

UNIVERSIDAD NACIONAL DEL CALLAO
FACULTAD DE INGENIERÍA INDUSTRIAL Y DE SISTEMAS
UNIDAD DE INVESTIGACIÓN



**“USO DE TALLERES DE DESARROLLO DE SOFTWARE EN LA
FORMACIÓN PROFESIONAL DE LOS ESTUDIANTES DE LA
ESCUELA DE INGENIERÍA DE SISTEMAS DE LA UNIVERSIDAD
NACIONAL DEL CALLAO”**

ANIVAL ALFREDO TORRE CAMONES

Resolución N° 878-2017-R
(01 septiembre 2017- 30 agosto 2019)

Callao, 2019

PERÚ

A small, handwritten signature in black ink, located in the bottom right corner of the page.

HOJA DE REFERENCIA DE APROBACIÓN

El Director de la Unidad de Investigación de la Facultad de Ingeniería Industrial y de Sistemas de la Universidad Nacional del Callao, hace constar que el Informe Final del Trabajo de Investigación titulado: **“USO DE TALLERES DE DESARROLLO DE SOFTWARE EN LA FORMACIÓN PROFESIONAL DE LOS ESTUDIANTES DE LA ESCUELA DE INGENIERÍA DE SISTEMAS DE LA UNIVERSIDAD NACIONAL DEL CALLAO”** presentado por el Mg. ANIVAL ALFREDO TORRE CAMONES, cumple con los requerimientos establecidos en la Resolución 499-2018-R de fecha 29 de Mayo del 2018 y su Directiva N° 008-2018-R.

Se expide la presente para los fines que el interesado crea conveniente, dejando constancia de la APROBACIÓN de dicho Trabajo de Investigación.

Atentamente,

30 de Enero del 2020



UNIVERSIDAD NACIONAL DEL CALLAO
FACULTAD DE INGENIERÍA INDUSTRIAL Y SISTEMAS
UNIDAD DE INVESTIGACIÓN

Dr. GUILLERMO MAS AZAHUANCHE
DIRECTOR DE LA UNIDAD DE INVESTIGACIÓN

DEDICATORIA:

A los docentes investigadores de la Facultad de Ingeniería de Industrial y de sistemas por su colaboración en el presente investigación.

A small, handwritten signature in black ink, located in the bottom right corner of the page. The signature is stylized and appears to be a single name or set of initials.

ÍNDICE

ÍNDICE DE CONTENIDO

ÍNDICE	1
RESUMEN.....	4
ABSTRACT	5
INTRODUCCIÓN:.....	6
I. PLANTEAMIENTO DEL PROBLEMA	9
1.1 Descripción de la realidad problemática	9
1.2 Formulación del Problema	11
1.2.1 Formulación del Problema General.....	11
1.2.2 Formulación de los Problemas Específicos:	11
1.3 Objetivos: General y Específicos	11
1.3.1 Objetivo General:.....	11
1.3.2 Objetivos Específicos:.....	11
1.4 Limitantes de la Investigación	12
1.4.1 Limitantes de la Investigación teórica:	12
1.4.2 Limitantes de la Investigación temporal:.....	12
1.4.3 Limitantes de la Investigación espacial:.....	12
II. MARCO TEÓRICO	13
2.1 Antecedentes:.....	13
2.1.1 Antecedentes Internacionales.....	13
2.1.2 Antecedentes Nacional	18
2.2 Bases Teóricas	32
2.3 Marco conceptual	66
2.3.1 Talleres de desarrollo de Software.....	66
2.3.2 Formación profesional.....	66
2.4 Definición de términos básicos:	68
III HIPÓTESIS Y VARIABLES.....	72
3.1 Hipótesis.....	72
3.1.1 Hipótesis General:	72
3.1.2 Hipótesis Específicas:	72

3.2	Definición conceptual de las Variables:.....	72
3.2.1	Operacionalización de variables	73
IV:	DISEÑO METODOLÓGICO	74
4.1	Tipo y diseño de la investigación	74
4.2	Método de investigación	75
4.3.	Población y muestra.....	75
4.4.	Lugar de estudio y periodos desarrollado	76
4.5	Técnicas e instrumentos para la recolección de la información documental	76
4.6	Análisis y procesamiento de datos.....	78
V.	RESULTADOS	83
5.1	Resultados descriptivos	83
VI.	DISCUSIÓN DE RESULTADOS	87
6.1	Contrastación y demostración de la hipótesis con los resultados	87
6-1.1	Prueba de la primera hipótesis secundaria.....	87
6.1.2	Prueba de la segunda hipótesis secundaria	90
6.1.3	Prueba de la hipótesis general.....	93
6.2	Contrastación de los resultados con otros similares	95
	CONCLUSIONES	106
	RECOMENDACIONES.....	107
	REFERENCIAS BIBLIOGRAFICAS.....	109
	ANEXOS	111
	MATRIZ DE CONSISTENCIA.....	111
	INSTRUMENTO DE RECOLECCIÓN DE DATOS	112
	VALIDACIÓN DEL INSTRUMENTO	113
	BASE DE DATOS.....	114

