUMEN

El presente trabajo de investigación, implementación de sistema de administración de contenidos web usando tecnologías internet Información Server - IIS y Asp.net para la Unidad de Gestión Educativa Local Canas Cusco – UGEL Canas 313, tiene como objetivo, Implementar un sistema de administración de contenidos web usando tecnologías Internet Información Server – IIS y ASP.Net para la UGEL Canas Cusco 313, el presente trabajo se realizó en la provincia de Canas, región Cusco, donde tiene 8 distritos, donde se utilizó la metodología de desarrollo de software la programación extrema XP, es una metodología muy rápida y ligera de desarrollo de software que se basa en la simplicidad, la comunicación y la realimentación o reutilización del código desarrollado, Se utilizó el lenguaje asp.net, para realizar el interfaz y la base de datos se trabajó en SQL Server; llegando a concluir que el sistema de administración de mejora eficazmente la comunicación, la administración de contenidos y servicios académicos y sociales que presta a los docentes de Educación Básica de la Unidad de Gestión Educativa Canas Cusco. Según los reportes estadísticos, gráficos, telecomunicaciones e informaciones académicos y sociales mediante al administración del sistema web.

Palabras Clave: Administración de contenidos, ASP.NET, UGEL Canas, Tecnología Web.

ABSTRACT

This research work, implementation of web content management system using Internet technologies Information Server - IIS and Asp.net for the Educational Management Unit Canas Cusco – UGEL Canas 313, aims, to implement a web content management system using Internet Information Server technologies – IIS and ASP.Net for UGEL Canas Cusco 313, the present work was carried out in the province of Canas, Cusco region, where it has 8 districts, where the methodology of software development extreme XP programming, is a very fast and lightweight methodology of software development that is based on simplicity, communication and feedback or reuse of developed code, The language asp.net, was used to perform the interface and database worked on SQL Server; concluding that the management system effectively improves communication, content management and academic and social services that it provides to Basic Education teachers of the Canas Cusco Educational Management Unit. According to statistical reports, graphs, telecommunications and academic and social information through the administration of the web system.

Keywords: Content management, ASP.NET, UGEL Canas, Web Technology.

CAPÍTULO I

INTRODUCCIÓN

Con la introducción de nuevas tecnologías, la mayoría de las Instituciones se ven en la necesidad de implementar sistemas de innovación y administración de últimas tecnologías para poder optimizar al máximo los diferentes servicios, del mismo modo la Unidad de Gestión Educativa Local Canas Cusco – UGEL Canas 313, como Institución competente y de prestigio requiere contar con sistemas modernos en este caso con un Sistema de administración de contenidos web. Muchos sistemas de información poseen información confidencial que sólo algunas personas deben de conocer o su acceso debe ser restringido. Los sistemas de administración presentan una solución a este problema.

El desarrollo de sistemas de información es una herramienta necesaria en las empresas e Instituciones Educativas puesto que ayuda a tomar decisiones, comunicación, información y comunicado a la población estudiantil, profesionales y público en general en su Región y a nivel Nacional e Internacional, también mejora el control administrativo y en tiempo real; la estructura del trabajo se presenta a continuación:

En el capítulo I, se presentan los aspectos generales relacionados al problema y justificación del trabajo de investigación en esta área.

En el capítulo II, Es la parte de la revisión literaria, antecedentes y marco teórico de sistemas de información, los lenguajes de programación y la metodología de desarrollo de software.

En el capítulo III, mostramos el tipo y diseño de investigación, ubicación de la población y el desarrollo de la metodología Programación Extrema (XP) y empleando el Lenguaje Unificado de Modelamientos (UML).

En el capítulo IV, se muestra los resultados de la investigación, la parte de análisis de requerimiento, diseño, implementación y evaluación del software. El sistema es de fácil manejo y de una interfaz amigable para todos.

1.1. PLANTEAMIENTO DEL PROBLEMA

Las oficinas de la Unidad de Gestión Educativa Local Canas Cusco sus comunicados, convocatorias, otros enlaces referidos a información y enlaces de direcciones por Internet con otras Instituciones no tenían en un sistemas virtual, que puedan informarse rápidamente, solo usaban volantes y afiches.

Hoy en día la informática es una herramienta indispensable que ha encontrado importantes aplicaciones en las actividades humanas, lo cual nos ofrece sistemas que permiten en forma rápida, ordenada y concreta en el manejo de diversos tipos de información De hecho, en contenidos web. Mediante noticias, comunicados, información de la Institución, que antes era tediosa para hacer conocer a los docentes y a la población en general.

Por lo que el presente trabajo está realizado con la finalidad de mejorar este servicio y así brindar una mejor atención, frente a esto nos planteamos:

¿ayudara la implementación de sistema de administración de contenidos web usando tecnologías internet Información Server - IIS y ASP.net en la Unidad de Gestión Educativa Canas Cusco – UGEL Canas 313?

1.2. JUSTIFICACIÓN

Las actuales arquitecturas de desarrollo están potenciado el desarrollo de aplicaciones web con la reducción del tiempo de desarrollo y el establecimiento de un estándar parcial de desarrollo para equipos de desarrolladores, actualmente permite el desarrollo de aplicaciones de propósito general, así como el procesado del lenguaje sea PHP o de ASP.NET, del cual beneficiara al personal administrativo en la administración de contenidos.

Mediante el sistema de administración de contenidos para la publicación mantenimiento y edición de diversos elementos, esto genera un lapso menor en la distribución de información y noticias a los usuarios en este caso profesores de la UGEL. Entonces nuestra propuesta obtenida después de una revisión y evaluación desarrollada íntegramente sobre el Framework MVC de ASP.NET Core 2.1, que revisamos y realizamos las pruebas necesarias para poder ofrecer una amplia gama de soluciones que serán propuestas por la institución. Finalmente se espera poder ofrecer en este trabajado de investigación una solución a los problemas de distribución de información electrónica de la institución.

1.3. OBJETIVOS

1.3.1. Objetivos Generales

Implementar un sistema de administración de contenidos web usando tecnologías Internet Information Server – IIS y ASP.Net para la UGEL Canas Cusco 313.

1.3.2. Objetivos Específicos

- Analizar los requerimientos arquitectónicos de desarrollo que permita la comunicación Web con los profesores de la UGEL.
- Diseñar el modelo de composición usando tecnologías internet Information Server - IIS y Asp.net para mejorar la administración de contenidos web la Unidad de Gestión Educativa Canas Cusco – UGEL Canas 313.
- Implementar la arquitectura basado mediante las tecnologías internet Information Server - IIS y Asp.net para mejorar la administración de contenidos web la Unidad de Gestión Educativa Canas Cusco – UGEL Canas 313.
- Evaluar el sistema usando tecnologías internet Information Server - IIS y Asp.net para mejorar la administración de contenidos web la Unidad de Gestión Educativa Canas Cusco – UGEL Canas 313.

1.4. HIPÓTESIS

- El sistema de la administración de contenidos web nos permite que la información sea rápida y ordenada y concreta, en beneficiosos para los profesores y público en general de la unidad de gestión educativa canas

CAPÍTULO II

REVISIÓN DE LITERATURA

2.1. ANTECEDENTES

AGUILAR GONZÁLEZ, I. E., GUZMÁN VÁSQUEZ, S. L., (2018) **Sistema Informático de Gestión Administrativa de la Secretaría de Cultura de San Salvador** donde concluye Que, con el análisis de la situación actual en Secretaria de Cultura, se logró la comprensión de los procesos que se realizan, permitiendo identificar oportunidades de mejora., surgiendo la necesidad de un Sistema Informático que permita ayudar al personal en la gestión de la información.

PASTOR RAMIREZ (2017) **Modelo para la generación de cursos virtuales usando tecnologías de la web semántica para sistemas de gestión de aprendizaje**, donde concluye que Las tecnologías de la web semántica han influido de forma positiva en muchas áreas de la gestión del conocimiento, especialmente en el área de la educación, y a su vez dentro de esta área se han identificado sub-áreas como el aprendizaje adaptativo, objetos de aprendizaje, aprendizaje colaborativo, diseño instruccional y herramientas de autoría. Para cumplir con los objetivos de esta investigación se analizó estudios que han combinado el uso del diseño instruccional con las tecnologías de la web semántica. Esto permitió hacer un análisis más abierto de la literatura, pero a su vez desde una perspectiva integrada

BRAMBILA, D. A. C., & OBANDO, A. L. (2015). **Sistema de evaluaciones en línea como herramienta para los niveles de educación**

media superior, donde concluye que este sistema no busca que los profesores cambien su forma de evaluar, sino que principalmente sean ellos quienes elaboren los exámenes para sus alumnos. Utilizar un sistema como este ahorra tiempo al profesor, quien puede dedicarlo a calificar los exámenes y a mejorar el conocimiento del estudiante gracias a los resultados que este sistema da de manera inmediata.

QUISPE (2014), “**Sistema Informático de Gestión Administrativa para la coordinación de Investigación de la Facultad de Ingeniería Estadística e Informática de la UNA PUNO 2014**”, Cuyo objetivo fue, la implementar un sistema para la gestión administrativa de la coordinación de investigación, de la FINESI. Y concluye que la implementación del sistema SIGACI permite realizar los procesos descritos en menor tiempo

GARCÍA, J., & ABRAHAM, F. (2012) **tiene como objetivo, desarrollo y aplicación de un sistema para soporte a la decisión (DSS), para el proceso de adaptación de contenidos educativos** en un ambiente virtual de aprendizaje de acuerdo al estilo sensorial perceptivo de aprendizaje de los alumnos. Donde concluye que el uso de los sistemas de información en el área de educación como herramienta al modelo tradicional de maestro-alumno donde se produce una triangulación de maestro-Cognitive DSS+LMS-Alumno, brinda la pauta de una nueva forma de trabajo híbrida, fracción presencial, fracción virtual que cambie el paradigma educativo como se conoce y evoluciona hacia la nueva tendencia educativa a nivel superior.

2.2. MARCO TEÓRICO

2.2.1. Desarrollo de Sistemas

El desarrollo de sistemas se define como la actividad consistente en crear o modificar los sistemas de negocios. Los proyectos de desarrollo de sistemas pueden ser pequeños o muy grandes y abarcar campos del conocimiento tan diversos como el análisis bursátil y los videojuegos. Algunos esfuerzos encaminados a desarrollar sistemas han tenido un rotundo éxito.

El desarrollo de sistemas de información que cumplan con las necesidades del negocio representa una tarea muy compleja y difícil, tanto que es muy común que los proyectos relacionados con los sistemas de información se “traguen” el presupuesto y excedan las fechas de terminación programadas. Una estrategia para mejorar los resultados de un proyecto de este tipo consiste en dividirlo en varias etapas, cada una de las cuales debe contar con una meta bien definida y un conjunto de tareas a cumplir, a continuación se muestra un resumen de dichas etapas.

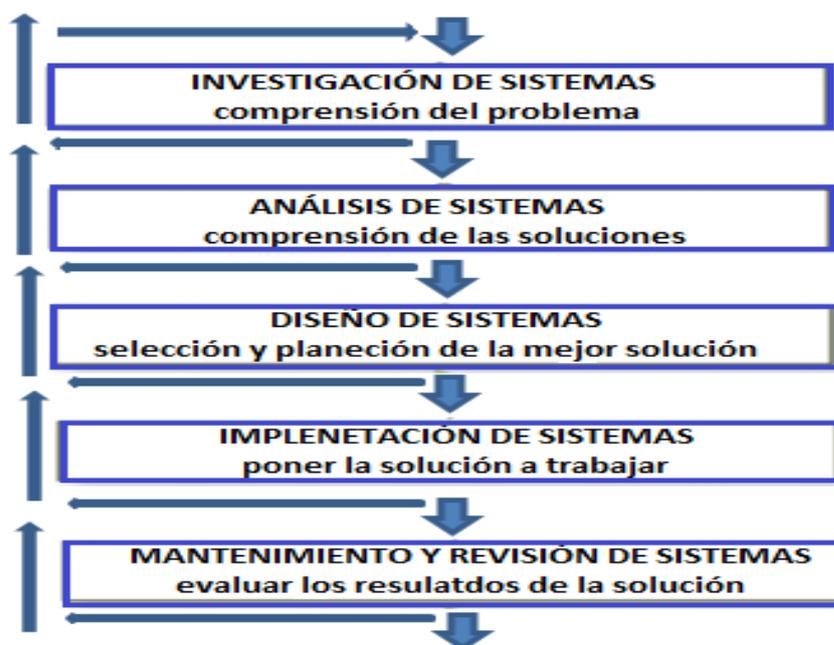


Figura 1. Etapas en el Desarrollo de Sistemas de Información

Fuente: (canós & letelier, 2012)

2.2.2. Sistemas de Información

Sistemas de Información es un conjunto de componentes interrelacionados que reúne (o recupera), procesa, almacena y distribuye información para apoyar la toma de decisiones y el control de la organización. Además de apoyar la toma de decisiones, la coordinación y el control, los sistemas de información también pueden ayudar a los gerentes y trabajadores a analizar problemas, a visualizar asuntos complejos y a crear productos nuevos. Entre las funciones principales que realiza el sistema de Información son: - Captura o recolecta datos tanto externos como internos. - Trata esos datos mediante procesos que operan con ellos. - Distribuye la información resultante a los usuarios y actividades que la requieran. (Valderrama Guayan & Benites Barrientos, 2014)

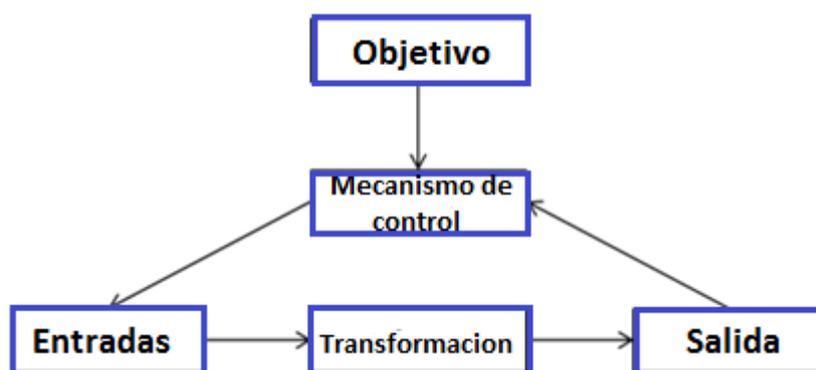


Figura 2. Modelo General de un Sistema

Fuente: (alarcón, 2006)

El modo en que los sistemas de información realizan sus funciones, los recursos con los que cuentan y el tipo de usuarios a los que satisfacen definen el alcance de los mismos.

