

UNIVERSIDAD NACIONAL DEL ALTIPLANO

FACULTAD DE INGENIERÍA ESTADÍSTICA E INFORMÁTICA

ESCUELA PROFESIONAL DE INGENIERÍA ESTADÍSTICA E INFORMÁTICA



SISTEMA DE ADMINISTRACIÓN Y EVALUACIÓN INTEGRAL ONLINE PARA EL PROCESO DE CONTRATACIONES EN INSTITUCIONES PÚBLICAS Y PRIVADAS

TESIS

PRESENTADA POR:

Bach. MIRIAN NOEMI TAPIA ARPASI

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

SISTEMA DE ADMINISTRACIÓN Y EVALUACIÓN INTEGRAL ONLINE PARA
 EL PROCESO DE CONTRATACIONES EN INSTITUCIONES
 PÚBLICAS Y PRIVADAS

TESIS PRESENTADA POR:

Bach. MIRIAN NOEMI TAPIA ARPASI

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. CHARLES MENDOZA MOLLOCONDO

SEGUNDO MIEMBRO:


 M.Sc. LEONID ALEMAN GONZALES

DIRECTOR/ASESOR:


 M.Sc. ELQUI YEYE PARI CONDORI

Área : Sistemas de información y bases de datos

Tema : Sistemas de información

Fecha de Sustentación: 16 de octubre del 2019

DEDICATORIA

El presente trabajo de investigación es dedicado en primer lugar a Dios, por ser el inspirador y brindarme las fuerzas necesarias para continuar en este proceso de obtener uno de mis anhelos más deseados.

A mi padre y mi madre, por su amor, trabajo y sacrificio en todos estos años, gracias a ustedes he logrado llegar hasta donde me encuentro y convertirme una persona de bien. Me inunda el orgullo y el privilegio de ser vuestra hija, son los mejores padres. Y a mi hermana por estar siempre presente al lado mío, acompañándome y apoyándome moralmente. Que me proporcionaste a lo largo de esta etapa de mi vida.

A todas las personas de la facultad, compañeros y docentes que me han apoyado y me han compartido sus conocimientos, en el trascurso del desarrollo de mi perfil profesional.

AGRADECIMIENTOS

Agradecer a Dios por bendecirme con la vida, por guiarme a lo largo de mi vida, ser el apoyo y fortaleza en aquellos momentos de dificultad y de debilidad.

Gracias a mis padres: Porfirio y Elisa, por confiar y creer en mis expectativas, por los consejos, valores y principios que me han inculcado. Y agradecer a mi hermana, Verónica, por el apoyo moral recibido por parte suya.

Agradezco a mis docentes de la Facultad de Ingeniería Estadística e Informática, por haber compartido sus conocimientos a lo largo de la preparación de mi profesión. También agradezco a mis amigos(as) y compañeros(as), por el apoyo, en el desarrollo de la ampliación de mis conocimientos.

ÍNDICE GENERAL

	Pág.
ÍNDICE DE FIGURAS	
ÍNDICE DE TABLAS	
ÍNDICE DE ACRÓNIMOS	
RESUMEN	10
ABSTRACT	11
CAPÍTULO I	
INTRODUCCIÓN	
1.1 PLANTEAMIENTO DEL PROBLEMA	13
1.2 JUSTIFICACION	14
1.3 OBJETIVOS	15
1.3.1 Objetivo general	15
1.3.2 Objetivos específicos	15
1.4 HIPÓTESIS	15
CAPÍTULO II	
REVISIÓN DE LITERATURA	
2.1 ANTECEDENTES	16
2.2. MARCO TEÓRICO.....	22
CAPÍTULO III	
MATERIALES Y MÉTODOS	
3.1 LUGAR DE ESTUDIO	32
3.2 POBLACIÓN	32
3.3 MUESTRA.....	32
3.4 METODO DE RECOLECCION DE DATOS	32
3.5 METODO DE INVESTIGACION.....	33
3.6 MODELO Y METODOLOGÍA DE DESARROLLO DE SOFTWARE	33
3.6.1 Modelo de desarrollo.....	33
3.6.2 Fase análisis:	34
3.6.3 Fase de diseño:.....	34
3.6.4 Fase de desarrollo	35

3.6.5	Fase pruebas	35
-------	--------------------	----

CAPÍTULO IV

RESULTADOS Y DISCUSIÓN

4.1	ANALIZAR Y DISEÑAR E IMPLEMENTAR UN SISTEMA WEB, BASADO EN FASES DE DESARROLLO	36
4.1.1	Análisis.....	36
4.1.2	Análisis de viabilidad.....	36
4.1.3	Análisis de roles	37
4.2	ACCESO Y VALIDACIÓN DEL SISTEMA.....	38
4.3	PROCESOS DE LA METODOLOGÍA CON MODELAMIENTO UML ..	39
4.3.1	Diseño del sistema Web.....	39
4.3.2	Modelamiento del sistema Web	40
4.3.3	Arquitectura de desarrollo	42
4.3.4	Versiones e iteraciones	45
4.3.5	Codificación.....	48
4.3.6	Pruebas.....	49
4.3.7	Instalación	49
4.3.8	Resultados de la prueba según ISO-9126	50
4.3.9	Cuadro comparativo según el tiempo resultados utilizando el sistema y sin el sistema	51
4.3.10	Resultados de satisfacción de usuario	52
	CONCLUSIONES	55
	RECOMENDACIONES.....	56
	REFERENCIAS BIBLIOGRÁFICAS.....	57
	ANEXOS.....	60
	ANEXO 1: MANUAL DEL SISTEMA.....	61
	ANEXO 2: EVALUACION DEL SISTEMA	63
	ANEXO 3: CÓDIGO FUENTE DEL SISTEMA	67

ÍNDICE DE FIGURAS

	Pág.
Figura 1: Ingreso y referenciación de datos del Usuario	39
Figura 2: Diagrama de caso de uso general del sistema	40
Figura 3: Diagrama de secuencia del sistema	41
Figura 4: Diagrama de base de datos	41
Figura 5: Esquema modelo vista controlador y la interacción entre sus componentes.....	43
Figura 6: Controlador del Software	43
Figura 7: Modelo del Software	44
Figura 8: Vista del Software	44
Figura 9: Considera Ud. Que el Sistema de Evaluación es amigable	52
Figura 10: Considera que el Sistema tiene facilidad De uso	52
Figura 11: Número de Usuarios según La Seguridad del Sistema.....	53
Figura 12: Número de Usuarios según la Claridad de la Información	53
Figura 13: Número de Usuarios según el Tiempo de Respuesta del Sistema .	54
Figura 14: Número de Usuarios según la Calidad de Soporte	54

ÍNDICE DE TABLAS

	Pág.
Tabla 1 : Cuadro de decisiones ISO - 9126	50
Tabla 2: Cuadro comparativo expresado en minutos y horas con y sin Sistema de Evaluación.....	51

ÍNDICE DE ACRÓNIMOS

DNI	: Documento Nacional de Identidad
EBIT	: Earnings Before Interest and Taxes
ECMA	: European Computer Manufacturers Association
GRASP	: Greedy Randomized Adaptive Search
HTML	: Hyper Text Markup Language
MGIP	: Manual Modelo de Gestión Integral por Procesos
MVC	: Model View Controller
PMBOK	: Project Management Body of Knowledge
PMO	: Project Management Office
PRINCE	: Projects in Controlled Environments
SGBD	: Sistema Gestor de Base de Datos
SGML	: Standard Generalized Markup Language
SRL	: Sociedad de Responsabilidad Limitada
UML	: Unified Modeling Language
WWW	: World Wide Web

RESUMEN

La presente investigación estuvo orientada con la finalidad de mejorar el proceso de evaluación y contratación de personal para áreas administrativas en empresas públicas o privadas, aprovechando los sistemas de información en su alcance y accesibilidad no limitada a la geografía y estrato social. El propósito de este sistema es posibilitar el registro, validación, evaluación, publicación de resultados e información de concurso de personal. Para la metodología de desarrollo de software se ha empleado la metodología Programación Extrema XP por su mayor afinidad y claridad de actividades y pone más énfasis en la adaptabilidad que en la previsibilidad, El objetivo de esta metodología de desarrollo ágil, está es la programación iterativa y retroalimentada aplicando de manera dinámica durante el ciclo de vida del software. Se ha concluido con resolver el análisis e implementación de un sistema de evaluación electrónica además que permita un seguimiento y visualización de las calificaciones obtenidas, así tener lineamientos para favorecer en la toma de decisiones en la selección del personal con el sistema de administración y evaluación integral online para el proceso de contrataciones en instituciones públicas y privadas mejorando y agilizando este proceso.

Palabras Clave: Evaluación Integral, Sistema de Administración, proceso de contrataciones, instituciones públicas y privadas.

ABSTRACT

This research was aimed at improving the process of evaluation and hiring of personnel for administrative areas in public or private companies, taking advantage of the information systems in their scope and accessibility not limited to geography and social stratum. The purpose of this system is to enable the registration, validation, evaluation, publication of results and personnel contest information. For the software development methodology, the XP Extreme Programming methodology has been used for its greater affinity and clarity of activities and places more emphasis on adaptability than on predictability. The objective of this agile development methodology is this is iterative programming and feedback applied dynamically during the software life cycle. It has been concluded with solving the analysis and implementation of an electronic evaluation system in addition to allowing a follow-up and visualization of the qualifications obtained, thus having guidelines to favor in the decision-making process in the selection of personnel with the administration and integral evaluation system online for the process of hiring in public and private institutions improving and speeding up this process.

Keywords: Integral Evaluation, Administration System, contracting process, public and private institutions.

CAPÍTULO I

INTRODUCCIÓN

El rápido avance de la tecnología, la transmisión de datos y disponibilidad de información son características de estas últimas décadas, esto ha originado el crecimiento de las fuentes de información, donde generalmente es común que personas de diferentes sectores acudan a consultar e intercambiar información. La facilidad con que ahora se tiene acceso a las redes de computadoras y el fuerte impulso que le dio el WWW a Internet.

Estos elementos facilitan el procesamiento y la distribución de la información, la cual es utilizada por los millones de usuarios en todo el mundo que demandan tanto almacenamiento como recuperación de información, aprovechando el alcance de los sistemas de información basados en navegador web.

Mediante la implementación de un sistema que administre los postulantes y la evaluación electrónica de los participantes se podrá culminar con uno de los criterios de evaluación de conocimientos así poder brindar un instrumento rápido de validación del mismo. Siempre teniendo en cuenta elementos de flexibilidad y ergonomía durante el proceso de desarrollo.

1.1 PLANTEAMIENTO DEL PROBLEMA

Un inconveniente general en las empresas es el control de personal a evaluar para contrato, es por este motivo que nos propusimos el desarrollo de un sistema informático capaz de administrar esta información así poder ofrecer acceso rápido a esta información.

Se requería un software que permita controlar el proceso de evaluación de postulantes, así como el registro y reporte de los mismos, entonces el problema requiere un planteamiento de la estructura de datos a ser recolectadas, así como las herramientas a utilizar para el desarrollo de la herramienta de recolección de datos, seguidamente el procesamiento el almacenamiento y la depuración de estos datos para evitar la duplicidad y las menores cantidades de errores durante la recolección de datos.

La información será almacenada en un sistema de gestión de base de datos y la interfaz de consulta y operación será codificado para un entorno accesible y de fácil visualización como aplicación web.

Lo que nos lleva a la pregunta de investigación: **¿En qué medida el desarrollo del sistema de administración y evaluación integral online para el proceso de contrataciones en instituciones públicas y privadas, mejoró el manejo de registro y reporte de postulantes a contrato?**

1.2 JUSTIFICACION

Al iniciar la automatización e implementación del sistema web, se logró manejar el proceso de evaluación durante el proceso de contrataciones de forma eficaz y rápida, beneficiando a los participantes a las mismas, quienes pueden ingresar la información progresivamente en el sistema, la misma que esta presentada de una forma organizada, permitiendo cumplir con los parámetros básicos de forma necesaria; además se puede generar un documento que conste el registro del participante en los concursos el mismo que es visible e imprimible además contempla los formatos aprobados por el sistema web. El sistema cuenta con un módulo de seguridad, en el cual los administradores y evaluadores tienen sus cuentas y claves necesarias para el acceso al sistema, con esto podemos transparentar sus respectivos procesos.

Tengamos en cuenta que los Sistemas de Información contribuyen al acceso instantáneo de la información por un grupo grande de usuarios, así como la interacción entre ellos para el caso particular de la toma de un examen que requiera calificación inmediata y confiable.

Finalmente, el sistema aporta información útil y relevante sobre cada proceso, permitiendo mantener la comunicación estrecha, entre evaluador y evaluado, haciendo en primer lugar que la distancia no sea un obstáculo que impida el avance progresivo.

1.3 OBJETIVOS

1.3.1 Objetivo general

Implementar un sistema de administración y evaluación integral online para el proceso de contrataciones en instituciones para administrar, registrar, validar, evaluar y publicar los resultados de las evaluaciones para el proceso del examen y posterior ranking y toma de decisiones para contrataciones.

1.3.2 Objetivos específicos

- Analizar, diseñar e implementar un sistema web, basándose en los requerimientos básicos de la metodología.
- Validar y garantizar la calidad de la información obtenida de los postulantes como usuarios del sistema.
- Evaluar los resultados de los participantes para minimizar el tiempo de calificación y publicación de los resultados de los concursos de personal.

1.4 HIPÓTESIS

El sistema de administración y evaluación integral online para el proceso de contrataciones en Instituciones Públicas y Privadas mejora y agiliza el proceso de contrataciones en Instituciones.

CAPÍTULO II

REVISIÓN DE LITERATURA

2.1 ANTECEDENTES

Sobrevilla (2017) plantean que desarrollar un Sistema de Localización Subterránea utilizando Scrum y Prácticas de Ingeniería de Software para la Mejora Continua del Desarrollo de un Sistema Ciber-Físico, no fue tan ágil como se esperaba y que necesario contemplar la arquitectura de software en todos los sistemas, y el hecho de que un proyecto se desenvuelva en un ambiente ágil no implica que no se deba hacer trabajo de arquitectura. Además, se detectaron y arreglaron algunos problemas de comunicación al reunir a los equipos en el mismo lugar y estableciendo un nuevo canal de comunicación. Por otro lado, tratar con entregas frecuentes e incrementales fue complicado al inicio, esta situación desmotivó al equipo y los miembros quisieron abortar su práctica. Afortunadamente en un periodo corto empezaron a percibir el valor de las mismas.

Nájera (2016) plantea el Desarrollo de un modelo integrado de procesos para la gestión de proyectos basada en PMBOK que integre los estándares profesionales de gestión de proyectos más reconocidos a nivel mundial como es el PMBOK (Project Management Body of Knowledge), el método PRINCE (Projects in Controlled Environments) e ISO 21500, de forma que permita la gestión de proyectos aprovechándose del conocimiento de las tres estrategias

de gestión y facilitando además la colaboración entre diferentes empresas y profesionales que utilicen estos métodos individualmente, concluyendo con el desarrollo del método MGIP, desde el punto de vista de los principios a cumplir, las áreas de conocimiento a tener en cuenta y ,sobre todo, se han diseñado los procesos adecuados para que una organización pueda adoptarlo como método prescriptivo de gestión de proyectos. Ello es así porque este método puede ser aplicado tanto considerado desde un punto de vista sectorial como desde una óptica de un proyecto más concreto. En lo relativo a la creación de procesos, se han desarrollado un total de 72 procesos

Carrera (2016) plantea una propuesta de un modelo de gestión de proyectos de Ingeniería de Software que aplicó a un caso de estudio, con el objetivo principal de identificar y documentar un conjunto de buenas prácticas de gestión de proyectos y de ingeniería de software que contribuyan a mejorar la capacidad y operatividad de los directores de proyectos de software y de sus equipos, concluyendo que luego de revisar, analizar e identificar las normas, estándares y buenas prácticas de proyectos de ingeniería de software son un aporte esencial para potenciar la gestión exitosa y otorgar productos de software que satisfacen los requerimientos de los clientes, también fue posible construir un modelo de gestión de proyectos de ingeniería de software. El modelo resulto aplicable y eficiente para gestionar y ejecutar proyectos de ingeniería de software, se logró disminuir la incertidumbre, aumentar la productividad, mantener una comunicación adecuada con los interesados, aumentar la probabilidad de éxito

Calvo y Ramírez (2016) plantean una Guía Metodológica para la Gestión de Proyectos en la Dirección General del Servicio Civil, con el objetivo de responder a la problemática de implementación de los proyectos del plan estratégico de forma estandarizada, llegando a la conclusión que el marco referencia brindado por la Norma INTE/ISO 21500: 2013. Directrices para la Dirección y Gestión de Proyectos, el cual se seleccionó debido a que se ajusta a las necesidades definidas por la organización, en lo que respecta a parámetros como relación geográfica, traducción al español, versión actualizada, aplicabilidad a proyectos de gestión pública, respaldo de alguna institución en el país y sobre todo la compatibilidad con normas que hayan sido adaptadas y utilizadas anteriormente por la DGSC.

Medina (2016) plantea el Diseño de Marco Ágil para la Dirección de Proyectos de desarrollo de producto en una EBIT integrando las mejores prácticas de PMBOK 5ta Edición a la metodología Scrum, donde se integren los parecidos y excluir las diferencias entre ellas y lo más fundamental, se realce la capacidad adaptativa y otras características que sean necesarios, llegó a la conclusión que las estructuras organizacionales son uno de los elementos que se deben tener en cuenta a la hora de gestionar un proyecto, ya que es un componente que puede afectar de forma muy significativa a la disponibilidad u optimización de recursos, y de hecho no menos importante, poder influir de forma determinante en el modo de dirigir los proyectos dentro de una organización, incluso, para complementar el conocimiento técnico que tienen los profesionales que laboran en cada una de las unidades funcionales y equipos de trabajo en la EBIT.

Chaparro y Forero (2005) Diseñó e Implementó una Aplicación Web para la administración de proyectos de Investigación, Innovación y desarrollo.

Borjas (2013) “Análisis, diseño e implementación de un sistema de información para la administración de horarios y rutas en empresas de transporte público” (p.08). El objetivo principal es Analizar, diseñar e implementar un sistema de información en un entorno Web, que permita la administración de rutas y la generación manual o automatizada de los horarios del sistema de transporte público por medio de un algoritmo meta heurístico GRASP Construcción.

Castillo (2016) “Desarrollo e implementación de un sistema web para generar valor en una pyme aplicando una metodología ágil. Caso de estudio: Manufibras Pérez SRL” (p.06). El objetivo principal es Desarrollar un sistema web mediante la aplicación de una metodología ágil para la empresa Manufibras Pérez SRL, con la finalidad de generar valor con la mejora en la promoción de productos, gestión de pedidos y el registro de ventas.

Atará (2016) obtienen conceptos clave para la Gestión de Proyectos Tecnológicos para determinar el nivel de competitividad que hay actualmente en las organizaciones para observar, evaluar y proponer soluciones a la estrategia organizacional, concluye que a pesar de que existe un gran número de profesionales que utilizan metodologías para el modelado y gestión de la información, el uso debería ser mayor si es que no en un 100% de los

profesionales, ya que son herramientas que ayudan a identificar el modelo de negocio de la empresa para poder desarrollar soluciones tecnológicas que apoyen la estrategia de la organización. Para evitar el fracaso del proyecto, un gestor de proyecto de software y los ingenieros de software que construyen el producto deben seguir un conjunto de señales de advertencia comunes, comprender los factores de éxito críticos que conducen a una buena gestión del proyecto y desarrollar un enfoque de sentido común para planificar, supervisar y controlar el proyecto.

Barrantes (2015) formula una propuesta Metodológica para la Gestión de Proyectos de Desarrollo de Software en la Organización XYZ.S.A., basada en los estándares y mejores prácticas del PMBOK, con el fin de optimizar los procesos actuales y hacer un mejor uso de los recursos; Se desarrolló un procedimiento para la implementación del grupo de procesos de Iniciación, con los pasos necesarios para definir el alcance y recursos iniciales del proyecto. Se estableció el procedimiento para la implementación del grupo de procesos de Ejecución que permitirá completar el trabajo definido en el plan de dirección.

Benites (2015) formulan una propuesta de Diseño y Despliegue de una Oficina de Gestión de Proyectos (PMO) en una Entidad Bancaria Peruana para los Proyectos de Desarrollo de Sistemas, donde las actividades propias de gestión de proyectos no se incluyan en el proceso de desarrollo del software, lo que permitirá, se tenga una visión amplia de los avances, problemas y necesidades de cada proyecto, permitiendo, reaccionar de manera oportuna

para ayudar a finalizar los proyectos satisfactoriamente según las necesidades definidas por los interesados y lo esperado por la institución; concluye con el éxito de la implementación de la PMO tiene como factor crítico el apoyo de la alta dirección, acompañado de un plan para gestionar el cambio organizacional que evidencie la necesidad de esta implementación para el logro de los objetivos estratégicos y permita obtener ventajas competitivas que impacten directamente en los resultados del negocio.

Chachaque (2015), "Sistema web para la administración académica del Instituto Superior de Educación Pública "Divino Jesús" Chumbivilcas 2014" (p.03). En el cual se concluyó que el sistema web para la administración académica del Instituto Superior de Educación Pública "Divino Jesús" Chumbivilcas 2014, así mismo se recortó el tiempo de espera de los alumnos, agilizando el trabajado del secretario académico. Para desarrollar el sistema web se utilizó la metodología ágil de XP, también se utilizó la metodología LIKERT para la validación del software. Valeriano G. (2015), "Sistema Web de Administración para la Gestión de Empresas de Servicios de Comercialización Puno - 2014" (p.06). Desarrollar un Sistema Web de administración para mejorar la calidad de servicio de atención de empresas de comercialización de productos en la región Puno.

2.2. MARCO TEÓRICO

A. Aplicaciones WEB

HTML5 está reemplazando previos complementos o plugins, como Flash o Java applets, por nuevas y mejores tecnologías ahora incluso WebGL y Canvas para representación gráfica de datos. El desarrollo web contribuye a la simplificación de la interfaz de desarrollo sobre el problema real de transportabilidad del programa final, como el caso particular de un correo electrónico o página web que únicamente requiere ejecutarse sobre un navegador web, dejando al desarrollador concentrarse en el algoritmo y el objetivo planteado. Las aplicaciones web están diseñadas para apoyar el desarrollo de sitios web dinámicos, aplicaciones y servicios web. Las aplicaciones Web intentan aliviar el exceso de carga asociado con actividades comunes usadas en desarrollos web. Por ejemplo, muchos de ellos proporcionan bibliotecas para acceder a bases de datos, administración general de archivos y carpetas, mantenimiento de ellas, estructuras para plantillas y gestión de sesiones, y con frecuencia facilitan la reutilización de código. (Pautasso et al., 2014)

B. Javascript

A principios de los años 90, la mayoría de usuarios que se conectaban a Internet lo hacían con módems a una velocidad máxima de 28.8 kbps. En esa época, empezaban a desarrollarse las primeras aplicaciones web y, por tanto, las páginas web comenzaban a incluir formularios complejos. (Upton. D., 2015)

Con unas aplicaciones web cada vez más complejas y una velocidad de navegación tan lenta, surgió la necesidad de un lenguaje de programación que se ejecutara en el navegador del usuario. De esta forma, si el usuario no rellenaba correctamente un formulario, no se le hacía esperar mucho tiempo hasta que el servidor volviera a mostrar el formulario indicando los errores existentes.

Brendan Eich, un programador que trabajaba en Netscape, pensó que podría solucionar este problema adaptando otras tecnologías existentes (como ScriptEase) al navegador Netscape Navigator 2.0, que iba a lanzarse en 1995. Inicialmente, Eich denominó a su lenguaje LiveScript.

Posteriormente, Netscape firmó una alianza con Sun Microsystems para el desarrollo del nuevo lenguaje de programación. Además, justo antes del lanzamiento Netscape decidió cambiar el nombre por el de JavaScript. La razón del cambio de nombre fue exclusivamente por marketing, ya que Java era la palabra de moda en el mundo informático y de Internet de la época.

La primera versión de JavaScript fue un completo éxito y Netscape Navigator 3.0 ya incorporaba la siguiente versión del lenguaje, la versión 1.1. Al mismo tiempo, Microsoft lanzó JScript con su navegador Internet Explorer 3. JScript era una copia de JavaScript al que le cambiaron el nombre para evitar problemas legales.

Para evitar una guerra de tecnologías, Netscape decidió que lo mejor sería estandarizar el lenguaje JavaScript. De esta forma, en 1997 se envió la especificación JavaScript 1.1 al organismo ECMA (European Computer Manufacturers Association).