TIPOS DE SISTEMAS

Se clasifica de acuerdo a los diferentes niveles de la organización: Los Sistemas a Nivel Operativo son Sistemas de Información que supervisan las actividades elementales y transacciones de la organización; los Sistemas a Nivel del Conocimiento apoyan los trabajadores del conocimiento y de datos de una organización; los Sistemas a Nivel Administrativo apoyan las actividades de supervisión, control, de tomas de decisiones y administrativas de los gerentes de nivel medio; y por último, los Sistemas a Nivel Estratégico apoyan las actividades de planeación a largo plazo de la dirección general de la empresa. (Valderrama Guayan & Benites Barrientos, 2014).

2.2.3. WEB

La World Wide Web, más conocida como Web, es una de las aéreas de Internet que se ha desarrollado más rápidamente. Nació en 1989, como parte de un proyecto de CERN de Suiza y con el objetivo de mejorar el intercambio de información dentro de Internet, y vea en lo que se ha convertido actualmente.

De ser un lugar en el que se podía encontrar información, ha pasado a ser un gran centro comercial. En muy pocos años, las sencillas paginas estáticas de la Web han evolucionado hasta convertirse en sofisticados sitios (sites) donde se pueden comprar, de forma segura, bienes y otros servicios. Mientras tanto, han nacido nuevas compañías que realizan sus negocios exclusivamente a través de la Web, Los pilares de la web son:

- **HTML** como lenguaje para crear los contenidos de la Web, basado en Estándar Generalized Markup Language (SGML).
- **HTTP** como protocolo de comunicación entre los ordenadores de la Web, encargado de la transferencia de las páginas web y demás recursos.
- **URL** como medio de localización (direccionamiento) de los distintos recursos en Internet. (Valderrama Guayan & Benites Barrientos, 2014).

El World-Wide Web (WWW) es un sistema hipermedia distribuido, accesible a través de Internet, que permite navegar con facilidad por una enorme cantidad de información. con el objeto de integrar información accesible a través de una única red de ordenadores, pero mediante sistemas diversos. El WWW se sustenta en cuatro elementos fundamentales: un nuevo protocolo de comunicación (HTTP o HyperText Transfer Protocol); Un lenguaje para escribir documentos hipermedia (HTML o Hypertext Markup Language); un sistema notacional para designar objetos en la Internet y las operaciones a realizar sobre ellos (URL o Uniform Resource Locator); y, finalmente, un conjunto de aplicaciones (los clientes o browsers WWW y los servidores httpd) que se dividen el trabajo de servir y presentar la información multimedia al usuario.

Los elementos básicos de la tecnología WWW se han descrito ya en diversas publicaciones en nuestro país (Adell, 1993; Adell y Bellver, 1993 y 1995). El número de servidores WWW instalados crece a un ritmo vertiginoso (en nuestro país superan los 300) y se estima que, a nivel mundial, su número se dobla cada 50 días. La cantidad de información accesible mediante el WWW

está en el orden de terabytes. Su uso masivo se realiza en campos como la comunicación científica, los negocios, el ocio y la educación. (Adell, 1995)

El término Web 2.0 hace referencia a la evolución que ha experimentado el servicio web. En constante progresión, ha pasado de unas primeras páginas estáticas en HTML (Web 1.0), a un segundo nivel más elaborado (Web 1.5), caracterizado por la creación “al vuelo” de documentos dinámicos. Pero los cambios que se intuyen ahora son más profundos y complejos. Este nuevo estadio de la Web es el que se conoce como Web 2.0.(Ribes, 2007)

En la charla inicial del Web Conferencia se habló de los principios que tenían las aplicaciones Web 2.0:

- La web es la plataforma.
- La información es lo que mueve al Internet.
- Efectos de la red movidos por una arquitectura de participación.
- La innovación surge de características distribuidas por desarrolladores independientes.
- El fin del círculo de adopción de software pues tenemos servicios en beta perpetuo(Van Der Henst, 2005).

2.2.4. Los lenguajes de programación para la WEB

El lenguaje PHP. Es un lenguaje creado en el año 1995 por PHP Group, PHP significa Hypertext Pre-processor aunque inicialmente se llamó “Personal Home Page”. PHP es un lenguaje de script que se interpreta en el lado del servidor, se usa para la generación de páginas web dinámicas, embebidas en

páginas HTML y ejecutadas en el servidor. Para su funcionamiento necesita tener instalado un servidor de Apache. La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas. Los archivos cuentan con la extensión (php).

El lenguaje ASP. Es un Lenguaje creado por Microsoft ASP lo que es en inglés “Active Server Pages”, al igual que el PHP sirve para crear páginas web dinámicas. Para que éste funcione de forma correcta necesita tener instalado los servicios de Internet Information Server (IIS)¹ . Su código se ejecuta del lado del servidor. Los archivos cuentan con la extensión (asp).

El Lenguaje ASP.NET. El lenguaje ASP.NET fue desarrollado por Microsoft para resolver las limitantes que tenía el ASP. Para el desarrollo de ASP.NET se puede utilizar C#, VB.NET o J#. Los archivos escritos en ASPx tienen la extensión (aspx). Al igual que el ASP, para el funcionamiento de las páginas se necesita tener instalado IIS con el Framework .Net.

El Lenguaje JSP. Es un lenguaje desarrollado por Sun Microsystems para la creación de sitios Web dinámicos. Se ejecuta del lado del servidor. JSP lo que es en inglés “Java Server Pages”. JSP Comparte características similares a las de ASP.NET, ya que fue desarrollado para la creación de aplicaciones Web potentes.

JSP tiene un motor de páginas basado en los servlets² de Java. Para su funcionamiento JSP necesita tener instalado un servidor Tomcat.(Salazar, Aguirre, & Osorio, 2011).

2.2.5. Metodología de desarrollo de software

➤ **METODOLOGÍAS ÁGILES**

En febrero de 2001, tras una reunión celebrada en Utah-EEUU, nace el término “ágil” aplicado al desarrollo de software. En esta reunión participan un grupo de 17 expertos de la industria del software, incluyendo algunos de los creadores o impulsores de metodologías de software. Su objetivo fue esbozar los valores y principios que deberían permitir a los equipos desarrollar software rápidamente y respondiendo a los cambios que puedan surgir a lo largo del proyecto. Se pretendía ofrecer una alternativa a los procesos de desarrollo de software tradicionales, caracterizados por ser rígidos y dirigidos por la documentación que se genera en cada una de las actividades desarrolladas. (Roger, 2002)

Tras esta reunión se creó The Agile Alliance 3, una organización, sin ánimo de lucro, dedicada a promover los conceptos relacionados con el desarrollo ágil de software y ayudar a las organizaciones para que adopten dichos conceptos. El punto de partida es fue el Manifiesto Ágil, un documento que resume la filosofía “ágil”.(Canós & Letelier, 2012)

El Manifiesto Ágil.

Según el Manifiesto se valora:

- **Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas.** La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir

primero el entorno y esperar que el equipo se adapte automáticamente.

Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.

- **Desarrollar software que funciona más que conseguir una buena documentación.** La regla a seguir es “no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante”. Estos documentos deben ser cortos y centrarse en lo fundamental.

- **La colaboración con el cliente más que la negociación de un contrato.** Se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.

- **Responder a los cambios más que seguir estrictamente un plan.** La habilidad de responder a los cambios que puedan surgir a lo largo del proyecto (cambios en los requisitos, en la tecnología, en el equipo, etc.) determina también el éxito o fracaso del mismo. Por lo tanto, la planificación no debe ser estricta sino flexible y abierta.

Los valores anteriores inspiran los doce principios del manifiesto. Son características que diferencian un proceso ágil de uno tradicional. Los dos primeros principios son generales y resumen gran parte del espíritu ágil. El resto tienen que ver con el proceso a seguir y con el equipo de desarrollo, en cuanto metas a seguir y organización del mismo. Los principios son:

- I. La prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software que le aporte un valor.
- II. Dar la bienvenida a los cambios. Se capturan los cambios para que el cliente tenga una ventaja competitiva.
- III. Entregar frecuentemente software que funcione desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre entregas.
- IV. La gente del negocio y los desarrolladores deben trabajar juntos a lo largo del proyecto.
- V. Construir el proyecto en torno a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos para conseguir finalizar el trabajo.
- VI. El diálogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo.
- VII. El software que funciona es la medida principal de progreso.
- VIII. Los procesos ágiles promueven un desarrollo sostenible. Los promotores, desarrolladores y usuarios deberían ser capaces de mantener una paz constante.
- IX. La atención continua a la calidad técnica y al buen diseño mejora la agilidad.
- X. La simplicidad es esencial.
- XI. Las mejores arquitecturas, requisitos y diseños surgen de los equipos organizados por sí mismos.
- XII. En intervalos regulares, el equipo reflexiona respecto a cómo llegar a ser más efectivo, y según esto ajusta su comportamiento.

COMPARACIÓN

La Tabla 1 recoge esquemáticamente las principales diferencias de las metodologías ágiles con respecto a las tradicionales (“no ágiles”). Estas diferencias que afectan no sólo al proceso en sí, sino también al contexto del equipo así como a su organización. (Poppendieck & Poppendieck, 2003)

Tabla 1. Metodologías Ágiles y Metodologías Tradicionales

METODOLOGÍAS ÁGILES	METODOLOGÍA TRADICIONAL
Basadas en heurísticas provenientes de prácticas de producción de código.	Basadas en normas proveniente de estándares seguidos por el entorno de desarrollo.
Especialmente preparados para cambios durante el proyecto.	Cierta resistencia de los cambios.
Impuestas internamente (por el equipo).	Impuestas externamente.
Proceso menos controlado, con pocos principios.	Proceso mucho más controlado, con numerosas políticas /normas.
No existe contrato tradicional o al menos es bastante flexible.	Existes un contrato prefijado.
El cliente es parte del equipo de desarrollo.	El cliente interactúa con el equipo de desarrollo mediante reuniones.
Grupo pequeño (<10 integrantes) y trabajo en el mismo sitio.	Grupos grandes y posiblemente distribuidos.
Pocos artefactos.	Más artefactos.
Pocos roles.	Más roles.
Menos énfasis en la arquitectura del software.	La arquitectura del software es esencial y se expresa mediante modelos.

Fuente: (Abrahamsson, Salo, Ronkainen, & Warsta, 2017)

PROGRAMACIÓN EXTREMA (EXTREME PROGRAMMING, XP)

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

Los principios y prácticas son de sentido común pero llevadas al extremo, de ahí proviene su nombre. Kent Beck, el padre de XP, describe la filosofía de XP, sin cubrir los detalles técnicos y de implantación de las prácticas. Posteriormente, otras publicaciones de experiencias se han encargado de dicha tarea. A continuación presentaremos las características esenciales de XP organizadas en los tres apartados siguientes: historias de usuario, roles, proceso y prácticas.

✓ **Las Historias de Usuario:** Son la técnica utilizada para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas.

Beck en su libro, presenta un ejemplo de ficha (customer story and task card) en la cual pueden reconocerse los siguientes contenidos: fecha, tipo de actividad (nueva, corrección, mejora), prueba funcional, número de historia, prioridad técnica y del cliente, referencia a otra historia previa, riesgo, estimación técnica, descripción, notas y una lista de seguimiento con la fecha, estado cosas por terminar y comentarios. A efectos de planificación, las historias pueden ser de una a tres semanas de tiempo de programación (para no superar el tamaño de una iteración). Las historias de usuario son descompuestas en tareas de programación (task card) y asignadas a los programadores para ser implementadas durante una iteración.

- ✓ **Roles XP:** Los roles de acuerdo con la propuesta original de Beck son:
- **Programador.** El programador escribe las pruebas unitarias y produce el código del sistema.
 - **Ciente.** Escribe las historias de usuario y las pruebas funcionales para validar su implementación. Además, asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio.
 - **Encargado de pruebas (Tester).** Ayuda al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.
 - **Encargado de seguimiento (Tracker).** Proporciona realimentación al equipo. Verifica el grado de acierto entre las estimaciones realizadas y

el tiempo real dedicado, para mejorar futuras estimaciones. Realiza el seguimiento del progreso de cada iteración.

- **Entrenador (Coach).** Es responsable del proceso global. Debe proveer guías al equipo de forma que se apliquen las prácticas XP y se siga el proceso correctamente.
- **Consultor.** Es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto, en el que puedan surgir problemas.
- **Gestor (Big boss).** Es el vínculo entre clientes y programadores, ayuda a que el equipo trabaje efectivamente creando las condiciones adecuadas. Su labor esencial es de coordinación.

✓ **Proceso XP:** El ciclo de desarrollo consiste (a grandes rasgos) en los siguientes pasos:

- .1. El cliente define el valor de negocio a implementar.
- .2. El programador estima el esfuerzo necesario para su implementación.
- .3. El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.
- .4. El programador construye ese valor de negocio.
- .5. Vuelve al paso 1.