ECMA creó el comité TC39 con el objetivo de "estandarizar de un lenguaje de script multiplataforma e independiente de cualquier empresa". El primer estándar que creó el comité TC39 se denominó ECMA-262, en el que se definió por primera vez el lenguaje ECMAScript.

Por este motivo, algunos programadores prefieren la denominación ECMAScript para referirse al lenguaje JavaScript. De hecho, JavaScript no es más que la implementación que realizó la empresa Netscape del estándar ECMAScript.

La organización internacional para la estandarización (ISO) adoptó el estándar ECMA-262 a través de su comisión IEC, dando lugar al estándar ISO/IEC-16262. (Upton. D., 2015).

C. Programación Extrema (XP)

“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.” (Letelier & Penadés, 2006). El

principal objeto de XP es la continua interacción entre el cliente y el equipo desarrollador, una comunicación correctamente organizada entre ellos puede asegurar un producto de calidad y la implementación de soluciones certeras. (Letelier & Penadés, 2006)

D. Framework CodeIgniter

CodeIgniter fue creada por la empresa Ellislab ubicada en Oregon – EEUU que desarrolla aplicaciones en PHP. La primera versión de CodeIgniter fue publicada el 28 de febrero del 2006 y la última versión estable 2.2.0 el 5 de junio del 2014. CodeIgniter generalmente abreviado CI, es un framework para aplicaciones web de código abierto desarrollado en PHP y que nos brinda un conjunto de librerías que nos ayudan a llevar a cabo desarrollos más rápido que empezando desde cero. Como cualquier otro framework, CodeIgniter contiene una serie de librerías que sirven para el desarrollo de aplicaciones web y además propone una manera de desarrollarlas que debemos seguir para obtener provecho de la aplicación. Esto es, marca una manera específica de codificar las páginas web y clasificar sus diferentes scripts, que sirve para que el código esté organizado y sea más fácil de crear y mantener. CodeIgniter implementa el proceso de desarrollo llamado Model View Controller (MVC), que es un estándar de programación de aplicaciones, utilizado tanto para hacer sitios web como programas tradicionales. (Upton. D., 2015).

E. Pruebas de Software

Pruebas del software El objetivo de la prueba del software es asegurar la apropiada navegación dentro del sistema, ingreso de datos, procesamiento y recuperación apropiada de datos, y la implementación apropiada de las reglas de negocio. Este tipo de pruebas se basan en técnicas de caja negra, esto es, verificar el sistema (y sus procesos internos), la interacción con las aplicaciones que lo usan vía GUI y analizar las salidas o resultados. En esta prueba se determina que pruebas de sistema (usabilidad, volumen, desempeño, etc.) aseguran que la aplicación alcanzara sus objetivos. (Sommerville, I., 2005)

La prueba del sistema incluye:

- Prueba funcionalidad
- Prueba usabilidad
- Prueba de performance
- Prueba de documentación y procedimientos
- Prueba de seguridad y controles
- Prueba de volumen
- Prueba de esfuerzo (Stress)
- Prueba de recuperación
- Prueba de múltiples sitios

Para sistema web se recomienda especialmente realizar mínimo el siguiente grupo de pruebas de sistema:

- Humo
- Usabilidad
- Performance
- Funcionalidad

F. Métricas de Validación

Métrica de Validación ISO-9126 239 ISO-9126 es un estándar internacional para la evaluación de la calidad del Software. Está reemplazado por el proyecto SQuare, ISO-25000:2005, el cual sigue los mismos conceptos. Este estándar es el más usado. El estándar está dividido en cuatro partes las cuales dirigen, realidad, métricas externas, métricas internas y calidad en las métricas de uso y expendido. El modelo de calidad establecido en la primera parte del estándar, ISO-9126, clasifica la calidad del Software en un conjunto estructurado de características y sub-características de la siguiente manera:

- Funcionalidad
- Fiabilidad
- Usabilidad
- Eficiencia
- Mantenibilidad
- Portabilidad
- Calidad en uso

G. PHP

El sistema fue desarrollado originalmente en el año 1994 por Rasmus Lerdorf como un CGI escrito en C que permitía la interpretación de un número limitado de comandos. El sistema fue denominado Personal Home Page Tools y adquirió relativo éxito gracias a que otras personas pidieron a Rasmus que les permitiese utilizar sus programas en sus propias páginas. Dada la aceptación del primer PHP su creador diseñó un sistema para procesar formularios al que le atribuyó el nombre de FI (FormInterpreter) y el conjunto de estas dos herramientas, sería la primera versión compacta del lenguaje: PHP/FI. PHP es uno de los lenguajes más populares dentro del software libre en la programación para Web, su diversidad y soporte han sido fundamentales en lograr esta popularidad. Principales características de PHP:

- Acceso a gran número de gestores de bases de datos (Adbas D, dbm, dBase, filePro, Hyperwave, Informix, Internase, LDAP, Microsoft SQL server, mSQL, MySQL, MariaDB, ODBC, Oracle, PostgreSQL, Solid y Sybase).
- Envío de correo con SMTP.
- Acceso a servidores de FTP.
- Acceso a SNMP para gestión de redes y equipos.
- Generación dinámica de gráficos y documentos PDF.
- Análisis de documentos XML.
- Generación de datos en WDDX (Intercambio Web de Datos distribuidos).

- Soporte de hilos de ejecución a partir de PHP 4

H. HTML5

El HTML no es más que una aplicación del SGML (Standard Generalized Markup Language), un sistema para definir tipos de documentos estructurados y lenguajes de marcas para representar esos mismos documentos. El término HTML se suele referir a ambas cosas, tanto al tipo de documento como al lenguaje de marcas.

El HTML, Hyper Text Markup Language es el lenguaje de marcas de texto utilizado normalmente en la WWW (World Wide Web). Fue creado en 1986 por el físico nuclear Tim Berners-Lee; el cual tomó dos herramientas preexistentes: el concepto de Hipertexto (conocido también como link o ancla) el cual permite conectar dos elementos entre sí y el SGML (Lenguaje Estándar de Marcación General) el cual sirve para colocar etiquetas o marcas en un texto que indique como debe verse.

HTML no es propiamente un lenguaje de programación como C++, Visual Basic, etc., sino un sistema de etiquetas. HTML no presenta ningún compilador, por lo tanto, algún error de sintaxis que se presente éste no lo detectará y se visualizara en la forma como éste lo entienda.

Por otro lado, el Lenguaje extensible de marcado de hipertexto (XHTML de sus siglas en inglés eXtensible Hypertext Markup

Language), es el lenguaje demarcado pensado para sustituir a HTML como estándar para las páginas Web.

I. **CSS3**

Las hojas de estilos en cascada (CSS) permiten dar a los documentos HTML una apariencia atractiva y coherente. Al vincular muchas páginas Web a la misma hoja de estilos externa. Puede definir un aspecto y un diseño coherentes para todo un sitio Web.

J. **MARIA DB**

Basado en MySQL, es la implementación libre del antiguo SGBD, es un sistema de administración de base de datos relacional (RDBMS). Se trata de un programa capaz de almacenar una enorme cantidad de datos de gran variedad y distribuirlo para cumplir las necesidades de cualquier tipo de organización, desde pequeños establecimientos comerciales a grandes empresas y organismos administrativos.

K. **Servidor Apache**

Es un servidor multiplataforma independiente, dentro del mismo en un solo paquete se encuentra: el Gestor de base de datos, servirá para crear el modelo y estructura de la base de datos. El servidor web (apache), permitirá levantar el servicio http, https, para mostrar o visualizar el sitio o aplicación web, totalmente funcional dentro de localhost, pudiéndose configurar también para funcionar dentro de toda

la intranet. El servidor FTP (filezilla), permite el levantamiento en la máquina instalada de un servidor ftp para gestión de archivos.

CAPÍTULO III

MATERIALES Y MÉTODOS

3.1 LUGAR DE ESTUDIO

Se ejecutó en la Universidad Nacional del Altiplano, Ciudad de Puno, Provincia de Puno y Región de Puno. Exactamente en el instituto de Ciencias de la Computación.

3.2 POBLACIÓN

Está constituida por información de los procesos de contrataciones en Instituciones Públicas y Privadas en el periodo Enero - Julio del presente, en la Ciudad de Puno.

3.3 MUESTRA

Para obtener la muestra se utilizó el no probabilístico a criterio para el funcionamiento del sistema, considerando los postulantes de un proceso de selección de la Universidad Nacional del Altiplano Puno 2019.

3.4 METODO DE RECOLECCION DE DATOS

Se recolectó mediante el sistema de administración y evaluación integral online para el proceso de contrataciones en instituciones públicas y privadas, en el periodo enero – julio de 2019.

3.5 METODO DE INVESTIGACION

Método de análisis de datos

Para el análisis y diseño del sistema de administración y evaluación integral online para el proceso de contrataciones en instituciones públicas y privadas su utilizo el lenguaje de modelamiento UML para ser desarrollada y modelada según los requerimientos de análisis del sistema a implementar.

Metodología

Para el desarrollo del sistema de administración y evaluación integral online para el proceso de contrataciones en instituciones públicas y privadas que permita el procesamiento de la información de una manera estructurada, y obtener resultados y reportes. se utilizó la metodología de programación extrema XP, la misma que permitirá disminuir los tiempos de desarrollo, además de otorgar cierta flexibilidad en el diseño del software, esto por ser una de sus características principales al ser una metodología ágil, iterativa y retroalimentada en su desarrollo.

3.6 MODELO Y METODOLOGÍA DE DESARROLLO DE SOFTWARE

3.6.1 Modelo de desarrollo

Se optó por la metodología XP por ser una de las metodologías ágiles y está formada por 4 fases importantes: Análisis, Diseño, Desarrollo y pruebas. Se adaptó fácilmente al sistema que se desarrolló, ya que se basa en la realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los actores, simplicidad en las soluciones

implementadas y coraje para enfrentar los cambios. Otro de los motivos es porque generó menor tiempo y coste, además es uno de los más utilizados actualmente por tal razón nos da la seguridad para utilizar este modelo de desarrollo.

3.6.2 Fase análisis:

Este es el primer paso en la metodología XP en donde los usuarios serán los actores principales para el desarrollo del sistema, por lo tanto, se realizó el ingreso de datos mediante el sistema web a los usuarios donde nos describió a grandes rasgos los puntos de interés que debemos de considerar, con esto se lograra construir la estructura del sistema para tener más facilidad en su desarrollo.

3.6.3 Fase de diseño:

Para la realización de la segunda fase de la metodología XP sugiere que los diseños deben ser simples y sencillos por tal motivo se incluirá el desarrollados Bootstrap, en donde ya se incluye clases predefinidas de CSS en el cual nos permite una simplicidad en el interfaz.

El código ya realizado se mejorará y modificará sin alterar su funcionalidad. Para poder tener un diseño que sea fácil entendible para que en el momento que sea necesario sea adaptable y nos demande menor tiempo, esfuerzo en desarrollar y optimizar su funcionamiento.

3.6.4 Fase de desarrollo

En esta fase de la metodología XP se requiere la presencia de los usuarios para poder interactuar con el desarrollador del sistema, donde el usuario tendrá que especificar más detalladamente los requerimientos para desarrollar el sistema, en donde se debe contar con la facilidad y comprensión de códigos para poder reutilizarlos y también hacer las correcciones inmediatamente.

3.6.5 Fase pruebas

Es una de las fases más importantes en donde comprobaremos el funcionamiento de los códigos que se van implementando al sistema de almacenamiento virtual. En la ejecución del sistema virtual de almacenamiento no se tuvo problemas sin embargo como todo software está dispuesto a cualquier tipo de cambio que los usuarios requieran con la finalidad de poder optimizar el mismo. Para lo cual se aplicó la ficha de evaluación ISO -9126 que nos permitió evaluar la:

- Funcionalidad
- Fiabilidad
- Usabilidad
- Eficiencia
- Mantenimiento
- Portabilidad.

CAPÍTULO IV

RESULTADOS Y DISCUSIÓN

Para el desarrollo de la implementación del sistema del software se ha desarrollado adecuadamente según la obtención de requerimiento de análisis, la metodología y de los objetivos de la investigación planteada a continuación detallamos:

4.1 ANALIZAR Y DISEÑAR E IMPLEMENTAR UN SISTEMA WEB, BASADO EN FASES DE DESARROLLO

4.1.1 Análisis

El sistema de evaluación, tenía como premisa que el trámite de postulación que se llevara a cabo con el registro manual de los interesados a través de del personal encargado de los recursos humanos y/o comités de evaluaciones, lo que ocasionaba en un momento dado aglomeraciones, y pérdida de tiempo por parte de los productores, además de un descontrol total si no se maneja bien toda esta información.

4.1.2 Análisis de viabilidad

Viabilidad Operativa: El sistema se encuentra en la capacidad de administrar y controlar todos los datos correspondientes a los postulantes, así como las y los operadores del sistema final y condiciones técnicas para poder coadyuvar a todos estos procedimientos aquí planteados.

Viabilidad Técnica: Las instituciones que usen nuestro sistema no requieren equipos físicos para la ejecución del software, será necesario por lo que los postulantes podrán realizarlo desde cualquier ordenador o dispositivo con conexión a internet.

4.1.3 Análisis de roles

Hay que tener en cuenta que los desarrolladores del proyecto eran dos personas por lo que los roles definidos en XP fueron ocupados por mi persona, en algunos casos por el asesor de esta investigación y/o el director de la misma.

- Programador: El código fuente fue hecho solo por el autor.
- Arquitecto de Software: de igual forma el modelado de la Plataforma fue realizado por el autor.
- Cliente: El desarrollador del proyecto y los encargados de administrar la plataforma fueron los que hicieron las pruebas funcionales para validar su implementación.
- Encargado de pruebas (Tester): Se ejecutaron las pruebas regularmente, e informó los resultados y apreciaciones al desarrollador.
- Encargado de seguimientos (Tracker): El personal del de investigación quien proporcionó realimentación al equipo y realizó el seguimiento del progreso de cada iteración.
- Capacitador: El autor de la tesis fue el responsable del proceso global también de realizar las guías para las prácticas XP

además responsable de que se siguiera el proceso correctamente.

Algunos roles definidos en la metodología de desarrollo ágil XP no fueron tomados en cuenta para este proyecto, bien porque no fuese necesario o porque el tamaño tan reducido del equipo de trabajo no lo permita.

4.2 ACCESO Y VALIDACIÓN DEL SISTEMA

Para la obtención de datos personales una sola vez y su posterior reutilización se implementa un repositorio de datos locales que son invocados mediante el número DNI se anexa el ID único para las tablas de productores y se mantiene una sola entidad para los datos personales de modo que servirán para escalar a otros sistemas de información futuros.

Implementar un buscador y listado detallado de los postulantes de la institución en proceso de evaluación de personal.



Figura 1: Ingreso y referenciación de datos del Usuario

Fuente: Elaboración propia.

- El buscador está basado en el ingreso del número de DNI, una vez presionado el sistema en su configuración obtiene los datos personales, así como los datos de la última campaña de producción previamente registrados en nuestra base de datos, de modo que se ingresa una sola vez y se reusa la información de forma óptima.

4.3 PROCESOS DE LA METODOLOGÍA CON MODELAMIENTO UML

4.3.1 Diseño del sistema Web

Para el diseño de la aplicación el equipo de trabajo siguió las recomendaciones de la metodología XP, tratando de evitar las soluciones complejas, y se trabajó en varias iteraciones, previendo las modificaciones futuras para la optimización del sistema.

Otro aspecto importante en el diseño, fue la constante reestructuración del código. El principal objetivo de la reestructuración fue evitar la duplicación del código poniendo en práctica la normalización de datos simplificarlo y hacerlo más flexible para facilitar los posteriores cambios. Esto se hizo constantemente en la programación de la aplicación.

4.3.2 Modelamiento del sistema Web

Para entender el modelamiento en el diseño del sistema se ha utilizado el lenguaje unificado de Modelado UML, la misma que nos ha permitido obtener los siguientes diagramas:

- Diagrama de Casos de uso.

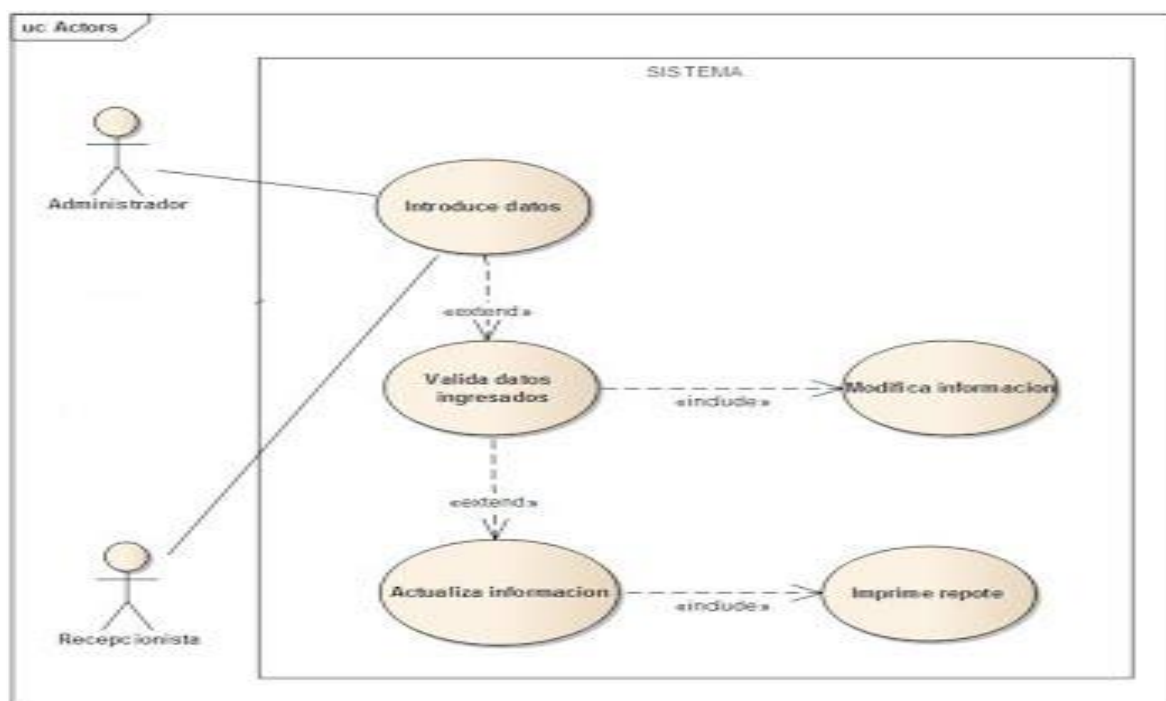


Figura 2: Diagrama de caso de uso general del sistema

Fuente: Elaboración propia.

- Diagrama de secuencia.

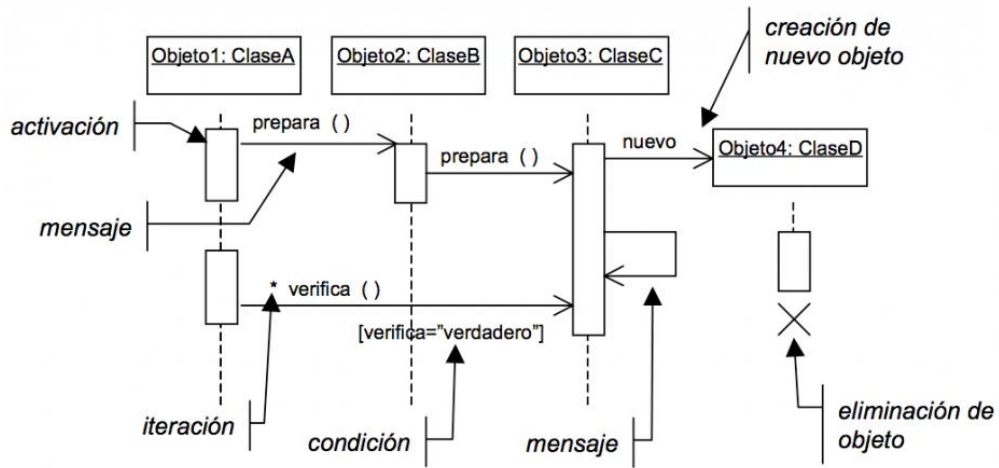


Figura 3: Diagrama de secuencia del sistema

Fuente: Elaboración propia.

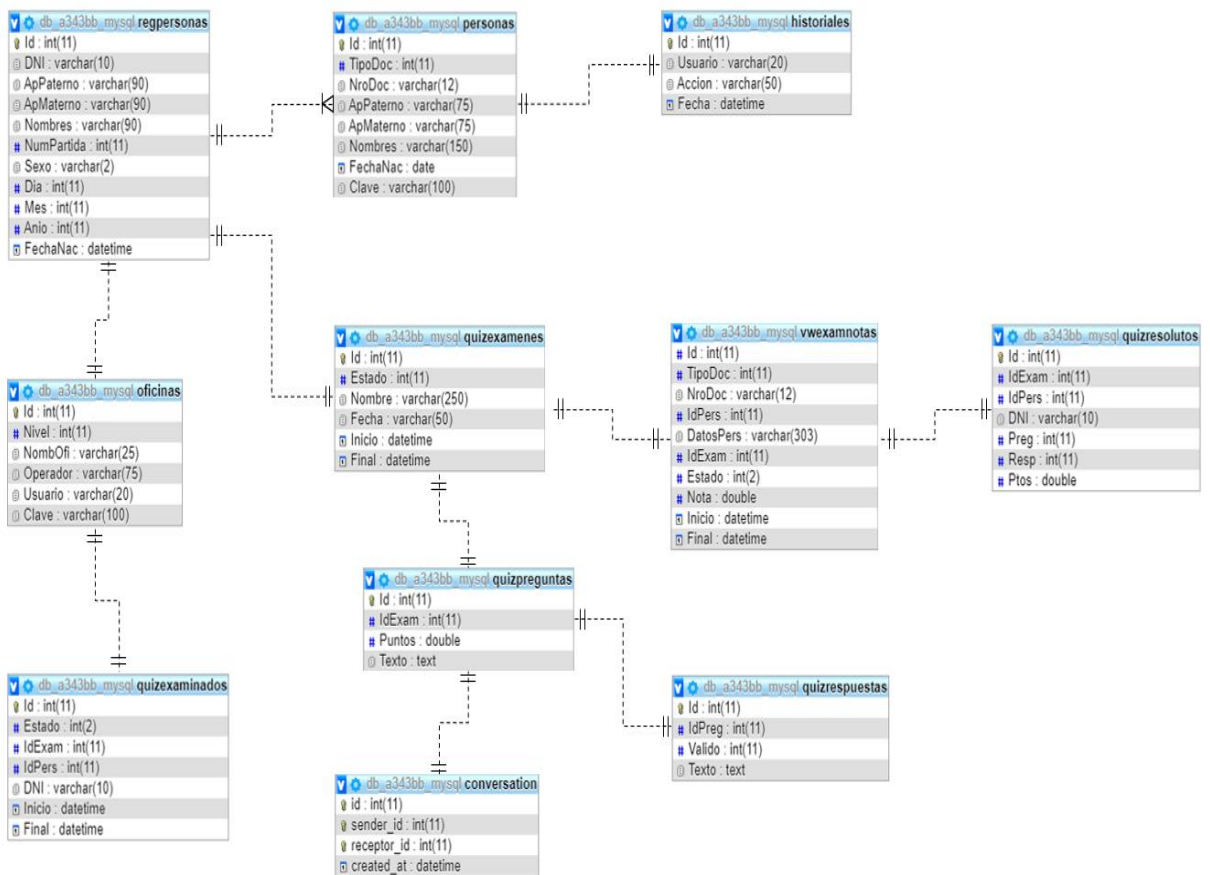


Figura 4: Diagrama de base de datos

Fuente: Elaboración propia.

4.3.3 Arquitectura de desarrollo

La arquitectura de software para nuestro sistema de información empleada ha sido la arquitectura MVC Modelo Vista Controlador, la cual se basa en una forma organizada de llevar tres entidades que conglomeran el cuerpo del sistema:

- Primero el modelo en que se implementan la capa de comunicación con la base de datos la generación de Queries, consultas a Views, la elección de los procedimientos almacenados
- Segundo la vista en la que los formularios son llenados y desplegados para mostrar información, así como la intercomunicación visual entre datos y el usuario final. La vista renderiza el contenido un pasando de contenido HTML y datos a un contenido legible y de fácil comprensión visual.
- Tercero el controlador que es la clase que une la capa de modelo de datos y la vista de datos, además el controlador interpela como la capa de negocios es decir la que administra las acciones, validaciones, comprobaciones, edición, eliminación y en suma las acciones del sistema, haciendo uso de los componentes antes mencionados.