En todas las iteraciones de este ciclo tanto el cliente como el programador aprenden. No se debe presionar al programador a realizar más trabajo que el estimado, ya que se perderá calidad en el software o no se cumplirán los plazos. De la misma forma el cliente tiene la obligación de

manejar el ámbito de entrega del producto, para asegurarse que el sistema tenga el mayor valor de negocio posible con cada iteración.

El ciclo de vida ideal de XP consiste de seis fases: Exploración, Planificación de la Entrega (Release), Iteraciones, Producción, Mantenimiento y Muerte del Proyecto.

✓ **Prácticas XP:** La principal suposición que se realiza en XP es la posibilidad de disminuir la mítica curva exponencial del costo del cambio a lo largo del proyecto, lo suficiente para que el diseño evolutivo funcione. Esto se consigue gracias a las tecnologías disponibles para ayudar en el desarrollo de software y a la aplicación disciplinada de las siguientes prácticas.

- **El juego de la planificación.** Hay una comunicación frecuente el cliente y los programadores. El equipo técnico realiza una estimación del esfuerzo requerido para la implementación de las historias de usuario y los clientes deciden sobre el ámbito y tiempo de las entregas y de cada iteración.
- **Entregas pequeñas.** Producir rápidamente versiones del sistema que sean operativas, aunque no cuenten con toda la funcionalidad del sistema. Esta versión ya constituye un resultado de valor para el negocio. Una entrega no debería tardar más 3 meses.
- **Metáfora.** El sistema es definido mediante una metáfora o un conjunto de metáforas compartidas por el cliente y el equipo de desarrollo. Una metáfora es una historia compartida que describe cómo debería funcionar el sistema (conjunto de nombres que actúen como

vocabulario para hablar sobre el dominio del problema, ayudando a la nomenclatura de clases y métodos del sistema).

- **Diseño simple.** Se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto.
- **Pruebas.** La producción de código está dirigida por las pruebas unitarias. Éstas son establecidas por el cliente antes de escribirse el código y son ejecutadas constantemente ante cada modificación del sistema.
- **Refactorización (Refactoring).** Es una actividad constante de reestructuración del código con el objetivo de remover duplicación de código, mejorar su legibilidad, simplificarlo y hacerlo más flexible para facilitar los posteriores cambios. Se mejora la estructura interna del código sin alterar su comportamiento externo.
- **Programación en parejas.** Toda la producción de código debe realizarse con trabajo en parejas de programadores. Esto conlleva ventajas implícitas (menor tasa de errores, mejor diseño, mayor satisfacción de los programadores).
- **Propiedad colectiva del código.** Cualquier programador puede cambiar cualquier parte del código en cualquier momento.
- **Integración continúa.** Cada pieza de código es integrada en el sistema una vez que esté lista. Así, el sistema puede llegar a ser integrado y construido varias veces en un mismo día.
- **40 horas por semana.** Se debe trabajar un máximo de 40 horas por semana. No se trabajan horas extras en dos semanas seguidas. Si

esto ocurre, probablemente está ocurriendo un problema que debe corregirse. El trabajo extra desmotiva al equipo.

- **Cliente in-situ.** El cliente tiene que estar presente y disponible todo el tiempo para el equipo. Éste es uno de los principales factores de éxito del proyecto XP. El cliente conduce constantemente el trabajo hacia lo que aportará mayor valor de negocio y los programadores pueden resolver de manera inmediata cualquier duda asociada. La comunicación oral es más efectiva que la escrita.
- **Estándares de programación.** XP enfatiza que la comunicación de los programadores es a través del código, con lo cual es indispensable que se sigan ciertos estándares de programación para mantener el código legible. (Abrahamsson, Salo, Ronkainen, & Warsta, 2017)

Agile Unified Process (AUP)

AUP es una metodología de desarrollo ágil heredera de otros paradigmas como la programación extrema (XP) y RUP. Esta metodología consta de principios y prácticas influyentes en la construcción del software en armonía con la documentación esencial de entregables específicos para el entendimiento de la solución. Entre sus objetivos destaca la reducción del costo del cambio en el proyecto en base a procedimientos iterativos (característica propia de RUP) donde la codificación y pruebas del software se llevan a cabo paralelamente (según XP).(Romero Galindo, 2012)

CAPÍTULO III

MATERIALES Y MÉTODOS

3.1. UBICACIÓN DE LA POBLACIÓN

La Unidad de Gestión Educativa Local Canas queda en la provincia de Canas es una de las trece provincias que conforman el departamento del Cusco, bajo la administración del Gobierno regional de Cusco, en la zona de los Andes de Perú. Limita al norte con la provincia de Acomayo, al este con la provincia de Canchis y el departamento de Puno, al sur con la provincia de Espinar y al oeste con la provincia de Chumbivilcas.

3.2. POBLACIÓN

La población del presente trabajo está conformada por el personal administrativo de todas las Unidades de Gestión Educativas Locales, UGELes de la región de Cusco

3.3. MUESTRA

La muestra de datos para el presente trabajo de investigación está conformado por el personal administrativo de la Unidad de Gestión Educativa Local UGEL Canas, Cusco.

3.4. MÉTODO DE RECOLECCIÓN DE DATOS

La recolección de datos para el presente trabajo se realizó a través de la observación y la entrevista con los expertos en el área de administración de contenidos de la Unidad de Gestión Educativa Canas Cusco – UGEL Canas.

3.5. DISEÑO ESTADÍSTICO A UTILIZAR

El presente trabajo se desarrolló un análisis comparativo y estadístico del desempeño de la administración de la documentación, antes y después, en la UGEL Canas.

3.6. METODOLOGÍA DE DESARROLLO

Extreme Programming (XP) surge como una nueva manera de encarar proyectos de software, proponiendo una metodología basada esencialmente en la simplicidad y agilidad. Las metodologías de desarrollo de software tradicionales (ciclo de vida en cascada, evolutivo, en espiral, iterativo, etc.) aparecen, comparados con los nuevos métodos propuestos en XP, como pesados y poco eficientes. La crítica más frecuente a estas metodologías “clásicas” es que son demasiado burocráticas. Hay tanto que hacer para seguir la metodología que, a veces, el ritmo entero del desarrollo se retarda. Como respuesta a esto, se ha visto en los últimos tiempos el surgimiento de “Metodologías Ágiles”. Estos nuevos métodos buscan un punto medio entre la ausencia de procesos y el abuso de los mismos, proponiendo un proceso cuyo esfuerzo valga la pena.

La metodología XP define cuatro variables para cualquier proyecto de software: costo, tiempo, calidad y alcance. Además, se especifica que, de estas cuatro variables, sólo tres de ellas podrán ser fijadas arbitrariamente por actores externos al grupo de desarrolladores (clientes y jefes de proyecto). El valor de la variable restante podrá ser establecido por el equipo de desarrollo, en función de los valores de las otras tres. Este mecanismo indica que, por ejemplo, si el cliente establece el alcance y la calidad, y el jefe de proyecto el

precio, el grupo de desarrollo tendrá libertad para determinar el tiempo que durará el proyecto. Este modelo es analizado por Kent Bec (Joskowicz, 2008)

Para el desarrollo del Software de compresión se aplicó la metodología de la Programación Extrema (Extremme Programming - XP) que se adapta hoy en día perfectamente al desarrollo del ciclo de vida del Software y para el modelado del Software se usó el Lenguaje Unificado de Modelamiento (Unified Modeling Lenguaje - UML).

FASES DE LA METODOLOGÍA XP

Las fases de la programación extrema se dividen en cuatro fases:

- ✓ **Análisis:** La metodología XP plantea en análisis como un permanente diálogo entre la parte empresarial y técnica del proyecto, en la que los primeros decidirán el alcance ¿Qué es lo realmente necesario del proyecto?, la prioridad qué debe ser hecho en primer lugar, la composición de las versiones que debería incluir cada una de ellas y la flecha de las mismas en cuanto a los técnicos, son los responsables de estimar la duración requerida para implementar las funcionalidades deseadas por el cliente, de informar sobre las consecuencias de determinadas decisiones, de organizar la cultura de trabajo y finalmente de realizar la planificación detallada dentro de cada versión. XP no solo es un método centrado en el código que lo es, sino que sobre todo es un método de gestión de proyectos software (Booch, 2010).

- ✓ **Diseño:** El Propósito del diseño es de crear una arquitectura para la naciente implementación, el diseño arquitectural sólo puede comenzar una vez que el equipo tenga un entendimiento razonable de los requerimientos del sistema. El diseño, como el análisis, nunca termina realmente hasta que el sistema final es entregado. Durante esta fase se alcanza un cierto diseño y al establecer políticas para diversos problemas tácticos.

El diseño se enfoca en la estructura, estática y dinámica, su propósito principal es de crear el esqueleto concreto del sistema sobre el cual todo el resto de la implementación se basa (Booch, 2010)

- ✓ **Desarrollo:** Esta etapa debe reunir las siguientes características o cualidades:
 - El software está siempre disponible.
 - Se debe escribir código de acuerdo a los estándares.
 - Desarrollar la unidad de pruebas primero.
 - Todo el código debe programarse por parejas.
 - Integrar frecuentemente.
 - Todo el código es común a todos.
- ✓ **Prueba:** Todo el código debe ir acompañando, Los casos de prueba se escriben antes que el código. Los desarrolladores escriben pruebas unitarias y los clientes especifican pruebas funcionales.

- ✓ **Evaluación.** Esta etapa es realizada aplicando la norma ISO/IEC 9126 donde detalla el modelo a usar para la calidad del producto software, tanto la calidad interna como la calidad externa. El estándar ISO/IEC 9126 puede ser usado desde varias perspectivas, en esta investigación es usado de la perspectiva de desarrollo del sistema de administración.

Para el caso de la utilización del ISO – 9126 está desarrollado en el anexo N° B donde se llenó la ficha de evaluación y de esa forma se tomó la decisión que el sistema es factible.

CAPÍTULO IV

RESULTADOS Y DISCUSIÓN

Después de describir los resultados antes de implantar el sistema denominado implementación de sistema de administración de contenidos web usando tecnologías internet Information Server - IIS y Asp.net para la Unidad de Gestión Educativa Canas Cusco – UGEL Canas 313, se discutió los resultados en función a los objetivos planteados.

4.1. ANÁLISIS DEL SISTEMA

Para el desarrollo del sistema de administración de contenidos web, el primer paso fue analizar los datos e información realizados en estadística e informática de la UGEL Canas, donde nos topamos que sus datos los guardaban en hojas de cálculo Excel. En varios de los casos la utilización de estos archivos trabajados en diseño y/o ofimática ha causado más demoras y pérdida de información y comunicación actualizada es así como se empezó con el diseño del sistema.

4.1.1. Análisis de requisitos del sistema

En esta parte se tomó en cuenta las historias de los usuarios que en este caso serían el administrador y todos los trabajadores quienes describieron de cómo les gustaría que sea el software que se va a construir. Para poder recolectar estas historias de usuario se han demoran 2 semanas.

DEFINICIÓN DE ROLES

Dada la coyuntura de la investigación, y la disponibilidad de recursos humanos, el investigador ha asumido los roles de directa relación con el desarrollo del sistema. Solo se han tomado en consideración los roles más importantes según el desarrollo de la presente.

- **Programador:** El investigador asume el rol de programador por tal motivo es el encargado de escribir el código fuente necesario para la implementación del sistema de administración de contenidos web.
- **Cliente:** El Director de la Ugel cumple el rol de cliente, define las especificaciones del sistema e influye en el desarrollo sin ejercer control, define las pruebas funcionales.
- **Tester:** Este rol es también asumido por el desarrollador con el fin de apoyar al Director en la preparación y realización de pruebas también está encargado de explicar los resultados al equipo.
- **Tracker:** El investigador analiza la información sobre la marcha del proyecto sin afectar demasiado el proceso.
- **Entrenador:** El investigador, es el responsable global del proyecto también es el encargado de verificar que se estén aplicando correctamente las guías XP.

4.1.2. Funcionalidades, requisitos de tipo de usuario

En el sistema de administración se tiene un solo administrador y un usuario que le apoya en la actualización del sistema

Administrador: Toda persona con una cuenta y accesos autorizados al sistema realiza funciones tales como: registro de nuevos usuarios, monitoreo del funcionamiento del sistema y notificación de los posibles errores a presentarse.

4.1.3. Casos de uso para el administrador y/o usuario

La propuesta de una interfaz está dirigida a desarrollar un sistema que facilite la comunicación e información vía internet a la población.

Utilizando las tecnologías y modelamientos según las faces a continuación se muestra un diagrama de casos de uso para mostrar el modelado del sistema.

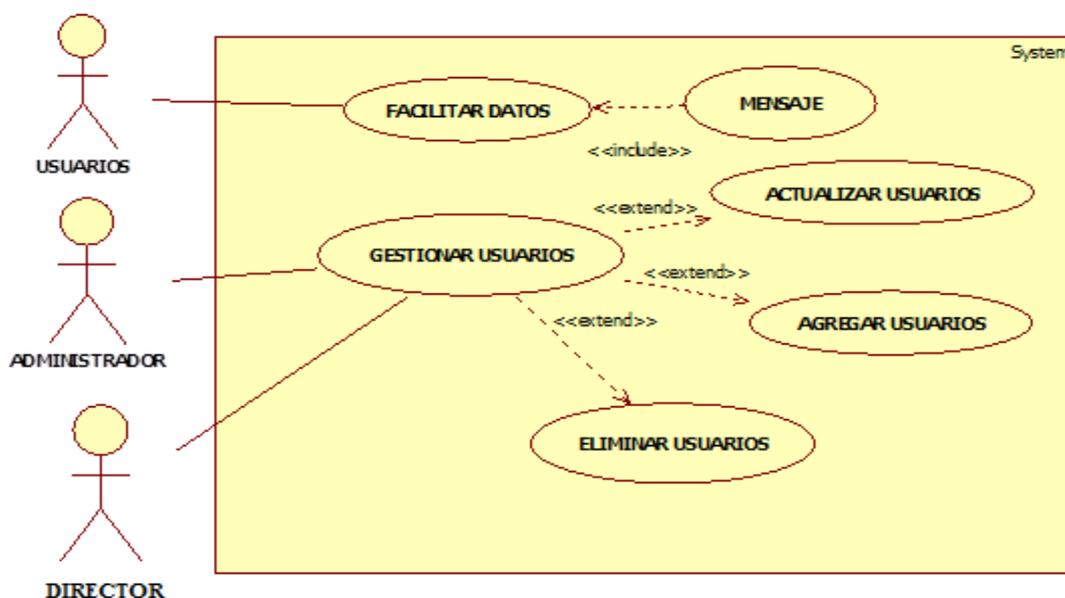


Figura 3. Diagrama de Casos de Uso del Administrador de usuario

Fuente: elaboración propia

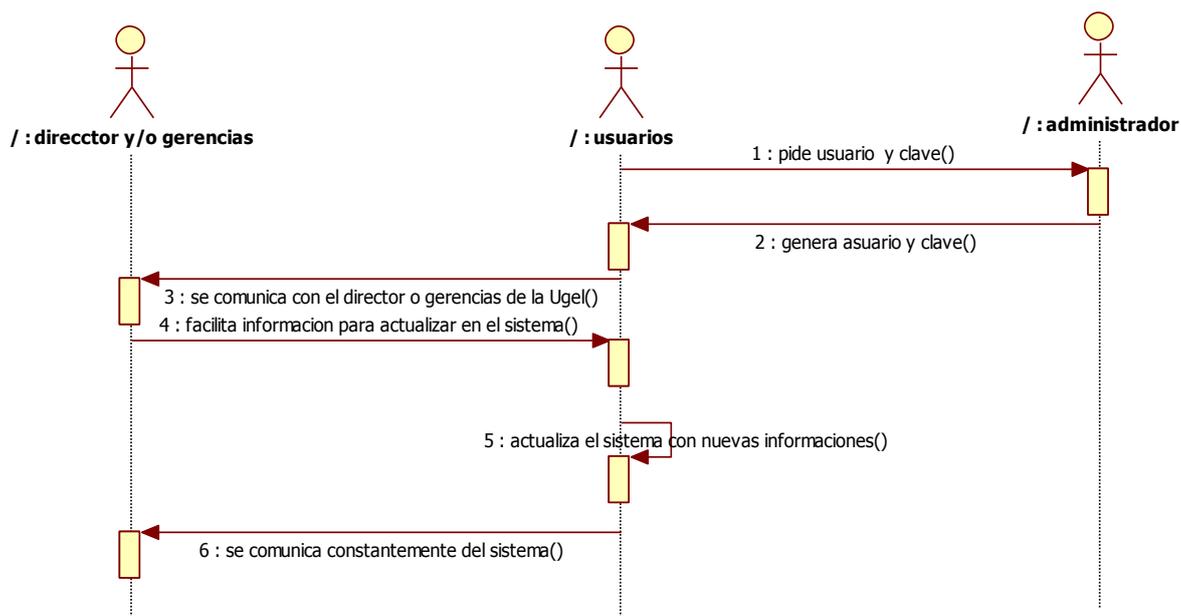


Figura 4. Diagrama de Secuencia del Administrador de Usuario

Fuente: Elaboración Propia

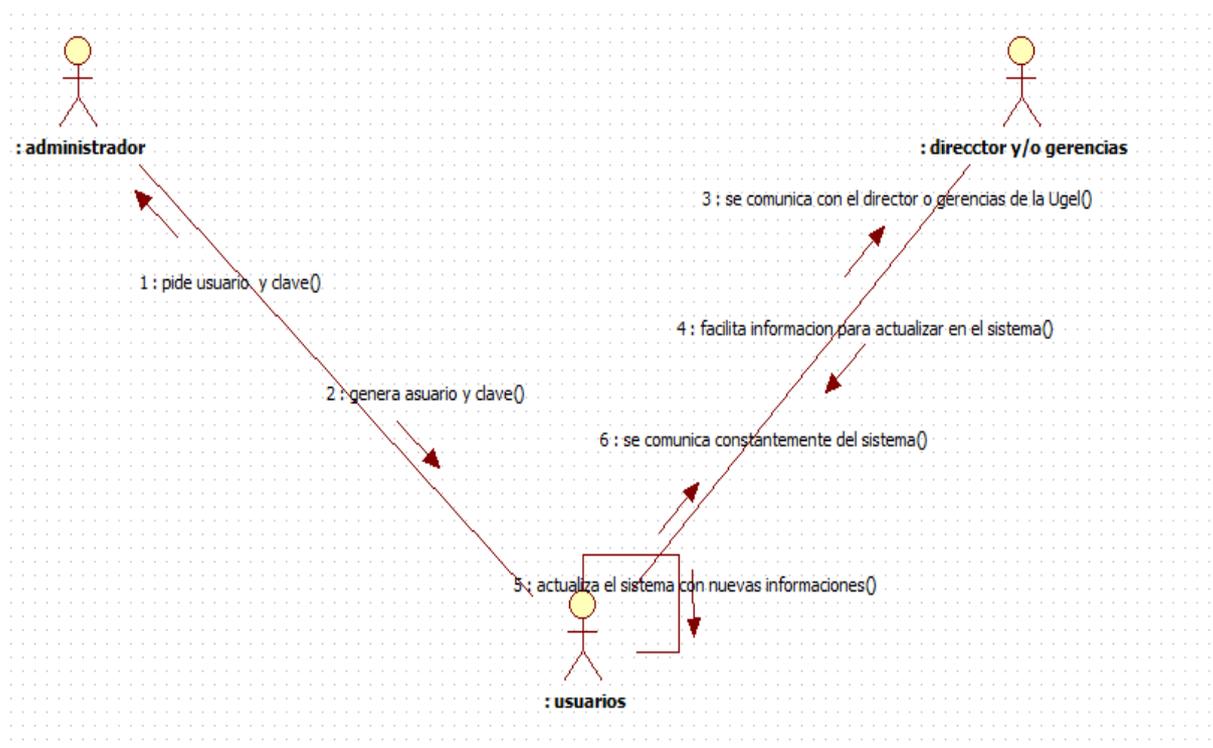


Figura 5. Diagrama de colaboración del administrador de usuario

Fuente: Elaboración Propia

4.1.4. Funcionalidades, del estadístico y director

En el sistema de administración se tiene un solo administrador y un usuario o estadístico y la dirección que le apoya en la actualización del sistema

Estadístico: Toda persona con una cuenta y accesos autorizados al sistema realiza funciones tales como: agregar, modificar y eliminar comunicados, noticias y otros según la autorización del Director de la Ugel Canas

4.1.5. Casos de uso para el usuario y/o estadístico

Utilizando las tecnologías y modelamientos según las faces a continuación se muestra un diagrama de casos de uso para mostrar el modelado del sistema.