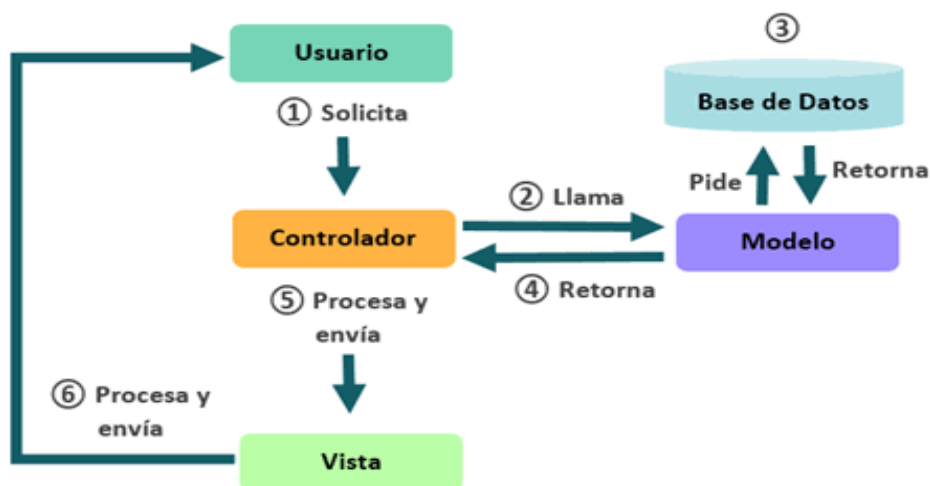


Figura 5: Esquema modelo vista controlador y la interacción entre sus componentes.

Fuente: Elaboración propia.

Para el caso particular de nuestro software las entidades descritas se muestran como sigue:

```

1 <?php if( !defined('LX_COREDIR') ) exit('LxCore is not present');
2 /*****
3 *
4 *   Sistema de evaluacion electronica
5 *
6 *****/
7
8
9 class Home extends LxApp
10 {
11     public function __construct()
12     {
13         parent::__construct();
14
15         $this->loadModel( "ado" );
16         $this->loadLibrary( "email" );
17         $this->loadLibrary( "session" );
18         $this->loadLibrary( "apilib" );
19     }
20
21     public function Index()
22     {
23         $sess = $this->session->GetData();
24
25         $this->loadView( "header" );
26         if( $sess )
27             $this->loadView( "examens", array(
28                 'sess' => $sess,
29                 'exam' => $this->ado->getTable( "quizExamenes", "Estado = 1" )
30             ) );
31         else
32             $this->loadView( "ingreso" );
33     }
34

```

Figura 6: Controlador del Software

Fuente: Elaboración propia.

```

1 <?php if( !defined('LX_COREDIR') ) exit('LxCORE is not present');
2
3 require_once ( LX_COREDIR."/db.php" );
4
5
6 class Ado extends db {
7
8     public function __construct()
9     {
10         parent::__construct();
11     }
12
13     public function inDatosPers( $Id )
14     {
15         $row = $this->getSnapRow( "Personas", "Id=$Id" );
16         if( !$row ) return null;
17
18         return "$row->ApPaterno $row->ApMaterno, $row->Nombres";
19     }
20
21     public function getQuery( $query )
22     {
23         $conn = $this->getConnection();
24         if( !$conn ) return null;
25
26         $result = $conn->query( $query );
27         if( !$result )
28             echo ">> LxDBi Error: ".mysqli_error($conn);
29
30         return $result;
31     }

```

Figura 7: Modelo del Software

Fuente: Elaboración propia.

```

1 <div class="row">
2     <div class="col-md-3 pull-right">
3         <div class="panel panel-primary">
4             <div class="panel-heading"> <b>Temporizador</b> </div>
5             <div class="panel-body">
6                 <h2> 00:00:00 </h2>
7                 <div id="panem" style="font-size: 10px"></div>
8                 <hr>
9                 <a href="." class="btn btn-warning btn-sm btn-block"> <i class="fa fa-user"></i> Me Rindo </a>
10            </div>
11        </div>
12    </div>
13    <!-- ----- -->
14    <form name="crap">
15        <div class="col-md-9">
16            <div class="panel panel-success">
17                <div class="panel-heading" id="dexc"> Listado de Preguntas </div>
18                <div class="panel-body" id="dexam" style="min-height: 200px">
19                    <?PHP
20                        echo "<script>";
21                        echo "seekQst('*')";
22                        echo "</script>";
23                    ?>
24                </div>

```

Figura 8: Vista del Software

Fuente: Elaboración propia.

4.3.4 Versiones e iteraciones

Versión 0.1

Iteración 1: El diseño de la base de datos es la parte primordial para empezar con el desarrollo de cualquier aplicación, ya que un buen diseño de esta, es la base fundamental para el óptimo funcionamiento y éxito de cualquier software.

Para definir el diseño se contó siempre con la ayuda del personal del sistema de inscripción de postulantes, quienes informaban que datos serían relevantes para el proyecto en varias iteraciones se logró diseñar las siguientes tablas.

- Tabla POSTULANTES: Esta tabla almacena a todos los postulantes del sistema.
- Tabla EXAMENES: Esta tabla almacena el contenido de los cuestionarios, así como la ponderación por cada respuesta.

De esta manera se terminó la versión 0.1, cumpliendo con las tareas registradas en el backlog.

- Versión 0.2

Iteración 1: Como la aplicación debía estar en la nube para poder ser consultada desde cualquier ubicación geográfica con conexión a internet, se seleccionó PHP 5.2 una versión estable, ya que es uno de los lenguajes de desarrollo para aplicaciones web más utilizadas y de gran difusión, por

estas razones y por otras que nombramos en el apartado de herramientas, se consideró una herramienta ideal para el desarrollo de esta aplicación.

Iteración 2: Como se buscaba tener un diseño amigable, intuitivo y de fácil uso para la aplicación se creó una página principal (máster page), la cual es una plantilla de la que heredaran toda las demás páginas de la aplicación web. De esta forma se tiene una estructura fija para todo el sitio web.

En la página principal plantilla máster se definió la estructura común para toda la aplicación web dándole los estilos y distribución correspondiente con BOOTSTRAP, en cuanto su optimización y agilidad con AJAX, herramientas de jQUERY y su validación respectiva de los formularios con JavaScript y el menú de navegación de la aplicación.

De esta manera se terminó la versión 0.2, cumpliendo con el 1er backlog del usuario.

- Versión 0.3

Iteración 1: Se implementó los módulos de registro de datos haciendo uso del MVC (Modelo Vista Controlador) para poder tener un código ordenado y óptimo.

En este módulo se hizo la aplicación vía web para el registro y validación de información de los productores empadronados, así como también se elaboró un manual de usuario para el registro de datos.

Esta versión se brindó una capacitación a los usuarios, los cuales dieron su visto bueno al sentirse conformes con los avances presentados.

- Versión 0.4.

Iteración 1: En esta versión se implementó el módulo para la edición de datos y reportes de todo el proceso que se realizó.

Este módulo fue muy similar al módulo de ingresar datos, porque por medio de los controles que proporciona las herramientas web para poder editar los datos que existían por el error que se dio por parte de los productores.

Iteración 3: En esta iteración se implementó el módulo de revisión de exámenes resueltos, por tener una similitud en el procedimiento de revisión y las correcciones se implementarán de forma similar.

Iteración 4: Para esta iteración se implementó las consultas a la base de datos de los estados, así como los datos de los postulantes.

Iteración 5: Para esta iteración se implementó los reportes de postulantes, así como de los usuarios del sistema, así como también de los estados de los mismos.

De esta manera se terminó la versión 0.4, en la que se cumple con los requerimientos del backlog del usuario, estando el sistema listo para las retrospectivas de los sprints, para realizar las mejoras correspondientes al sistema.

Esta versión se le enseñó a los clientes, los cuales dieron su visto bueno al sentirse conformes con los avances presentados además de la puesta en funcionamiento del sistema.

- Versión 0.5 hasta la 0.9

Desde la versión 0.5 hasta la versión 0.9 se trabajó en el diseño estético de la aplicación.

En cada una de las iteraciones que involucraba la entrada de datos por parte de los postulantes, se diseñaron e implementaron los diferentes tipos de validaciones respectivas para cada uno de los campos que lo requerían.

En este punto fue primordial validar los campos requeridos, los cuales no debían quedar vacíos ya que representarían un error al momento de ser guardados en la base de datos y emitir su constancia de inscripción y su respectivo carné.

Además de estar listos para que pueda ser instalado de una manera fácil y poder ser controlado desde cualquier sitio remotamente por el administrador del sistema.

4.3.5 Codificación

Toda la transcripción de algoritmos y métodos utilizados para el software, se han realizado en PHP, esto hace al software independiente del sistema que se esté utilizando, sin exigir las características de hardware, únicamente un navegador web (browser) porque lo más importante para el software es el cumplimiento de sus funciones para las cuales fue desarrollado, así como para la administración del mismo.

Cliente siempre disponible

XP recomienda como factor de éxito que el cliente esté involucrado en toda la etapa de desarrollo, esto se cumplió satisfactoriamente.

Estándares de codificación

La metodología XP aconseja seguir estándares de codificación para que cualquier integrante del equipo de desarrollo, pueda entender y asimilar fácilmente el código fuente del programa por otro integrante.

Seguir esta recomendación fue fácil para el desarrollador ya que venía trabajando de tiempo atrás en otros desarrollos y esta buena práctica siempre se tuvo presente.

4.3.6 Pruebas

Al momento de las ejecuciones reiteradas del sistema no se tuvo inconvenientes sin embargo no está demás explicar que como todo software este se encuentra predispuesto a cualquier tipo de cambio que el usuario requiera con la finalidad de poder optimizar el mismo, esto incluye también las actualizaciones que por tiempo de vida de uso se deben de realizar periódicamente.

4.3.7 Instalación

Durante la instalación no se presentó ningún tipo de problemas, porque este sistema fue desarrollado en un entorno web.

Lo que si debemos de tener en cuenta de que el tráfico de información el cual aumentará progresivamente por la cantidad de postulantes que se registren en el sistema, también el manual de usuario se encuentra en el ANEXO A.

4.3.8 Resultados de la prueba según ISO-9126

Promedio general de las fichas de evaluación ISO-9126.

Tabla 1 : Cuadro de decisiones ISO - 9126

Clasificación	Intervalo	Decisión
A) Inaceptable	[27 - 54 >	
B) Mínimamente aceptable	[54 – 81 >	
C) Aceptable	[81 – 95 >	
D) Cumple los requisitos	[95 – 122 >	102
E) Excede los requisitos	[122 – 135]	

Fuente: Cuadro de decisiones ISO - 9126

Según los resultados el promedio de 103, 102, 108, 100, 98, 107 y 99 nos resulta 102, indicando que cumple con los requisitos según el ISO - 9126. Todo esto se muestra en el ANEXO B.

DECISIÓN: De acuerdo a los resultados de la calidad del software se concluyó que el sistema web para la evaluación rápida online, cumple los requisitos con un promedio de 110 puntos del total de 135 puntos que se considera en el cuadro de decisiones del ISO - 9126.7.

4.3.9 Cuadro comparativo según el tiempo resultados utilizando el sistema y sin el sistema

Tabla 2: Cuadro comparativo expresado en minutos y horas con y sin Sistema de Evaluación.

	CON	SIN
Registro de Postulante	1 min.	2 horas.
Validación de Postulante	2 min.	20 min.
Evaluación de Postulante	20 min.	3 días.
Publicación de Resultado	1 min.	1 día.
Información de Concurso	1 min	2 días
TOTAL	25 min.	6 días, 2 horas y 20 min.

Fuente propia: Cuadro comparativo expresado en minutos con y sin el sistema.

Decisión:

De acuerdo al cuadro comparativo expresado en minutos de una inscripción con y sin el sistema, se aprecia que utilizándolo hay una demora de 25 min. Promedio por persona mientras que sin el sistema la diferencia es amplia.