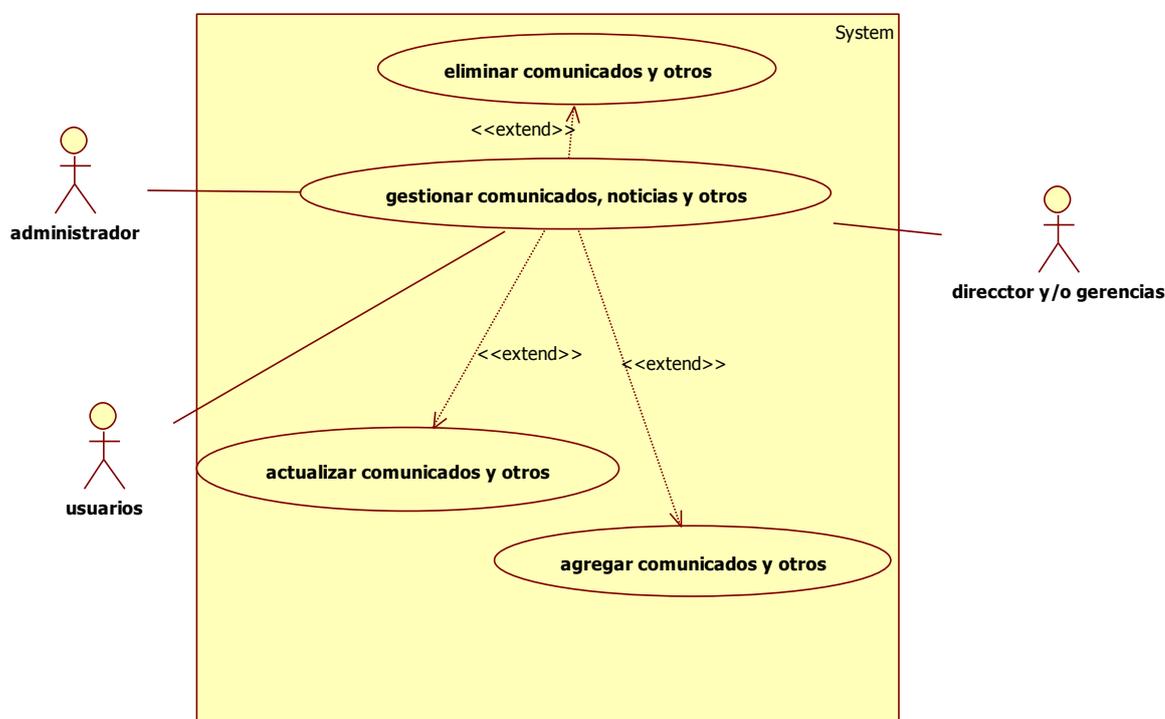


Figura 6. Diagrama de Casos del Sistema de Administración

Fuente: Elaboración Propia

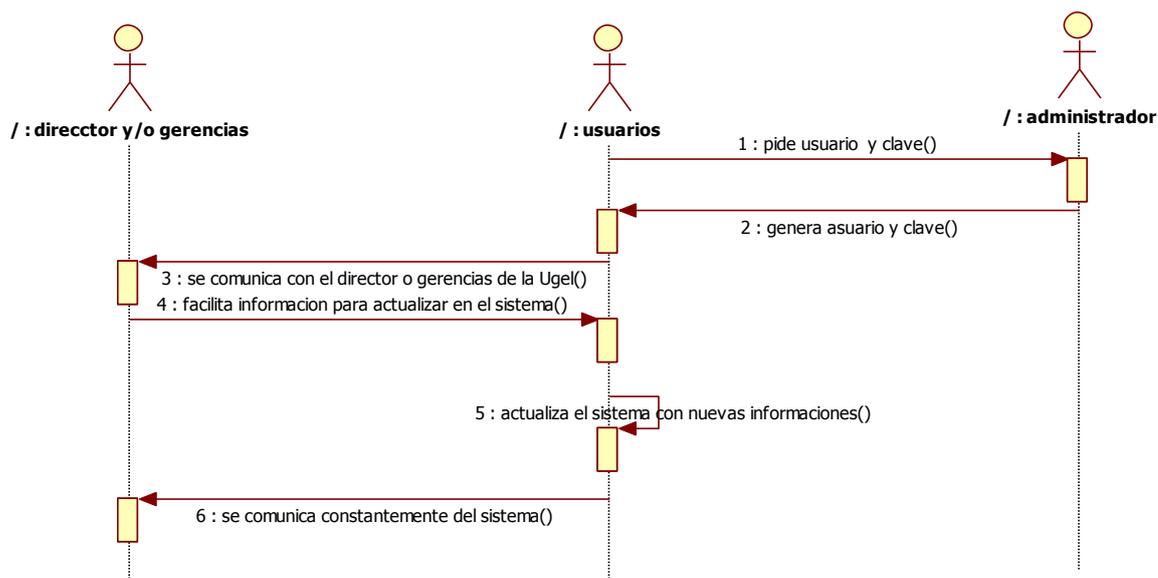


Figura 7. Diagrama de Secuencia del Sistema de Administración

Fuente: Elaboración Propia

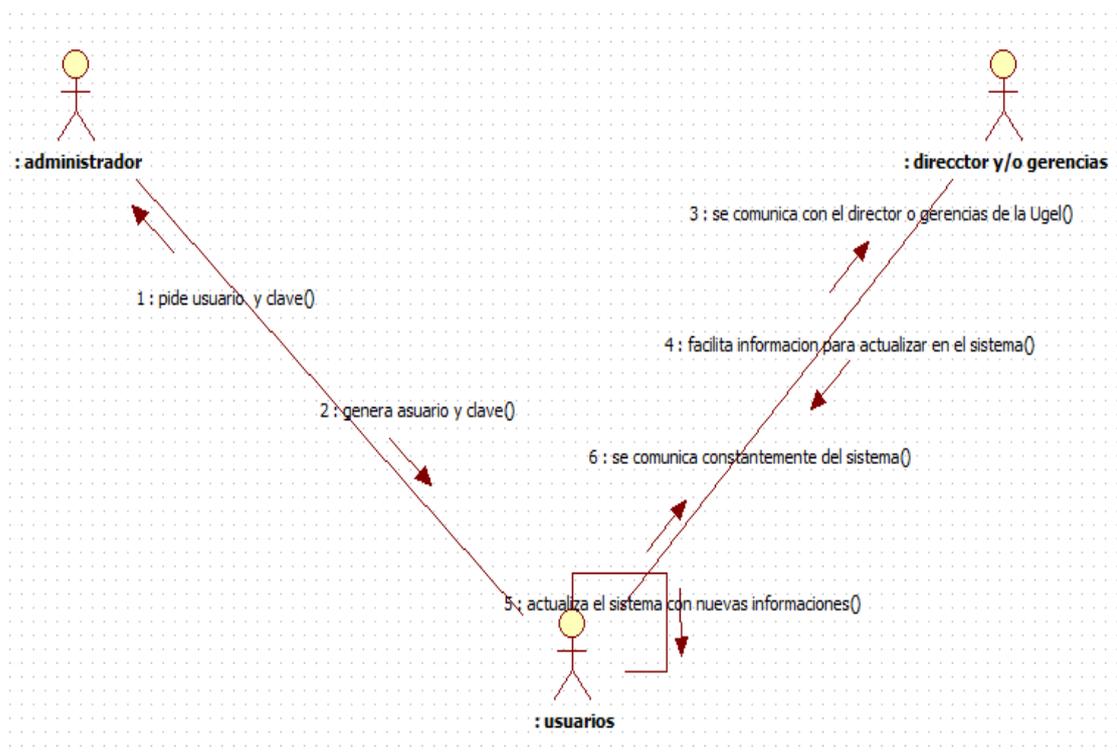


Figura 8. Diagrama de Colaboración del Sistema de Administración

Fuente: Elaboración Propia

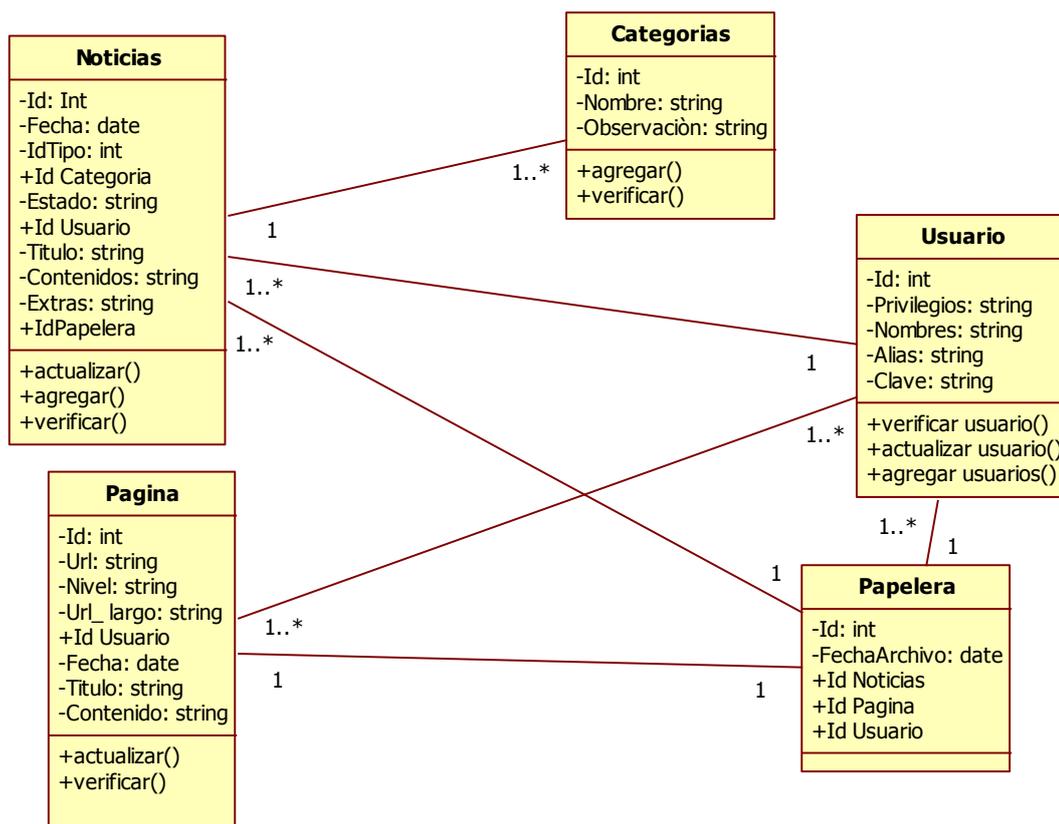


Figura 9. Diagrama de Clases del Sistema de Administración de Contenidos Web

Fuente: Elaboración Propia

4.2. DISEÑO

Se tiene las siguientes arquitecturas para la naciente implementación, el diseño arquitectural se comenzó una vez que se tuvo un entendimiento razonable de los requerimientos del sistema.

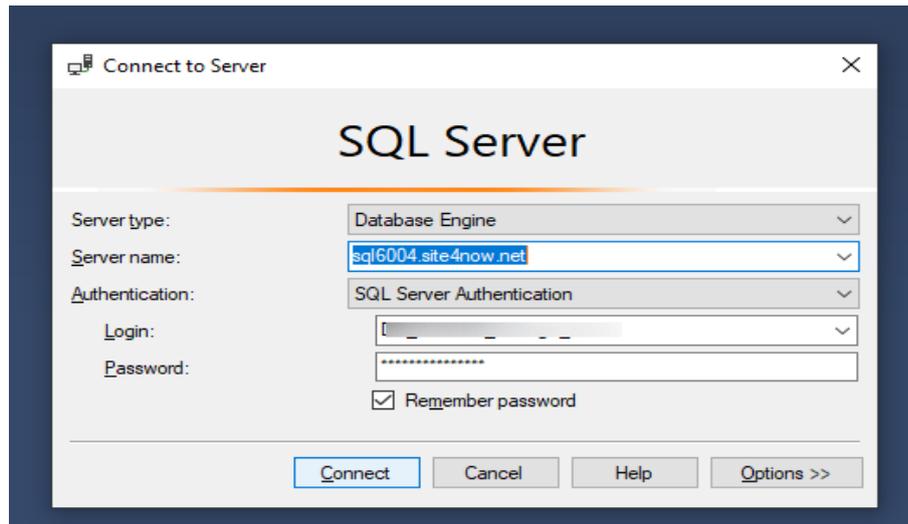


Figura 10. Acceso a la Base de Datos del Sistema de Administración de Contenidos Web

Fuente: Elaboración Propia.

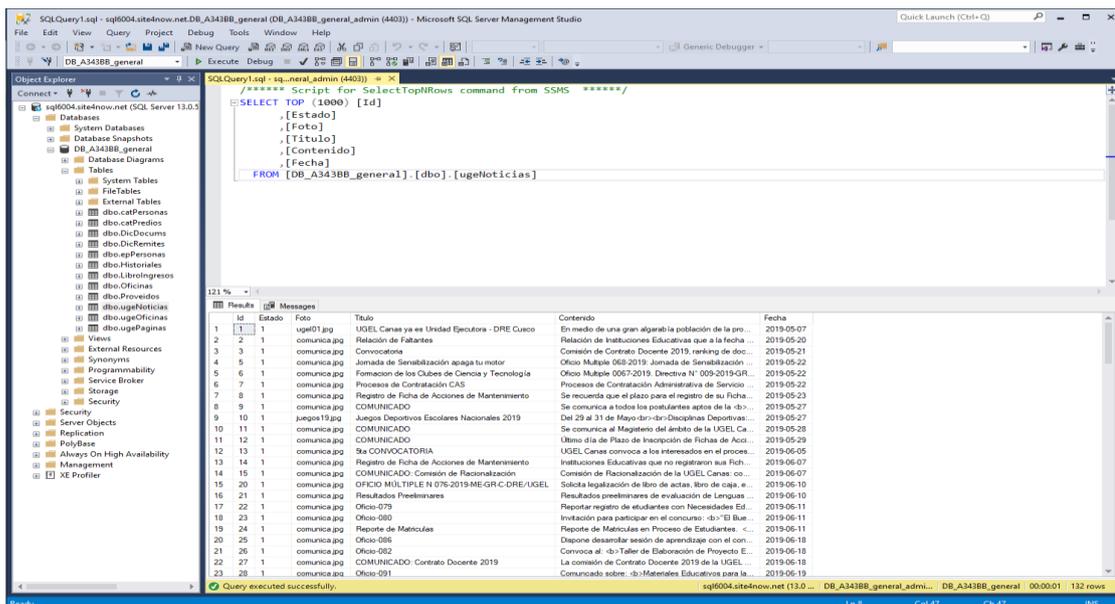


Figura 11. Tablas con que se Trabajó la Base de Datos Base de Datos, del Sistema de Administración de Contenidos Web

Fuente: Elaboración Propia

Del cual el diseño como el análisis, nunca termina realmente hasta que el sistema final es entregado.



Figura 12. Interfaz del Sistema de Administración de Contenidos Web. ugelcanas.gob.pe

Fuente: Elaboración Propia

4.3. DESARROLLO



Figura 13. Pantalla Principal del Sistema de Administración de Contenidos web Canas – Cusco 2019

Fuente: Elaboración Propia

En la figura N° 13, pantalla principal del sistema de administración de contenidos web, que tiene 7 menús, inicio, institucional, estructura orgánica, convocatorias, normatividades, noticias y admin.



Figura 14. Noticias y Enlaces Importantes vinculados a Educación en la Ugel Canas – Cusco 2019

Fuente: Elaboración Propia

4.4. PRUEBA

A. Pruebas no convencionales:

Son pruebas que consisten en las revisiones técnicas formales que se realizaron en las etapas de análisis y diseño del sistema, que corrigen errores básicamente de:

- Omisiones y ambigüedad en las definiciones de clases y jerarquías, así como en las relaciones.
- Inconsistencias en la elaboración de Diagrama de Casos de Uso, Interacción, Clases y Actividades.

B. Pruebas convencionales:

Son pruebas que se pueden ejecutarse o probarse, los cuales se realizan en la etapa de la implementación del sistema como son las pruebas de caja negra y caja blanca.

C. Prueba de caja blanca:

Esta prueba de software, fue desarrollada durante la construcción de cada módulo. Mediante esta prueba se garantiza que el prototipo del sistema de control, cumplió con:

- Ejecutar todos los caminos independientes de cada módulo.
- La estructura de los datos es compatible.
- Ninguno de los bucles es infinito o su ejecución es por demás.
- Cuenta con todas las decisiones lógicas necesarias.
- Dar mayor prioridad a las consultas de búsqueda

D. Prueba de caja negra:

Esta prueba de software se aplicó en el desarrollo de los módulos así como también estuvieron terminados y enlazados entre ellos para su funcionamiento como sistema. Mediante esta prueba aseguramos que el sistema de administración de contenidos web, no tiene errores de:

- Procedimientos o funciones incorrectas.
- El pie de reporte muestran información de continuación.
- Errores de entrada y salida.
- Errores de rendimiento.
- Errores de inicialización y finalización.
- Los resultados de la consulta muestra lo requerido

E. Prueba del sistema:

Esta fase de prueba del sistema se realiza mediante el método Prueba Basado en Escenarios, con el fin de descubrir errores de interfaz y errores del procesamiento de datos al nivel de los resultados esperados. La prueba se concentra en lo que el usuario hace interacción con el sistema. La validación de la funcionalidad integrada del sistema se prueba con los datos de la población de trabajadores de la empresa, esto permite verificar la certidumbre de los resultados proporcionados.

Por lo tanto concluimos que el sistema de administración de contenidos web mejoró notablemente para los encargados de estadística e informática en beneficio de la Institución.

4.5. PROCESAMIENTO DE DATOS SEGÚN LAS ENCUESTAS

Tabla 2. Distribución del Personal según la Calificación del Interfaz Ergonómico y Adecuado, en la Ugel Canas 2019

Indicadores	Frecuencia	Porcentaje
SI	12	80
NO	1	6
REGULAR	2	13
Total	15	100

Fuente: Entrevista Directa

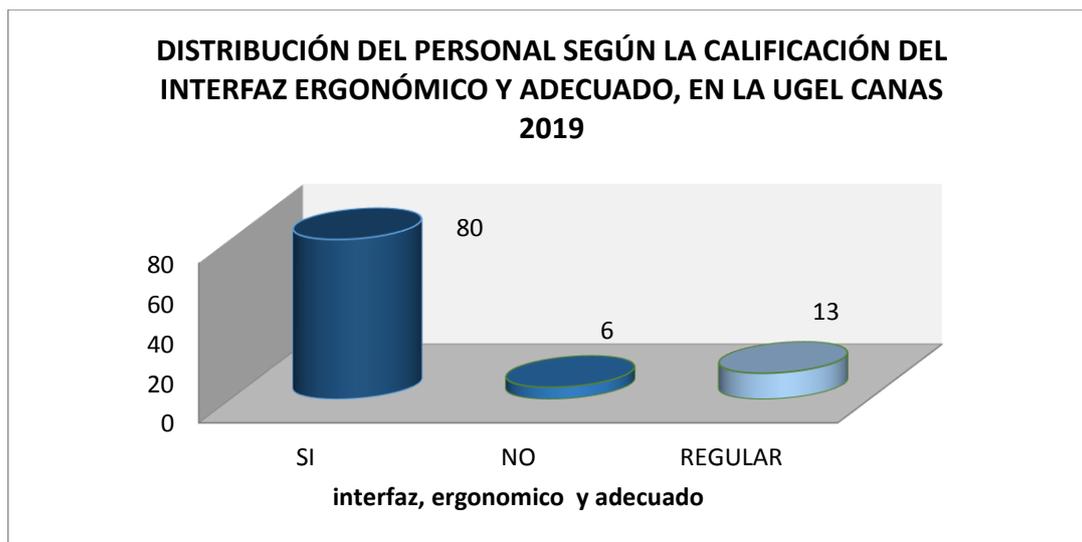


Figura 15. Distribución del Personal Según la Calificación del Interfaz Ergonómico y Adecuado, en la Ugel Canas 2019

Fuente: Entrevista Directa

Interpretación.- según la tabla 2 y figura 15, podemos ver que el 80% del personal de la UGEL, indican que el interfaz es ergonómico y adecuado, el 13% del personal dijeron es regular la adecuación del interfaz.