4.3.10 Resultados de satisfacción de usuario

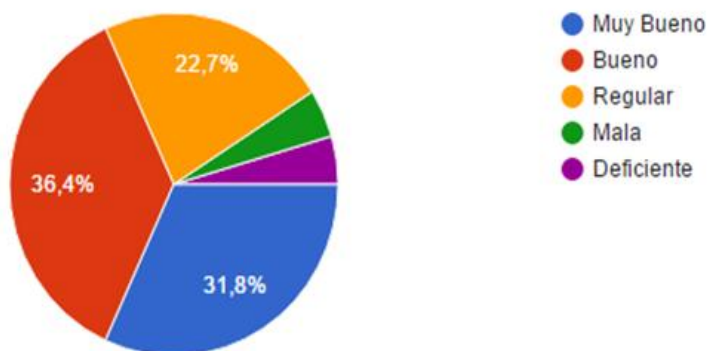


Figura 9: Considera Ud. Que el Sistema de Evaluación es amigable

Fuente: Entrevista Directa

Interpretación: Un 36,4% considera que tiene una interfaz amigable

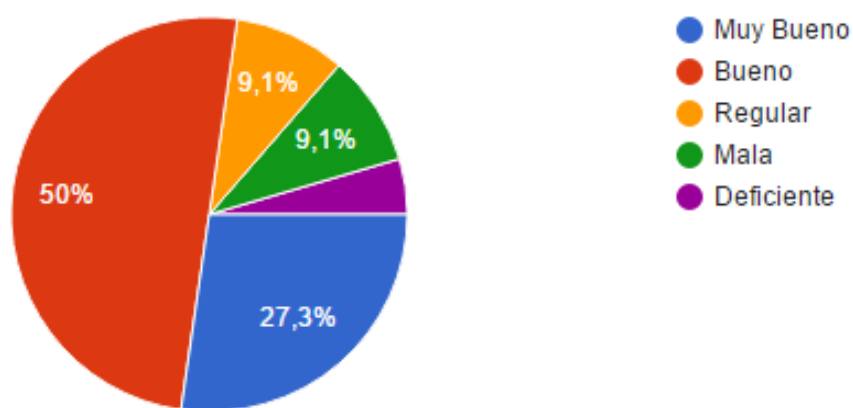


Figura 10: Considera que el Sistema tiene facilidad De uso

Fuente: Entrevista Directa

Interpretación: Un 50% considera que el sistema tiene facilidad de uso.

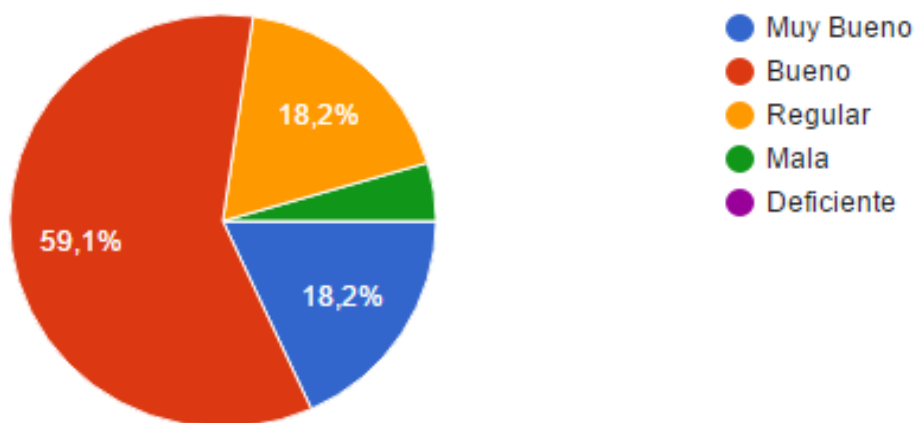


Figura 11: Número de Usuarios según La Seguridad del Sistema

Fuente: Entrevista Directa

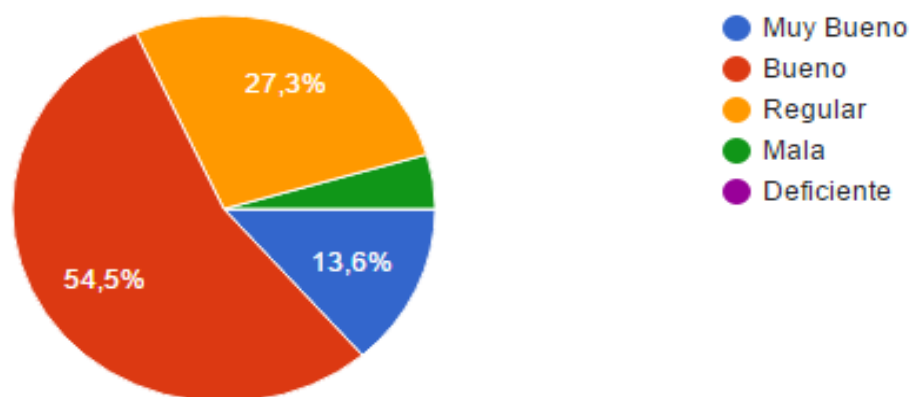


Figura 12: Número de Usuarios según la Claridad de la Información

Fuente: Entrevista Directa

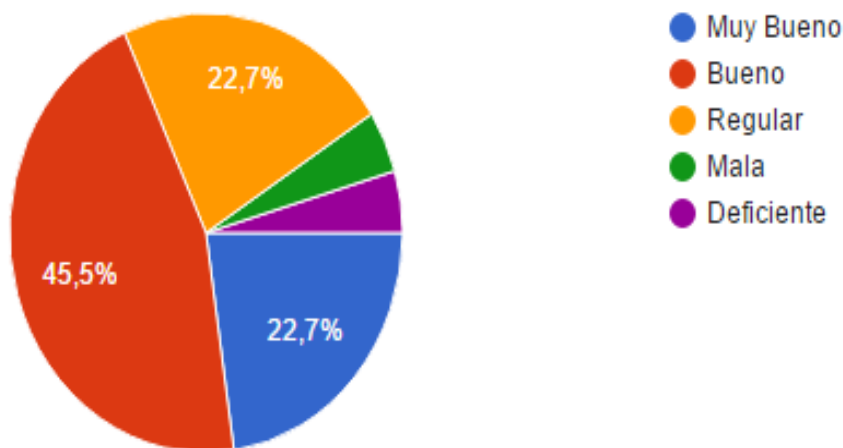


Figura 13: Número de Usuarios según el Tiempo de Respuesta del Sistema

Fuente: Entrevista Directa

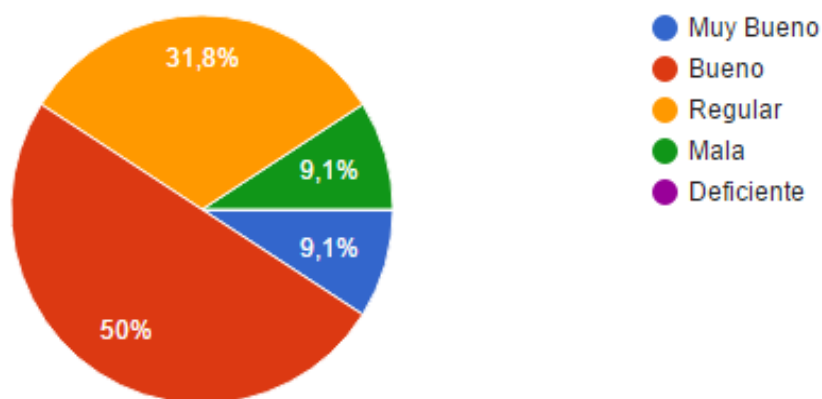


Figura 14: Número de Usuarios según la Calidad de Soporte

Fuente: Entrevista Directa

Conclusión de la encuesta.

De acuerdo a los resultados obtenidos, se puede concluir que la implementación del sistema web contribuye en el proceso de inscripción, registro, mantenimiento y seguimiento de los postulantes y control de resultados para los procesos de calificación.

CONCLUSIONES

PRIMERO. Se concluye con el análisis de los casos de uso del sistema web, también basándonos en una interfaz para su adecuada navegación sobre el sistema, cabe destacar que se ha empleado un framework bastante estable para la adaptabilidad de la interfaz sobre los distintos dispositivos, internamente administra las marcas CSS3 que permiten modificar estilos en base a las dimensiones condicionales sobre la hoja de estilos CSS (Cascade Style Shet).

SEGUNDO. Para el ingreso de información a la base de datos se realiza mediante el sistema y los postulantes no deben estar presentes físicamente durante la inscripción, para luego verificar la validez de los datos ingresados, en caso de ser ganadores.

TERCERO. La interfaz ofrece un diseño moderno, ergonómico y de fácil acceso para usuarios no expertos en el uso de computadores personales, con una capacitación de un tiempo mínimo pueden operar los elementos del software. Aplicando la menor cantidad de controles visuales para poder desplegar e ingresar la información de los postulantes, como el módulo de administración muestra directamente los resultados de evaluación, se agregar en la capa de visualización además una serie de sub opciones para que evite movimientos largos del puntero del mouse.

RECOMENDACIONES

PRIMERO. Se recomienda agregar el uso de Mapas por API REST para mejorar la geo localización en la interfaz del sistema, por tanto es posible aun implementar y obtener una aplicación mejorada basada en lo elementos ya existentes.

SEGUNDO. Se recomienda la implementación sobre una base de datos PosgreSQL o SQL Server para ampliar la fiabilidad de datos.

TERCERO. Se recomienda seguir mejorando la interfaz con probando la efectividad de otros Frameworks como VueJS para su funcionalidad entre el controlador y la vista en la renderización de contenido HTML.

CUARTO. Se recomienda el uso de software auditor para la evaluación de infiltraciones durante el desarrollo de aplicaciones web, en especial las que estarán disponibles para una gran cantidad de usuarios.

REFERENCIAS BIBLIOGRÁFICAS

- Arcaya William, (2000). Prototipo de Aplicación Cliente para Consulta y Actualización de Datos Académicos Mediante Internet. Tesis de Maestría, UNA Puno.
- Borjas, G. (2013). “Análisis, diseño e implementación de un sistema de información para la administración de horarios y rutas en empresas de transporte público” (p. 02).
- Chachaque, L. (2015). “Sistema web para la administración académica del instituto superior de educación pública “Divino Jesús” chumbivilcas 2014” (p. 01).
- Coronel Castillo, Gustavo (2005). Creando soluciones web con php 4.5 y 5.x y MySQL Server para versiones 3.2.3, 4.x y 5.x.. Editorial Empresa Editora Macro EIRL., Primera Edición. . Lima. Perú (2005).
- Coad Peter Y Yourdon E. (1990). Object Oriented Analysis. Prentice Hall.
- Crovetto Huerta Christian, (2005). Oracle 10g: Base de datos. . Editorial Megabyte. Primera Edición. . Lima. Perú.
- Guzmán Jiménez Rosario (2001). Base de datos relacionales: Un enfoque práctico. Editorial Fondo de Desarrollo., Primera Edición. Santiago. Chile.
- Jiménez M., José A. (2009). Matemáticas para la Computación. Alfaomega Grupo Editor, Pitagoras 1139 Col. Del Valle. México Diciembre de 2008.
- Joseph Vilalta,(2001). UML Guía Visual – Cómo Crear Formas de Vida Organizativa –Consultores 2001.

- Landis william e. And chandler robin I,(2007). Archives and the Digital Library.
- Liliana Favre Irm Press (2003) UMLand the Unified Process, 2003.
- López, A. I. (2015). “Diseño e Implementación de una Aplicación Web para la administración de proyectos de Investigación, Innovación y desarrollo” (p.01).
- McDonald, Mattew (2009). Creación y Diseño Web. O’Reilly Media, Inc. Edicion en Español, Ediciones ANAYA Multimedia, Fernandez Ciudad, S.L. Madrid España.
- Mcgarry John (2001). Practical Software Measurement. Addisson Wesley.
- Rojas G., Octavio (2008). Clasificación Decimal DEWEY Abreviada e Índice Relativo, (Desarrollada por Melvil Dewey). Panamericana Technology Group. PTG. Rojas Eberhard Editores Ltda. Bogotá, D.C. 2008.
- Pacho, M. A. (2009). “Desarrollo de Un Sistema De Gestión Documentario En Entorno Web para la Escuela de Post Grado de la UNA”. PERÚ: PUNO.
- Pressman Roger (2002). Ingeniería del Software, Un enfoque práctico. Tercera edición. McGraw Hill Cap 12.
- Otoniel Perez G., (2003) Métricas, Estimación y Planificación en Proyectos de Software.
- Torres, P. I. & Sánchez, L. E. A. (2003). “Metodologías Ágiles en el Desarrollo de Software” (p. 01).
- Schmuller Joseph, (2003) Aprendiendo UML en 24 Horas, Editorial Prentice Hall.

Sommerville, I. (2005). Ingeniería del software. Pearson educación.

Upton, D. (2015). CodeIgniter for Rapid PHP Application Development.

Packt Publishing Ltd.