Tabla 3. Distribución del Personal Según si Ayudara en Informarse de la Institución, en la Ugel Canas 2019

Indicadores	Frecuencia	Porcentaje
SI	13	86
NO	1	6
AVECES	1	6
Total	15	100

Fuente: Entrevista Directa

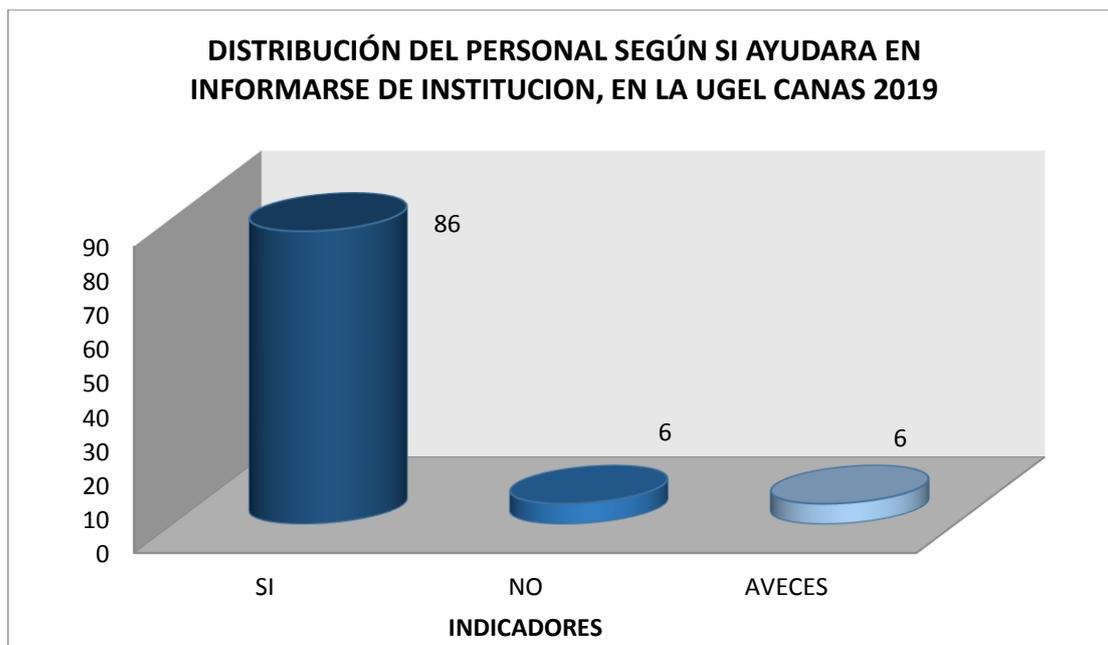


Figura 16. Distribución del Personal Según si Ayudara en Informarse de la Institución, en la Ugel Canas 2019

Fuente: Entrevista Directa

Interpretación.- según la tabla N° 3 y figura N° 16, podemos ver que el 86% del personal de la UGEL, indican que ayudara a informarse de la Institución y el 6% del personal dijeron que no y con el mismo porcentaje indicaron a veces ayudara a informarse de los comunicados y otros informaciones.

Tabla 4. Distribución del Personal Según como Obtenían Información antes, en la Ugel Canas 2019

Indicadores	Frecuencia	Porcentaje
Tedioso tenías que ir hasta el lugar de la UGEL	6	40
No estaba informado nada de la UGEL	9	60
Total	15	100

Fuente: Entrevista Directa

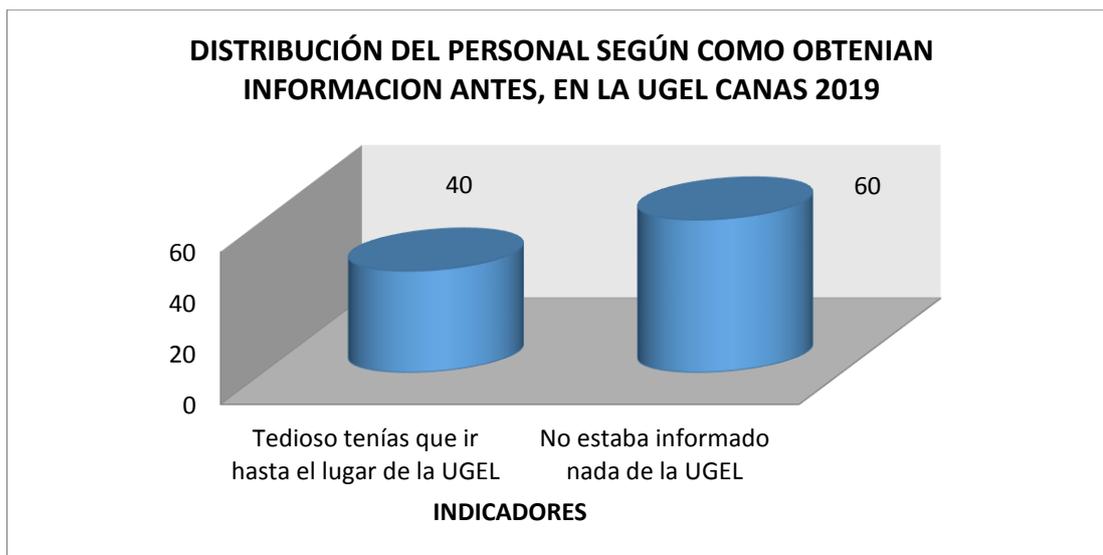


Figura 17. Distribución del Personal Según como Obtenían Información antes, en la Ugel Canas 2019

Fuente: Entrevista Directa

Interpretación.- según la tabla N° 4 y figura N° 17, podemos ver que el 60% del personal de la UGEL, indican que no estaba informado nada de la UGEL, el 40% del personal dijeron que tenían que ir hasta la misma Institución para averiguar alguna información.

Tabla 5. Distribución del Personal según como Obtienen Información ahora, en la Ugel Canas 2019

Indicadores	Frecuencia	Porcentaje
Tedioso tenías que ir hasta el lugar de la UGEL	3	20
Es rápido con respecto a adquirir la información	12	80
Total	15	100

Fuente: Entrevista Directa

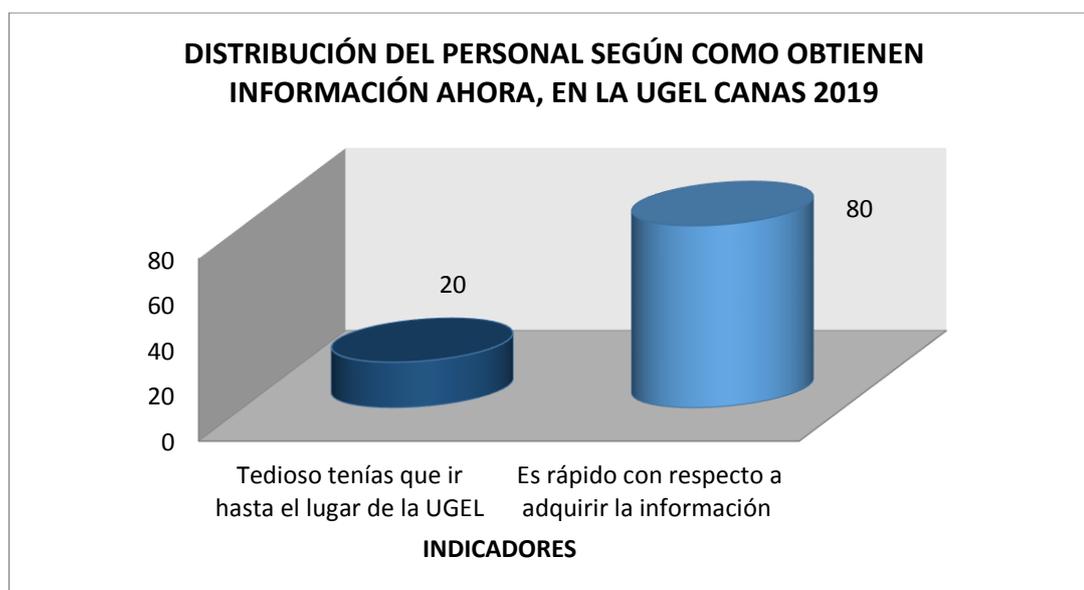


Figura 18. Distribución del Personal según como Obtienen Información ahora, en la Ugel Canas 2019

Fuente: Entrevista Directa

Interpretación.- según la tabla N° 5 y figura N° 18, podemos ver que el 80% del personal de la UGEL, indican que si es rápido con respecto a adquirir información y el 20% del personal dijeron que tenían que ir hasta la misma Institución para averiguar alguna información (Tedioso).

4.6. MÉTRICAS DE LA FUNCIÓN

Valores del dominio de información de administración de contenidos web usando tecnologías internet information server - IIS y asp.net para la unidad de gestión educativa canas cusco – Ugel Canas 313.

Tabla 6. Parámetros de medición

Parámetro De Medición	Cuenta	Factor De Ponderación			Cuenta PF
		Simple	Media	Compleja	
Nº de entradas de usuario	3	3	4	6	9
Nº de salidas de usuario	8	4	5	7	32
Nº de peticiones de usuario	5	3	4	6	15
Nº de archivos	4	7	10	15	28
Nº de interfaces externas	1	5	7	10	5
Cuenta total					89

Fuente: Elaboración Propia

Para calcular los puntos de función se utilizó la siguiente relación.

$$PF = CUENTA_TOTAL * [0.65 + 0.01 * SUM(fi)]$$

Tabla 7. Valores de ajuste de la complejidad del sistema

Fi	FACTORES DE PONDERACION	VALOR
1	¿Requiere el sistema copias de seguridad y de recuperación fiables?	1
2	¿Se requiere comunicación de datos?	5
3	¿Existen funciones de procesamiento distribuido?	0
4	¿Es crítico el rendimiento?	0
5	¿Se ejecutara el sistema en un entorno operativo existente y fuertemente utilizado?	5
6	¿Requiere el sistema entrada de datos interactiva?	3
7	¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas u operaciones?	2
8	¿Se actualizan los archivos maestros de forma interactiva?	4
9	¿Son complejos las entradas, salidas, los archivos o las peticiones?	1
10	¿Es complejo el procesamiento interno?	3
11	¿Se ha diseñado el código para ser reutilizable?	4
12	¿Están incluidas en el diseño la conversión y la instalación?	4
13	¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	4
14	¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?	4
TOTAL		40

Fuente: elaboración propia

Por consiguiente:

$$PF = 89 * [0.65 + 0.01*40]$$

$$PF = 93.45$$

- El tiempo estimado que se tardó en la implementación del sistema es de 3 meses entonces:

$$\text{Productividad media} = PF / \text{mes-persona} = 93.45/3 = 31.15$$

- Según la tarifa laboral estimada en persona/mes, el coste por PF es aproximadamente:

Tarifa laboral= sueldo mínimo / productividad media

$$= 950/31.15 = 16.05$$

El coste estimado del sistema es de 2745 soles es el esfuerzo de una persona.

Tabla 8. Escala Valorativa de la Calificación del Sistema de Administración Mediante ISO 9126

CLASIFICACIÓN	INTERVALO	DECISIÓN
A)INACEPTABLE	[27-54>	
B)MÍNIMAMENTE ACEPTABLE	[54-81>	
C)ACEPTABLE	[81-95>	
D)CUMPLE LOS REQUISITOS	[95-122>	99,entonces cumple los requisitos
E)EXCEDE LOS REQUISITOS	[122-135>	

Fuente: Elaboración Propia

CONCLUSIONES

- PRIMERA.** La información recabada durante el desarrollo del sistema de administración contribuyo a una implementación adecuada, satisfaciendo a todos y a cada uno de los requerimientos del trabajo de investigación para la comunicación Web con los profesores de la UGEL Canas.
- SEGUNDA.** Para el diseño del modelo se hizo un análisis orientado a objetos para el modelado de base de datos y el diseño de interfaz principal y otros para la administración de contenidos web la Unidad de Gestión Educativa Canas Cusco – UGEL Canas.
- TERCERA.** Se desarrolló de acuerdo a la metodología XP (programación extrema), donde todos los artefactos son objeto de modificaciones a lo largo del proceso de desarrollo, lo cual, fue eje fundamental para que al término del proceso se pueda tener una versión definitiva y completa, para mejorar la administración de contenidos web la Unidad de Gestión Educativa Canas Cusco – UGEL Canas.
- CUARTA.** La evaluación del sistema usando tecnologías internet Information Server - IIS y Asp.net mejoro la administración de contenidos web, se utilizó una encuesta al personal administrativo de la Unidad de gestión Educativa Local Canas.

RECOMENDACIONES

- PRIMERA.** Para posteriores investigaciones similares, se recomienda, involucrar en forma completa todo lo que respecta a la parte administrativa, para tener una visión y un control completo de la unidad de Gestión Educativa Local.
- SEGUNDA.** Que en la próxima implementación de un sistema de contenidos web, se pueda realizar trámites documentarios que faciliten a los docentes de educación básica y a la población en general.
- TERCERA.** Se recomienda que se mantenga el uso de la metodología XP, en caso individual y/o SCRUM si uno trabaja en equipo, como proceso de desarrollo de software, dado que garantiza un producto final óptimo, por ser iterativo y ágil, asegurar la producción de software de calidad dentro de plazos predecibles.
- CUARTA.** Que en las siguientes investigaciones de sistemas de administración puedan realizar video conferencias, capacitación y otros en tiempo real.

REFERENCIAS BOBLOGRÁFICAS

- Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2017). Agile software development methods: Review and analysis. *arXiv preprint arXiv:1709.08439*.
- Adell, A. S. (1995). *La navegación hipertextual en el World-Wide Web: implicaciones para el diseño de materiales educativos*: L'autor.
- Aguilar González, I. E., Guzmán Vásquez, S. L., Mejía López, J. A., Rivera Machado, M. R., & Ventura Arenivar, J. A. (2018). *Sistema Informático de Gestión Administrativa de la Secretaría de Cultura de San Salvador*. Universidad de El Salvador.
- Alarcón, V. F. (2006). *Desarrollo de sistemas de información: una metodología basada en el modelado* (Vol. 120): Univ. Politèc. de Catalunya.
- Apaza Quispe, G. (2014). Sistema informático de gestión administrativa para la Coordinación de investigación de la Facultad de Ingeniería estadística e Informática de la UNA Puno año 2014.
- Brambila, D. A. C., & Obando, A. L. (2015). Sistema de evaluaciones en línea como herramienta para los niveles de educación media superior. *RIDE Revista Iberoamericana para la Investigación y el Desarrollo Educativo*, 6(11).
- Canós, J. H., & Letelier, M. C. P. P. (2012). Metodologías ágiles en el desarrollo de software.
- Figueroa, R. G., Solís, C. J., & Cabrera, A. A. (2008). Metodologías tradicionales vs. metodologías ágiles. *Universidad Técnica Particular de Loja, Escuela de Ciencias de la Computación*.

- García, J., & Abraham, F. (2012). *Desarrollo y aplicación de un sistema de soporte a la decisión diagnóstico cognitivo en web para la gestión de contenidos en un sistema de educación virtual*.
- Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, P. (2010). *Metodología de la investigación (Vol. 3)*: México: McGraw-Hill.
- Joskowicz, J. (2008). Reglas y prácticas en eXtreme Programming. *Universidad de Vigo*, 22.
- Pástor Ramírez, D. M. *Modelo para la generación de cursos virtuales usando tecnologías de la web semántica para sistemas de gestión de aprendizaje*. Universidad Nacional de Colombia-Sede Medellín.
- Poppendieck, M., & Poppendieck, T. (2003). *Lean Software Development: An Agile Toolkit: An Agile Toolkit*. Addison-Wesley.
- Reyna, N. A. C. (2017). Implementación de la Nube Computacional para la Gestión de Proyectos En La Empresa B&B Murillo Sac–Arequipa; 2017. In *Crescendo Ingeniería*, 4(2), 159-166.
- Ribes, X. (2007). La Web 2.0. El valor de los metadatos y de la inteligencia colectiva. *Telos*, 73, 36-43.
- Roger, S. P. (2002). *Ingeniería de Software: Un enfoque práctico*: McGraw Hill.
- Romero Galindo, R. M. (2012). Análisis, diseño e implementación de un sistema de información aplicado a la gestión educativa en centros de educación especial.
- Salazar, O. A., Aguirre, F. A. M., & Osorio, J. A. C. (2011). Herramientas para el desarrollo rápido de aplicaciones web. *Scientia et technica*, 1(47), 254-258.

Torres Cruz, F. (2016). Plataforma web basada en cloud computing para el seguimiento de proyectos de tesis de pregrado UNA Puno 2016.

Valderrama Guayan, F. E., & Benites Barrientos, R. (2014). Desarrollo de un sistema informático web para la gestión de producción de calzados de la empresa jaguar SAC Utilizando la metodología AUP y tecnología asp. net framework mvc3.

Van Der Henst, C. (2005). ¿ Qué es la Web 2.0. *Recuperado el, 15.*

ANEXOS

**ANEXO A: ENCUESTA AL PERSONAL**

UNIVERSIDAD NACIONAL DEL ALTIPLANO-PUNO



ESCUELA PROFESIONAL DE INGENIERIA ESTADISTICA E INFORMATICA

ENCUESTA

IMPLEMENTACIÓN DE SISTEMA DE ADMINISTRACIÓN DE
CONTENIDOS WEB USANDO TECNOLOGIAS INTERNET INFORMATION
SERVER - IIS Y ASP.NET PARA LA UNIDAD DE GESTION EDUCATIVA
CANAS CUSCO – UGEL CANAS 313

Por favor responda con sinceridad las preguntas:

1. ¿Cree Ud. Que el sistema de administración de contenidos web de la Institución su interfaz es ergonómico y adecuado?
 - a) Si
 - b) No
 - c) regular

2. Con la implementación del sistema ¿cree Ud. que le ayudara en informarse de comunicados, noticias, convocatorias y otros?
 - a) Si
 - b) No
 - c) a veces

3. Antes de tener el sistema de administración, ¿cómo eran para adquirir información de la UGEL Canas?
 - a) Tedioso tenías que ir hasta el lugar de la UGEL
 - b) No estaba informado nada de la UGEL
 - c) Es rápido con respecto a adquirir la información

4. Ahora al tener el sistema de administración, ¿cómo eran para adquirir información de la UGEL Canas?
- a) Tedioso tenías que ir hasta el lugar de la UGEL
 - b) No estaba informado nada de la UGEL
 - c) Es rápido con respecto a adquirir la información

Gracias por su colaboración

ANEXO B: FICHA DE EVALUACION DEL ISO 9126

Ficha de Evaluación de la calidad del Producto Estándar ISO – 9126

INDICADORES	PUNTUACIÓN				
	1	2	3	4	5
1.funcionalidad					
Adecuación: la capacidad del producto software para proporcionar un conjunto apropiado de funciones para tareas específicas y objetivos de los usuarios.					X
Exactitud: la capacidad del producto de software para proporcionar los resultados o efectos correctos y con el grado de precisión acordado.			X		
Interoperabilidad: la capacidad del producto de software para interactuar con uno o más sistemas específicos.		X			
Seguridad: referido a la capacidad del producto software para proteger la información y los datos.			X		
Conformidad: la capacidad del producto software para adaptarse a los estándares, convenciones o regulaciones en leyes y prescripciones relativos a la funcionalidad.					X
2.FIABILIDAD					
Madurez: la capacidad del producto software para evitar fallos provocados por errores en el software.					X
Tolerancia o Fallos: la capacidad del producto					X

software para mantener un nivel de rendimiento determinado en caso de defectos en el software o incumplimiento en su interfaz.

Recuperabilidad: la capacidad del producto software para restablecer un determinado nivel de rendimiento y recuperar los datos afectados directamente en caso de ocurrir un fallo.

X

Conformidad: la capacidad del producto software para adaptarse a los estándares, convenciones o regulaciones referidas a la fiabilidad.

X

3.USUABILIDAD

Comprensibilidad: la capacidad del producto software para permitir al usuario que entienda si el software es adecuado, y como debe utilizarse para determinadas tareas y bajo ciertas condiciones de uso.

X

Facilidad de Aprendizaje: la capacidad del producto software para permitir al usuario aprender su aplicación.

X

Atracción: la capacidad del producto software para atraer al usuario.

X

Conformidad: la capacidad del producto software para adaptarse a estándares, convenciones, guías de estilo y regulaciones relacionadas con la usabilidad.

X

Operabilidad: la capacidad del producto software para permitir que el usuario lo opere y lo controle. X

4.EFICACIA

Comportamiento Temporal: la capacidad del producto software para proporcionar tiempos de respuesta y procesamiento apropiados cuando realiza sus funciones bajo condiciones determinadas. X

Utilización de Recursos: la capacidad del producto software para utilizar cantidades y tipos de recursos apropiados cuando el software realiza su función bajo determinadas condiciones. X

Conformidad: la capacidad del producto software para adaptarse a estándares o convenciones relacionadas con la eficacia. X

5.MANTENIBILIDAD

Analizabilidad: capacidad del producto software de diagnosticar sus deficiencias o causas de fallos, o de identificar las partes que deben ser modificadas. X

Confiabilidad: capacidad del producto software de permitir implementar una modificación específica. X
La implementación incluye los cambios en el diseño, el código y la documentación.

Estabilidad: capacidad del producto software de X

evitar los defectos inesperados de las modificaciones.

Facilidad de Prueba: capacidad del producto software de permitir validad las partes modificadas. X

Conformidad: capacidad del producto software de cumplir los estándares o convenciones relativas a la mantenibilidad. X

6.PORTABILIDAD

Adaptabilidad: la capacidad del producto software para ser adaptado para ambientes determinados sin realizar acciones o aplicar medios, más que los proporcionados para este propósito para el software considerado. X

Facilidad de Instalación: la capacidad del producto software para ser instalado en un ambiente determinado. X

Coexistencia: la capacidad del producto software para coexistir con otro software independientemente en un ambiente común compartiendo recursos. X

Reemplazabilidad: la capacidad del producto software para ser utilizado en lugar de otro producto de software para el mismo propósito en el mismo ambiente. X

Conformidad: la capacidad del producto software X

para adaptarse a estándares relacionados con la portabilidad.

SUB TOTALES	2	21	76
TOTAL		99	

Fuente: Ficha de Evaluación ISO – 9126

ANEXO C: CODIGO FUENTE: ASP WEB CONFIG

```
<Project Sdk="Microsoft.NET.Sdk.Web">
  <PropertyGroup>
    <TargetFramework>netcoreapp2.0</TargetFramework>
  </PropertyGroup>
  <ItemGroup>
    <PackageReference Include="Microsoft.AspNetCore.All" Version="2.0.0" />
  </ItemGroup>
  <ItemGroup>
    <DotNetCliToolReference
      Include="Microsoft.VisualStudio.Web.CodeGeneration.Tools" Version="2.0.0" />
  </ItemGroup>
</Project>
```

ANEXO D: CODIGO FUENTE: CONTROLADOR WEB

```
/*  
  
**Paso 1. coneccion manual en DbContext  
  
using Microsoft.EntityFrameworkCore;  
  
using MySql.Data.EntityFrameworkCore;  
  
**Paso 2.  
  
Install-Package MySql.Data.Entity -Version 6.9.5  
  
Install-Package MySql.Data -Version 6.9.5  
  
*/  
  
using System;  
  
using System.Text;  
  
using System.Collections.Generic;  
  
using System.Diagnostics;  
  
using System.Threading.Tasks;  
  
using System.Linq;  
  
using Microsoft.AspNetCore.Mvc;  
  
using Microsoft.AspNetCore.Http;  
  
using coreTramDoc.Models;  
  
namespace coreTramDoc.Controllers  
  
{  
  
    public class WebController : Controller  
  
    {  
  
        DbLocal db;  
  
        //DbPosgrado pg;  
  
    }  
  
}
```

```
public WebController( DbLocal _db )
{
    db = _db;
}

public IActionResult Index()
{
    return Content("Módulo en Desarrollo");
}

[Route("Web/{id}")]
public IActionResult Desplegar( string id )
{
    id = id.ToLower();

    ///DateTime dt; dt.Date

    // buscar en BD los alias si encontramos a renderizar como WEB

    var tbl = from e in db.ugePaginas
              where e.Alias==id
              select e;

    if( tbl.Count() > 0 ){
        var row = tbl.First();

        // elementos limpios RAW

        if( row.Layer == 0 )

            return Content( row.Contenido );

        // elementos anidados subpaginas

        ViewBag.Contenido = row.Contenido;
    }
}
```

```
    } else {  
        // avisar que no hay este alias  
        ViewBag.Contenido = "Módulo en desarrollo";  
    }  
    return View("Desplegar");  
}  
[Route("Web/About")]  
[Route("Web/About/{id}")]  
public IActionResult About( string id )  
{  
    if( id == null ) id = "123";  
    ViewData["msg1"] = Crypt.doSHA1( id );  
    ViewData["msg2"] = Crypt.doPass( id );  
    return View();  
}  
[Route("Web/Transparencia")]  
public IActionResult Transparencia()  
{  
    return View();  
}  
[Route("Web/apiData/{dni}")]  
public async Task<string> apiData( string dni )  
{  
    if( dni==null || dni.Length != 8 )  
{
```

```
        return "{\"success\": false}";
    }

    string res = await LibHttp.apiPronabec( dni );

    return res;
}
}
}
```

ANEXO E: CODIGO FUENTE: CONTROLADOR NOTICIAS

```
using System;
using System.Text;
using System.Collections.Generic;
using System.Diagnostics;
using System.Threading.Tasks;
using System.Linq;
using Microsoft.AspNetCore.Mvc;
using Microsoft.AspNetCore.Http;
using coreTramDoc.Models;
namespace coreTramDoc.Controllers
{
    public class NoticiasController : Controller
    {
        DbLocal db;

        public NoticiasController( DbLocal _db )
```

```
{  
    db = _db;  
}  
  
// renderizar con Layout completo  
  
public IActionResult Index()  
{  
    ViewBag.Layout = true;  
  
    ViewBag.ugeNotis = from e in db.ugeNoticias  
                        orderby e.Id descending  
                        select e;  
  
    return View("Top");  
}  
  
// renderizar solo lista de noticias sin Layout  
  
public IActionResult Top( int id )  
{  
    ViewBag.Layout = false;  
  
    ViewBag.ugeNotis = ( from e in db.ugeNoticias  
                          orderby e.Id descending  
                          select e ).Take( id ); // Take Top N  
  
    return View();  
}  
}
```