ANEXOS

ANEXO 1: MANUAL DEL SISTEMA

A. MÓDULO DE INGRESO AL SISTEMA



MÓDULO DE MANTENIMIENTO DE DATOS

PANEL DE CONTROL



MÓDULO DE COMPONENTES Y ACTIVIDADES

PANEL DE CONTROL

OPCIONES

- [Puntuaciones](#)
- [Ingreso de Exámenes](#)

Listado de Exámenes

+ Agregar
Regresar

Nro	Pregunta	Respuestas	Ponderado
1	Según el Discovery Channel ¿Cuál es el "monte" mas alto del mundo?	ABCD	5
2	Y recuerde el alma dormida, aviva el seso y despierta contemplado. ¿Fue escrito por?	ABCD	5
3	¿Quien descubrió el escondite del Conde de Montecristo?	AB	10
4	Entonces Julio pregunto: ¿Cuanto te faltará Fred?. Lo que llevo a la respuesta inmediata y algo alocada por parte del agraviado, entonces: ¿cuanto tiempo era el faltante para Fred?	AB	10

© 2018 : Desarrollado con Lightning-X 1/16 MVC Framework

ANEXO 2: EVALUACION DEL SISTEMA

SISTEMA DE ADMINISTRACIÓN Y EVALUACIÓN INTEGRAL ONLINE PARA EL PROCESO DE CONTRATACIONES EN INSTITUCIONES PÚBLICAS Y PRIVADAS

INDICACIONES: Al responder cada uno de los ítems marcará con una “X” solo una de las alternativas propuestas.

INDICADORES	PUNTUACION				
	MUY BUENO	BUENO	REGULAR	MALA	DEFICIENTE
1. ¿cómo considera el interfaz del sistema de almacenamiento de datos?					
2. ¿cómo considera usted la navegación en el sistema de almacenamiento de datos?(Velocidad de Carga)					
3. ¿cómo considera usted el acceso a sistema de almacenamiento (login)?					
4. ¿cómo considera usted la seguridad de su información en el sistema de almacenamiento de datos?					
5. ¿El sistema de almacenamiento de datos muestra su información con claridad? *(Sus archivos y documentos son fácil de ubicar)					
6. ¿Cómo considera usted el sistema virtual de almacenamiento de datos en los procesos de almacenar y compartir archivos?					
7. ¿Los mensajes de error son adecuados y entendibles?					
8. ¿cómo considera la organización de los menús?					

FICHA DE EVALUACIÓN DE LA CALIDAD DEL PRODUCTO SOFTWARE

NORMA ISO 9126

INDICADORES	PUNTUACION				
	1	2	3	4	5
1. FUNCIONALIDAD					
Adecuación: Capacidad del producto software para proporcionar un conjunto apropiado de funciones para tareas específicas y objetivos de los usuarios.					
Exactitud: Capacidad del producto software para proporcionar los resultados o efectos correctos y con el grado de precisión acordado.					
Interoperatividad: Capacidad del producto software para interactuar con uno o más sistemas especificados.					
Seguridad: Capacidad del producto software para proteger la información y los datos.					
Conformidad: Capacidad del producto software para adaptarse a los estándares, convenciones o regulaciones en leyes y prescripciones relativos a la funcionalidad.					
2. FIABILIDAD					
Madurez: Capacidad del producto software para evitar fallos provocados por errores en el software.					
Tolerancia a fallos: Capacidad del producto software para mantener un nivel de rendimiento determinado en caso de defectos en el software o incumplimiento de su interfaz.					
Recuperabilidad: Capacidad del producto software para restablecer un nivel de rendimiento y recuperar los datos afectados directamente en caso de ocurrir un fallo.					
Conformidad: Capacidad del producto software para adaptarse a los estándares, convenciones o regulaciones a la fiabilidad.					
3. USABILIDAD					
Comprensibilidad: 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.					
Facilidad de aprendizaje: Capacidad del producto software para permitir al usuario aprender su aplicación.					
Operabilidad: Capacidad del producto software para permitir que el usuario lo opere y lo controle.					

Atracción: Capacidad del producto software para atraer al usuario.					
Conformidad: Capacidad del producto software para adaptarse a los estándares, convenciones, guías de estilo y regulaciones con la usabilidad.					
4. EFICIENCIA					
Tiempo de respuesta: Capacidad del producto software para proporcionar tiempos de respuesta y de procesamiento apropiados cuando realiza sus funciones bajo condiciones determinadas.					
Utilización de recursos: Capacidad del producto software para utilizar cantidades y tipos de recursos apropiados cuando el software realiza su función bajo determinadas condiciones.					
Conformidad: Capacidad del producto software para adaptarse a los estándares, convenciones relacionadas con la eficiencia.					
5. MANTENIBILIDAD					
Analizabilidad: Capacidad del producto software de diagnosticar sus deficiencias o causas de fallos, o de identificar las partes que deben ser modificadas.					
Cambiabilidad: Capacidad del producto software de permitir implementar una modificación especificada. La implementación incluye los cambios en el diseño, el código y la documentación.					
Estabilidad: Capacidad del producto software de evitar los efectos inesperados de las modificaciones.					
Facilidad de prueba: Capacidad del producto software de permitir validar las partes modificadas.					
Conformidad: Capacidad del producto software de cumplir los estándares o convenciones relativas a la mantenibilidad.					
6. PORTABILIDAD					
Adaptabilidad: 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.					
Facilidad de instalación: Capacidad del producto software para ser instalado en un ambiente determinado.					
Coexistencia: Capacidad del producto software para coexistir con otro software independiente en un ambiente común compartiendo recursos.					



Reemplazabilidad: Capacidad del producto software para ser utilizado en lugar de otro producto de software para el mismo propósito en el mismo ambiente.					
Conformidad: Capacidad del producto software de cumplir los estándares relacionados con la portabilidad.					
SUB TOTALES					
PUNTAJE TOTAL					

ANEXO 3: CÓDIGO FUENTE DEL SISTEMA

A. LOGIN

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title> SIDE REC+ - 2019 </title>
    <meta name="description" content="">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="robots" content="all, follow">
    <link href="<?=base_url()?>includes/img/favicon.jpg" rel="icon" sizes="32x32" />
    <link href="<?=base_url()?>includes/img/favicon.jpg" rel="icon" sizes="192x192" />

    <link rel="stylesheet"
href="<?=base_url("includes/adm/css/bootstrap.min.css")?>">
    <link rel="stylesheet"
href="https://fonts.googleapis.com/css?family=Poppins:300,400,700">
    <link rel="stylesheet" href="<?=base_url("includes/adm/font-awesome/css/font-awesome.css");?>">
    <link rel="stylesheet" href="<?=base_url("includes/adm/css/fontastic.css");?>">
    <link rel="stylesheet" href="<?=base_url("includes/adm/css/style.green.css");?>"
id="theme-stylesheet">
    <link rel="stylesheet" href="<?=base_url("includes/adm/css/custom.css");?>">
    <!-- Favicon-->
    <!-- <link rel="shortcut icon" href="img/favicon.ico"> -->
    <!-- Tweaks for older IEs--><!--[if lt IE 9]>
      <script
src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
      <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
    <![endif]-->
  </head>

  <body>
    <div class="page login-page tinted-back">
      <div class="container d-flex align-items-center">
        <div class="form-holder">
          <div class="row">
            <div class="col-lg-6 imgblur1">
              <div class="info d-flex align-items-center">
                <div class="content">
                  <div class="logo" style="font-size: 4em; text-align: center">
                    <img class="img-responsive" src="">
                    <span>SIDE</span><b>REC</b>+
                  </div>
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </body>
</html>

```

```

<br>
<!--<h4>PROYECTO:</h4>!-->
<p class="text-right">Sistema Descentralizado de Identificación y
Registro Civil</p>
<br><br><br><br><br>
<p class="text-gray-dark">
versión: 0.702b<br>
Junio de 2019<br>
</p>
</div>
</div>
<!-- Form Panel -->
<div class="col-lg-6 bg-white">
<div class="form d-flex align-items-center">
<div class="content">
<form method="post" class="form-validate"
action="<?==base_url('admin/login');?>">
<div class="form-group">
<label for="login-username" class="form-control-label">USUARIO
:</label>
<input id="login-user" type="text" name="user" required
placeholder="nombre de usuario" class="input-material" autofocus
autocomplete="off">
</div>
<div class="form-group">
<label for="login-password" class="form-control-
label">CONTRASEÑA :</label>
<input id="login-pass" type="password" name="pass" required
placeholder="ingrese contraseña" class="input-material">
</div>
<div class="form-row">
<div class="form-group col-md-6" style="padding: 4px">
<button id="login" type='submit' class="form-control btn btn-
primary"> Validar mis datos </button>
</div>
<div class="form-group col-md-6" style="padding: 4px">
<!--
<button type="button" class="form-control btn btn-default"
onclick="location.href='<?==base_url()?>'"> Ir a la Pagina Web </button>!-->
</div>
</div>
</form>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
<div class="copyrights text-center">
<p class="text-success"> &copy; MicroLogic Software - 2019 </p>
</div>

```

```

</div>

<!-- JavaScript files-->
<!-- <script src="https://code.jquery.com/jquery-3.3.1.min.js"></script> -->
<script src="<?=base_url("includes/adm/js/jquery.min.js"); ?>"> </script>
<script src="<?=base_url("includes/adm/js/bootstrap.min.js"); ?>"> </script>
<!-- Main File-->
</body>
</html>

```

B. PANEL DE CONTROL

```

<form id="frmPar" name=frmPar onsubmit="return
loadWebFrm('dvDisp','admin/parGraba',this)" method="post">

  <div class="form-row">
    <div class="form-group col-md-2">
      <input name="ldni" value="<?=$arg["DNI"]?>" type="text" class="form-
control" placeholder="Ingrese num de DNI" autofocus>
    </div>
    <div class="form-group col-md-1">
      <button class="form-control"
onclick="loadWebFrm('dvDisp','admin/parNuevo',frmPar)"> Buscar </button>
    </div>
    <div class="form-group col-md-9"></div>

    <div class="form-group col-md-3">
      <label for="">Nombres</label>
      <input name="noms" value="<?=$arg["Nombres"]?>" type="text"
class="form-control" readonly required>
    </div>
    <div class="form-group col-md-3">
      <label for="">Apell. Paterno</label>
      <input name="pate" value="<?=$arg["ApPaterno"]?>" type="text"
class="form-control" readonly>
    </div>
    <div class="form-group col-md-3">
      <label for="">Apell. Materno</label>
      <input name="mate" value="<?=$arg["ApMaterno"]?>" type="text"
class="form-control" readonly>
    </div>
    <div class="form-group col-md-3">
      <label for="">Fecha de Nacim.</label>
      <input name="fecn" value="<?=$arg["FechaNac"]?>" type="text"
class="form-control" readonly>
      <input name="sexo" value="<?=$arg["Sexo"]?>" type="hidden">
    </div>

```

```

<div class="form-group col-md-1">
  <label for="">Num Partida</label>
  <input name="pnum" value="" type="text" class="form-control"
placeholder="0000000" required>
</div>
<div class="form-group col-md-2">
  <label for="">Fecha de Registro</label>
  <input name="anio" value="" type="text" class="form-control"
placeholder="dd/mm/aaaa">
</div>
<div class="form-group col-md-2">
  <label for="">Hora de Nacim.</label>
  <input name="hora" value="" type="text" class="form-control"
placeholder="hh:mm am/pm">
</div>
<div class="form-group col-md-4">
  <label for="">Centro Poblado / Comunidad / Cacerio</label>
  <input name="luga" value="" type="text" class="form-control" placeholder=""
required>
</div>
<div class="form-group col-md-3">
  <label for="">Registrador</label>
  <input name="regi" value="" type="text" class="form-control"
placeholder="Nombres y apellidos">
</div>

<div class="form-group col-md-3">
  <label for="">Nombre de Padre</label>
  <input name="pad1" value="" type="text" class="form-control"
placeholder="Nombres y apellidos" required>
</div>
<div class="form-group col-md-3">
  <label for="">Nombre de Madre</label>
  <input name="pad2" value="" type="text" class="form-control"
placeholder="Nombres y apellidos" required>
</div>
<div class="form-group col-md-3">
  <label for="">Testigo 1</label>
  <input name="tes1" value="" type="text" class="form-control"
placeholder="Nombres y apellidos">
</div>
<div class="form-group col-md-3">
  <label for="">Testigo 2</label>
  <input name="tes2" value="" type="text" class="form-control"
placeholder="Nombres y apellidos">
</div>

<div class="form-group col-md-2">
  <button type="submit" class="form-control btn btn-info btn-sm"> <i
class="glyphicon glyphicon-save"></i> Grabar Partida </button>
</div>

```