ANEXO F: CODIGO FUENTE: CONTROLADOR ADMIN

```
using System;
using System.Text;
using System.Threading.Tasks;
using System.Linq;
using Microsoft.AspNetCore.Mvc;
using Microsoft.AspNetCore.Http;
//using System.Text.RegularExpressions;
//using System.Collections.Generic;
//using System.Diagnostics;
using coreTramDoc.Models;
namespace coreTramDoc.Controllers
{
    public class AdminController : Controller
    {
        DbLocal db;
        public AdminController( DbLocal _db )
        {
            db = _db;
        }
        public IActionResult Index()
        {
            ViewBag.Sess = GetSession();
            if ( ViewBag.Sess != null )
```

```
{  
    ViewBag.NotisN = (from e in db.ugeNoticias select e).Count();  
    ViewBag.pagisN = (from e in db.ugePaginas select e).Count();  
    return View("Index");  
}  
else  
    return View("Login");  
}  
  
//-----  
  
// Area de acciones del sistema de administración  
  
//-----  
  
public IActionResult Noticias( string id="" )  
{  
    if ( GetSession() == null ) return Logout();  
  
    //-----  
  
    String[] args;  
  
    var tbl = from e in db.ugeNoticias  
              orderby e.Id descending  
              select e;  
  
    ViewBag.ugeNotis = tbl;  
  
    ViewBag.totNotis = tbl.Count();  
  
    if( id.Length > 0 )  
    {  
        args = id.Split("=");  
    }  
}
```

```
        ViewBag.notild = StrToInt( args[1] );

        ViewBag.rowNot = db.ugeNoticias.Find( ViewBag.notild );

        if( args[0] == "duhast" ) return View("NotiAdd");

        if( args[0] == "editit" ) return View("NotiEdit");

        if( args[0] == "delete" ) return View("NotiDel");

    }

    return View();

}

[HttpPost]

public IActionResult delNoti( int notld )

{

    ugeNoticia noti;

    if( notld > 0 ){

        noti = db.ugeNoticias.Find( notld );

        db.ugeNoticias.Remove( noti );

        db.SaveChanges();

    }

    return RedirectToAction("Noticias");

}

[HttpPost]

public IActionResult GrabNoti( int notld, string archi,

                                string titul, string conte, int tipo )

{

    ugeNoticia noti;
```

```
if( notId == 0 ){  
    noti = new ugeNoticia{  
        Estado    = 1,  
        Foto      = archi,  
        Titulo    = titul,  
        Contenido = conte,  
        Fecha    = DateTime.Now  
    };  
    db.ugeNoticias.Add( noti );  
    db.SaveChanges();  
    //db.SaveChangesAsync();  
    //return "Registro insertado";  
    return RedirectToAction("Noticias");  
}  
  
noti = db.ugeNoticias.Find( notId );  
noti.Contenido = conte;  
noti.Titulo    = titul;  
noti.Foto      = archi;  
db.ugeNoticias.Update( noti );  
//db.SaveChangesAsync();  
db.SaveChanges();  
//return "Edición finalizada";  
return RedirectToAction("Noticias");  
}
```

```
private int StrToInt( string val )
{
    int value = 0;
    if(int.TryParse(val, out value))
        return value;
    else
        return value;
}

public IActionResult Paginas()
{
    if ( GetSession() == null ) return Logout();
    return View();
}

[HttpPost]
public ActionResult Login(string user, string pass, int[] valor)
{
    // foreach( int item in valor )
    // string.IsNullOrEmpty( user )

    var usr = from e in db.ugeOficinas
              where e.Usuario==user && e.Clave==Crypt.doPass(pass)
              select e;

    if ( usr.Count() == 0 )
    {
        ViewBag.Error = "(" + user + ") : Usuario incorrecto";
        return View("Login");
    }
}
```

```
}  
  
// quitar los mensajes de alerta  
ViewBag.Error = "";  
  
// datos de admin en session  
var ofi = usr.First();  
HttpContext.Session.Set<SessData>( "tmpSess", new SessData{  
    IdOfi    = ofi.Id,  
    Level    = ofi.Nivel,  
    UserName = ofi.NombOfi,  
    UserResp = ofi.Usuario  
}  
);  
  
return RedirectToAction("");  
}  
  
public ActionResult Logout()  
{  
    SessData sess = HttpContext.Session.Get<SessData>( "tmpSess" );  
  
//if( sess!=null )  
  
// logAdd( sess.UserResp, "Logout" );  
  
    HttpContext.Session.Clear();  
  
    ViewBag.Error = null;  
  
    return RedirectToAction("");  
}  
  
private SessData GetSession()  
{
```

```
return HttpContext.Session.Get<SessData>( "tmpSess" );  
  
}  
  
//-----  
  
// Area de acciones del sistema de conex  
  
//-----  
  
public async Task<string> apiDatos( string id )  
{  
    return await LibHttp.apiSunaDatos( id );  
}  
  
public async Task<string> apiBecas( string id )  
{  
    return await LibHttp.apiPronabec( id );  
}  
}  
}
```

ANEXO G: CODIGO FUENTE: CONTROLADOR SESIONES

```
using System;
using System.Text;
using System.Threading.Tasks;
using Microsoft.AspNetCore.Http;
using Newtonsoft.Json;
namespace coreTramDoc.Controllers
{
    // solo sesion local sin DB
    public class SessData
    {
        public int IdOfi;
        public int Level;
        public string UserName;
        public string UserResp;
    }
    public static class Crypt
    {
        // funciones de Extension para clase Session usar Objetos
        public static void Set<T>(this ISession session, string key, T value)
        {
            session.SetString( key, JsonConvert.SerializeObject(value) );
        }
        public static T Get<T>(this ISession session, string key)
```

```
{  
  
    var value = session.GetString( key );  
  
    return (value == null) ?  
  
        default(T) :  
  
        JsonConvert.DeserializeObject<T>( value );  
  
}  
  
    public static string doSHA1( string input )  
  
    {  
  
        if (input == null) return "";  
  
        var sha1 = System.Security.Cryptography.SHA1.Create();  
  
        byte[] b = Encoding.UTF8.GetBytes(input);  
  
        b = sha1.ComputeHash(b);  
  
        StringBuilder sb = new StringBuilder();  
  
        foreach (byte x in b)  
  
            sb.Append( x.ToString("x2" ) );  
  
        return sb.ToString().ToUpper();  
  
    }  
  
    // * += 2 x SHA1  
  
    public static string doPass( string input )  
  
    {  
  
        if (input == null) return "";  
  
        var sha1 = System.Security.Cryptography.SHA1.Create();  
  
        byte[] b = Encoding.UTF8.GetBytes(input);  
  
        b = sha1.ComputeHash(b);  
  
        b = sha1.ComputeHash(b);  
  
    }
```

```

        StringBuilder sb = new StringBuilder();

        foreach (byte x in b) // Hex00

            sb.Append( x.ToString("x2") );

        return "*" + sb.ToString().ToUpper();

    }

}

```

ANEXO H: CODIGO FUENTE: CONTROLADOR HTTP REQUESTS

```

using System;

using System.Linq;

using System.Collections.Generic;

using System.Threading.Tasks;

using Microsoft.AspNetCore.Mvc;

using System.Net.Http;

using System.Net.Http.Headers;

using Newtonsoft.Json;

using System.Text;

public static class LibHttp
{
    /*

    public static Task<string> apiReniDatos( string numdni )

    {

        var args = new Dictionary<string, string>

        {

            { "numdni", numdni },


```

```

        { "certif", "OF231" }
    };

    HttpContent form = new FormUrlEncodedContent( args );
    return GetHttpPost( "http://softinge.com/reniec/", form );
}

*/

public static Task<string> apiPronabec( string numdni )
{
    var args = new Dictionary<string, string>
    {
        { "numeroDocumento", numdni }
    };

    HttpContent form = new FormUrlEncodedContent( args );

    x = postulaciones

return GetHttpPost("https://x.pronabec.gob.pe/bap2019/Base/GetReniec",form)
;
}

public static Task<string> apiSunaDatos( string numdni )
{
    // ?codTipDocIdentidad=01&numDocumento=01010101

    return GetHttpGet( "https://www.sunat.gob.pe/ol-at-
itseleccion/acceso/buscarPersona?txtCaptcha=ZZa4&codTipDocIdentidad=01&
numDocumento=" + numdni, "application/json" );
}

//-----

```

```
private static async Task<string> GetHttpPost( string url, HttpContent form )
{
    HttpClient http = new HttpClient();

    using( HttpResponseMessage resp = await http.PostAsync(url,form) )
    {
        //HttpContentHeaders header = content.Headers;

        HttpContent content = resp.Content;

        string result = await content.ReadAsStringAsync();

        return result;

        // JsonConvert.SerializeObject( PostJS );
        // JsonConvert.DeserializeObject<PostJS>(mycontent);
    }
}

//-----

private static async Task<string> GetHttpGet( string url, string accept )
{
    HttpClient http = new HttpClient();

    if( accept != null )

        http.DefaultRequestHeaders.Accept.Add(new MediaTypeWithQualityHeaderValue(accept));

    var response = await http.GetAsync( url );

    var content = await response.Content.ReadAsStringAsync();

    return content;
}
```



ANEXO I: MANUAL DE USUARIO

SISTEMA DE ADMINISTRACION DE CONTENIDO WEB

Prof. Mario Zavaleta Quispe

Director de la Unidad de Gestión Educativa Local de Canas

Ing. William Rene Quispe Aquino

Responsable de la Oficina de Estadística

Bach. Delia Apaza Campos

Responsable del desarrollo de la Pagina Web

“La tecnología es sólo una herramienta. En términos de motivar a los niños y lograr que trabajen juntos, el profesor es el recurso más importante”.

Bill Gates

INGRESO AL ADMINISTRADOR

Una vez dentro de la pagina web se procede a hacer click en la barra de menú en la opcion ADMIN.

Seguidamente obtendremos la ventana de validación de usuarios.

ugelcanas.gob.pe/admin

UNIDAD DE GESTIÓN EDUCATIVA LOCAL CANAS - UE 313

FUT facebook Portal de Transparencia

INICIO INSTITUCIONAL ESTRUCTURA ORGÁNICA CONVOCATORIAS NORMATIVIDADES NOTICIAS ADMIN

UGEL CANAS - CUSCO

LOGIN
Verificación de datos

Usuario
Ingrese su **usuari**

Contraseña
ingrese su clave

Verificar

[¿Olvidó su contraseña?](#)

CONVOCATORIAS	NORMATIVIDAD	DOCUMENTOS DE GESTIÓN	ENLACES DE INTERÉS
Docente	Resoluciones	TUPA	Ubicación Y Anexos
Administrativo	Directivas	POI	Galería De Fotos
CAS	Instructivos	CAP	Galería De Videos
Auxiliar De Educación	Oficios	MOF	Actividades Oficiales
Todos Las Convocatorias	Todas Las Normatividades	MAPRO	Buzón De Sugerencias
		Todos Los Documentos	

1. Ingrese nombre de usuario
2. Ingrese clave asignada
3. Presione <ENTER> o click en el botón <VERIFICAR>

Panel principal

El panel de control se divide en dos areas:

- a) A la izquierda la barra de menús, para seleccionar los elementos a trabajar (Inicio, Noticias o Páginas)
- b) A la derecha el panel de despliegue de datos. (inicialmente un listado y conteo de elementos registrados en la página web)



**UNIDAD DE GESTIÓN
EDUCATIVA LOCAL
CANAS - UE 313**





INICIO
INSTITUCIONAL ▾
ESTRUCTURA ORGÁNICA
CONVOCATORIAS ▾
NORMATIVIDADES ▾
NOTICIAS
ADMIN

Inicio

Noticias

Páginas

Salir

Progreso histórico

Total de elementos ingresado como Noticias	135
elementos ingresado por el administrador	Total

| Total de elementos ingresado como Páginas | 3 |
| elementos editados por el administrador | Total |

Fuente :

Ratio de tráfico en el sitio web

Fuente : Our Robots

Ventajas del Sistema

Ingreso de Documento
Reportes de Movimiento
Acceso por Datos Personales
Rápido e Intuitivo



ADMINISTRAR NOTICIAS

La vista general se observa así:



**UNIDAD DE GESTIÓN
EDUCATIVA LOCAL
CANAS - UE 313**





INICIO
INSTITUCIONAL ▾
ESTRUCTURA ORGÁNICA
CONVOCATORIAS ▾
NORMATIVIDADES ▾
NOTICIAS
ADMIN

Inicio

Noticias

Páginas

Salir

Mantenimiento de Elementos de Noticias

Nro	Tipo	Título	Fecha	Opciones
135	1	COMUNICADO	11/10/2019 0:00:00.	+ ✎ ✖
134	1	CONVOCATORIA CETPRO	10/10/2019 0:00:00.	+ ✎ ✖
133	1	CONVOCATORIA CAS	10/10/2019 0:00:00.	+ ✎ ✖
132	1	OFICIO MÚLTIPLE N° 176-2019: INVITA A LAS ACTIVIDADES POR LA SEMANA DE EDUCACIÓN FÍSICA	03/10/2019 0:00:00.	+ ✎ ✖
131	1	OFICIO MÚLTIPLE N° 174-2019: DÍA DE LA EDUCACION INCLUSIVA.	03/10/2019 0:00:00.	+ ✎ ✖
130	1	CONVOCATORIA CONTRATA DOCENTE 2019	02/10/2019 0:00:00.	+ ✎ ✖
129	1	COMUNICADO A LOS POSTULANTES DEL PROCESO DE NOMBRAMIENTO 2019	01/10/2019 0:00:00.	+ ✎ ✖
128	1	COMUNICACO NOMBRAMIENTO DOCENTE	01/10/2019 0:00:00.	+ ✎ ✖
127	1	CONVOCATORIA	30/09/2019 0:00:00.	+ ✎ ✖
126	1	CONVOCATORIA	30/09/2019 0:00:00.	+ ✎ ✖
125	1	Oficio Múltiple 170- 2019	30/09/2019 0:00:00.	+ ✎ ✖
124	1	Oficio Múltiple 0042-2019 MINEDU/VMGP-DIGEDD	30/09/2019 0:00:00.	+ ✎ ✖
123	1	DIRECTIVA N° 19 -2019: PROCEDIMIENTO PARA LA TOMA DE INVENTARIO FÍSICO DE BIENES MUEBLES	26/09/2019 0:00:00.	+ ✎ ✖
122	1	COMUNICADO PROCESO DE RACIONALIZACIÓN 2019	24/09/2019 0:00:00.	+ ✎ ✖

92

Repositorio Institucional UNA-PUNO

No olvide citar esta tesis

LISTADO DE NOTICIAS:

Contador, Tipo de noticia por prioridad, título, fecha y opciones por item.

Nro	Tipo	Título	Fecha	Opciones
135	1	COMUNICADO	11/10/2019 0:00:00.	  

AGREGAR EDITAR BORRAR



EDITOR DE ITEMS NOTICIAS

Ya sea cuando se ingrese un nuevo item o para editar, la vista rapida de edicion se muestra de la siguiente forma.

Editar

Archivo

Título

Contenido

Tipo de Publ.

Nota : Actualice la pagina para ver resultados