```
<div class="form-group col-md-2">  
  <button type="button" class="form-control btn btn-default btn-sm"  
  onclick="$('#frmPar').find('input,textarea,select').val('').end();"> Borrar Todo  
</button>  
</div>  
</div>  
</form>
```

C. OBJETO ADMINISTRADOR

```

<?php if( !defined('LX_COREDIR') ) exit('LxCore is not');

define( "DR_ADM", "eAdmin" );

class Admin extends IxApp
{
    public function listaPDF()
    {
        $rep = $this->ado->getTable("dicPersonas", "" );

        $pdf = new SexPdf();
        $pdf->SetMargins(18, 15, 18);

        $nro = 1;
        $pdf->AddPage();

        $pdf->SetFont('Arial','',23);
        $pdf->MultiCell( 178, 9, toUTF("PRO+MEGRAN"), 0, 'C' );

        $pdf->SetFont('Arial','',14);
        $pdf->MultiCell( 178, 5, toUTF("Listado de Producers 2018"), 0, 'C' );

        $pdf->Ln(5);
        $pdf->SetFont('Arial','',9);
        foreach( $rep->result() as $row ){

            $pdf->Cell( 10, 8, $nro, 1, 0, 'C' );
            $pdf->Cell( 30, 8, toUTF($row->NUM), 1, 0, 'L' );
            $pdf->Cell( 120, 8, toUTF("$row->ApPaterno $row->ApMaterno $row-
>Nombres"), 1, 0, 'L' );
            $pdf->Cell( 2, 8, "", 0, 1);
            $nro = $nro + 1;
        }

        $pdf->SetDisplayMode('real');
        $pdf->Output("l", "lista1.pdf");
    }

    public function __construct()
    {
        parent::__construct();
        $this->loadModel( "ado" );
        $this->loadLibrary( "session" );
        $this->loadLibrary( "sexpdf" );
    }
}

```



```

        $this->loadLibrary( "GenApi" );
    }

    public function index()
    {
        $sess = $this->session->GetData(DRAP_ADM);

        if( $sess ) {
            $this->loadView( "adm/pagina", array('sess'=>$sess) );
            return;
        }

        $this->loadView( "adm/login" );
    }

    public function login()
    {
        // lectura anti XSS/SQL Inject
        $user = mISecurePost("user");
        $pass = mISecurePost("pass");
        $pass = sqlPassword( $pass );

        if( $row = $this->ado->getSnapRow("Usuarios","User='$user' AND
Pass='$pass'" ) ) {

            $this->session->SetAdminLogin (
                DRAP_ADM,
                $row->Id, // Id
                $row->User,
                $row->Nombres,
                $row->Nivel // Nivel
            );
            //echo "login:ok";

        } else {
            //echo "login:err";
        }

        redirect( base_url("admin"), 'refresh');
    }

    public function logout()
    {
        $this->session->SessionDestroy( DRAP_ADM );
        redirect( base_url("admin"), 'refresh');
    }

    //-----
    public function inPadron()

```

```

{
    $this->session->IsLoggedAccess(DRAP_ADM);
    $sess = $this->session->GetData(DRAP_ADM);

    $this->loadView("adm/inPadro" );
}

public function buscar( $dni=null )
{
    $this->session->IsLoggedAccess(DRAP_ADM);
    $sess = $this->session->GetData(DRAP_ADM);

    if( $dni ){
        $this->busqPorDni( $dni );
        return;
    }

    $ldni = mISecurePost( "ldni" );
    $apes = mISecurePost( "apes" );

    if($ldni) $this->busqPorDni( $ldni );
    else $this->busqPorApes( $apes );
}

private function busqPorApes( $apes )
{
    $tbl = $this->ado->getSnapView("dicPersonas","ApPaterno LIKE '$apes%' or
ApMaterno LIKE '$apes%'");
    $nro = $tbl->numRows();

    echo "<table class='table table-striped'>";
    foreach( $tbl->result() as $row ){

        $evn = "loadWeb('dvBsq','admin/buscar/$row->NUM')";

        echo "<tr>";
        echo "<td> <button onclick=\"\$evn\" class='btn btn-default btn-xs'>
<small>ver</small> </button> </td>";
        echo "<td> <b>$nro</b> </td>";
        echo "<td> $row->NUM </td>";
        echo "<td> <b>$row->ApPaterno $row->ApMaterno $row->Nombres</b>
</td>";
        echo "<td> $row->Sexo </td>";
        echo "<td>"; $this->ubigeos($row->ubigeoDom); echo "</td>";
        echo "</tr>";

        $nro--;
    }
    echo "</table>";
}
}

```

```

private function busqPorDni( $Idni )
{
    $domi = 0;

    if( $row=$this->ado->getSnapRow("dicPersonas","NUM=$Idni") ){

        $prod = $this->ado->getSnapRow("dicProducc","IdPers=$row->Id");

        $args = array(
            'NUM'      => $Idni,
            'ApPaterno' => $row->ApPaterno,
            'ApMaterno' => $row->ApMaterno,
            'Nombres'  => $row->Nombres,
            'Sexo'     => $row->Sexo,
            'FechaNac' => $row->FechaNac,
            'ubigeoNac' => $row->ubigeoNac,
            'ubigeoDom' => $row->ubigeoDom,
            'IdComu'   => $prod->IdComu,
            'IdCulti'  => $prod->IdCulti,
            'Area'     => $prod->Area,
            'Prod'     => $prod->Prod
        );

        $domi = $row->ubigeoDom;

    } else {

        $res = json_decode( $this->GenApi->getDataPer($Idni) );

        $args = array(
            'NUM'      => $res->NUM,
            'ApPaterno' => $res->ApPaterno,
            'ApMaterno' => $res->ApMaterno,
            'Nombres'  => $res->Nombres,
            'FechaNac' => $res->FechaNac,
            'ubigeoNac' => $res->ubigeoNac,
            'ubigeoDom' => $res->ubigeoDom,
            'Sexo'     => (($res->Sexo==2099)? "M" : "F"),
            'IdComu'   => 0
        );

        $domi = $res->ubigeoDom;
    }

    $cult = $this->ado->getTable( "dicCultivos" );
    $ubig = $this->ado->getSnapRow("dicUbigeos","Ubigeo = '$domi'");
    $comu = $this->ado->getTable( "dicComunis", "Prov='$ubig->Prov' ORDER
BY Nombre" );
    //$comu = $this->ado->getTable( "dicComunis", "Prov='$ubig->Prov' AND Dist
LIKE '%$ubig->Dist%' );

```

```

        $this->loadView( "adm/inDatos", ['arg'=>$args, 'cult'=>$cult, 'comu'=>$comu]
    );
}

public function grabar()
{
    $this->session->IsLoggedInAccess(DRAP_ADM);
    $sess = $this->session->GetData(DRAP_ADM);

    $dni = mISecurePost( "dni" );
    $pate = mISecurePost( "pate" );
    $mate = mISecurePost( "mate" );
    $noms = mISecurePost( "noms" );

    $sexo = mISecurePost( "sexo" );
    $fech = mISecurePost( "fech" );
    $naci = mISecurePost( "nacim" );
    $resi = mISecurePost( "resid" );

    $comu = mISecurePost( "comu" );
    $camp = mISecurePost( "camp" );
    $area = mISecurePost( "area" );
    $prod = mISecurePost( "prod" );
    $cult = mISecurePost( "cult" );

    // grabar cuando no exista duplicados
    //
    $per = $this->ado->getSnapRow("dicPersonas","NUM=$dni");

    if( ! $per ){
        $idper = $this->ado->Insert( "dicPersonas", array(
            'Tipo' => "01",
            'NUM' => $dni,
            'ApPaterno' => $pate,
            'ApMaterno' => $mate,
            'Nombres' => $noms,
            'FechaNac' => $fech,
            'Sexo' => $sexo,
            'ubigeoNac' => $naci,
            'ubigeoDom' => $resi,
            'FechaReg' => mICurrentDate()
        ) );

        $reg = $this->ado->Insert( "dicProducc", array(
            'IdPers' => $idper,
            'Periodo' => $camp,
            'IdComu' => $comu,
            'IdCulti' => $cult,
            'Area' => $area,
            'Prod' => $prod,

```

```

    ));

    echo "Guardado con Id($reg)";

} else {

    $pro = $this->ado->getSnapRow( "dicProducc", "Periodo=$camp AND
    IdPers=$per->Id"); ////" AND IdComu=$comu" );

    // como ya existe lo obvio es la edicion/actualizacion
    $reg = $this->ado->Update( "dicProducc", array(
        'IdComu' => $comu,
        'IdCulti' => $cult,
        'Area' => $area,
        'Prod' => $prod
    ), "Id=$pro->Id" );

    echo "Produccion ($pro->Id) editada.";
}

}

public function ubigeos( $ubi=null )
{
    $row = $this->ado->getSnapRow("dicUbigeos","Ubigeo = '$ubi'");
    if( !$row ){
        echo "Otro DEP.";
        return;
    }

    echo "<b>$row->Depa / $row->Prov / $row->Dist</b>";
}

public function inListar()
{
    $this->session->IsLoggedAccess(DRAP_ADM);
    $sess = $this->session->GetData(DRAP_ADM);

    $comu = $this->ado->getTable( "vwComuniAli" );
    $umed = $this->ado->getTable( "dicUniMed" );
    $this->loadView( "adm/inLista", ['comu'=>$comu,'umed'=>$umed] );
}

public function inListax()
{
    $this->session->IsLoggedAccess(DRAP_ADM);
    $sess = $this->session->GetData(DRAP_ADM);

    $idare = mISecurePost("area");
    $idcom = mISecurePost("comu");

```

```

$nro = 1;
$tbl = $this->ado->getTable( "dicProducc", "IdComu=$idcom" );

echo '<div style="overflow-x:auto">';
echo "<table class='table table-bordered table-striped'>";
echo "<tr>";
echo "<th> Nro </th>";
echo "<th> Zona </th>";
echo "<th> Distrito </th>";
echo "<th> Periodo </th>";
echo "<th> DNI </th>";
echo "<th> Apellidos y Nombres </th>";
echo "<th> Cultivo </th>";
echo "<th> Area (Ha) </th>";
echo "<th> Prod (Tm) </th>";
echo "</tr>";

foreach( $tbl->result() as $row ){

    $per = $this->ado->getSnapRow("dicPersonas","Id=$row->IdPers");
    $cul = $this->ado->getSnapRow("dicCultivos","Id=$row->IdCulti");
    $com = $this->ado->getSnapRow("dicComunis","Id=$row->IdComu");

    echo "<tr>";
    echo "<td> $nro </td>";
    echo "<td> $com->Zona </td>";
    echo "<td> $com->Prov / $com->Dist </td>";
    echo "<td> $row->Periodo </td>";
    echo "<td> $per->NUM </td>";
    echo "<td> $per->ApPaterno $per->ApMaterno $per->Nombres </td>";
    echo "<td> $cul->Nombre </td>";
    echo "<td> $row->Area </td>";
    echo "<td> $row->Prod </td>";

    echo "</tr>";
    $nro++;
}

echo "</table>";
echo "</div>";
}

public function cboComus()
{
    $this->session->IsLoggedAccess(DRAP_ADM);

    $idare = mISecurePost("area");
    $idcom = mISecurePost("comu");

    $tbl = $this->ado->getTable( "vwComuniAli", "Zona='$idare" );

```

```

        foreach( $tbl->result() as $row ){
            echo "<option value='$row->IdComu'> $row->Comunidad - ($row-
->NumEmpads)</option>";
        }
    }

    public function seeActivis()
    {
        $this->session->IsLoggedAccess(DRAP_ADM);
        $sess = $this->session->GetData(DRAP_ADM);

        $idare = mlSecurePost("area");
        $idcom = mlSecurePost("comu");

        $args = array(
            'comu' => $idcom,
            'comp' => $this->ado->getTable( "dicComponens" ),
            'acti' => $this->ado->getTable( "dicActividades", "IdComu=$idcom" )
        );

        $this->loadView( "adm/inActiv", $args );
    }
}

```

D. OBJETO LANDSCAPE

```

<?php if( !defined('LX_COREDIR') ) exit('LxCore is not!');

class Web extends IxApp
{
    public function __construct()
    {
        parent::__construct();

        $this->loadModel( "ado" );
    }

    public function index()
    {
        $this->loadView( "web/base" );
    }

    public function listAsocs()
    {
        ;
    }
}

```