ÍNDICE DE GRÁFICOS

Gráfico 1 Descripción de la Realidad problemática	10
Gráfico 2 Formas de Software	49
Gráfico 3 Fases del desarrollo del Software	50
Gráfico 4 Ciclo de Vida de un software.....	63
Gráfico 5 Actividades de Implementación de la fábrica de software	80
Gráfico 6 Rack de Servidores streaming	81
Gráfico 7 Servidor streaming.....	81
Gráfico 8 Propaganda para la Capacitación	82
Gráfico 9 curva chi cuadrado.....	89
Gráfico 10 Curva chi-cuadrado	92
Gráfico 11 Curva chi cuadrado	96
Gráfico 12 Curva chi-cuadrado	97
Gráfico 13 Curva Normal	98
Gráfico 14 Tics en la Facultad de Ingeniería Industrial y de Sistemas.....	115
Gráfico 15 Aplicación de las Tics.....	116
Gráfico 16 Confiabilidad de los Tics.....	117
Gráfico 17 Nivel de Implementación de las Tics	118
Gráfico 18 Gestión en la facultad	119
Gráfico 19 Opinión sobre creación de Talleres de Software.....	120
Gráfico 20 Opinión sobre si los Talleres es una buena alternativa.....	121
Gráfico 21 El Estatuto y los talleres de Software	122
Gráfico 22 Los talleres y los conceptos teóricos.....	123
Gráfico 23 Los talleres de software y la experticia	124
Gráfico 24 Los talleres y la solución de problemas	125
Gráfico 25 Los talleres y la pertinencia.....	126
Gráfico 26 Trabajo en equipo	127
Gráfico 27 El Liderazgo	128
Gráfico 28 Necesidades de los clientes.....	129
Gráfico 29 Actitudes positivas	130

ÍNDICE DE TABLAS

Tabla 1 OPERACIONALIZACIÓN	73
Tabla 2 RESUMEN DE RESULTADOS	84
Tabla 3 RESULTADOS DE LA EVALUACIÓN A LOS DOS GRUPOS	85
Tabla 4 FRECUENCIA OBSERVADA	88
Tabla 5 FRECUENCIA ESPERADA	88
Tabla 6 VALOR CHI-CUADRADA.....	89
Tabla 7 FRECUENCIA OBSERVADA	91
Tabla 8 FRECUENCIA ESPERADA	91
Tabla 9 VALOR CHI-CUADRADA.....	92
Tabla 10 VALOR CHI-CUADRADA.....	96
Tabla 11 VALOR CHI-CUADRADA.....	97
Tabla 12 CUESTIONARIO.....	112
Tabla 13 ALFA DE CRONBACH	113
Tabla 14 RESUMEN DE RESULTADOS	114

RESUMEN

Durante la investigación se estudió el uso de talleres de desarrollo de software en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao. El objetivo de esta investigación fue explicar de qué manera el uso de Talleres de desarrollo de Software influye en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao en el 2017 – 2018.

El método fue el ex post facto, considerando que manipula deliberadamente al menos una variable independiente para ver su efecto y relación con las variables dependientes. Se concluye que, el Uso de Talleres de desarrollo de Software influye significativamente en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao en el 2017 - 2018. La población en estudio fue 120 estudiantes del IV y V ciclo de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao que están matriculados en el Semestre académico 2017 - 2018. La muestra estará conformada por 92 estudiantes del IV y V ciclo de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao que están matriculados en el Semestre académico 2017 - 2018. La muestra fue dividida en dos grupos de manera aleatoria. En uno de los grupos se aplicó el programa de fábrica de software mientras que en el otro grupo seguirá la enseñanza tradicional.

En la Escuela de Ingeniería de sistemas de la Universidad Nacional del Callao, los alumnos son prácticamente consumidores de programas y lenguajes de programación, su experiencia en el desarrollo de programas lo tiene recién cuando egrese de la Universidad y cuando la empresa le indique sus requerimientos. Esta experticia de los estudiantes se puede adelantar con las “fábrica de software” trabajando con necesidades reales, desarrollando en equipo los programas solicitados.

PALABRAS CLAVE: TALLERES DE DESARROLLO DE SOFTWARE FORMACIÓN PROFESIONAL

ABSTRACT

During the investigation, the use of software development workshops in the professional training of students of the School of Systems Engineering of the National University of Callao was studied. The objective of this research was to explain how the use of Software Development Workshops influences the professional training of the students of the School of Systems Engineering of the National University of Callao in 2017 - 2018.

The method was the ex post facto, considering that it deliberately manipulates at least one independent variable to see its effect and relationship with the dependent variables. It is concluded that, the Use of Software Development Workshops significantly influences the professional training of the students of the School of Systems Engineering of the National University of Callao in 2017 - 2018. The population under study will be 120 students of the IV and V cycle of the School of Systems Engineering of the National University of Callao who are enrolled in the Academic Semester 2017 - 2018. The sample will consist of 92 students of the IV and V cycle of the School of Systems Engineering of the National University of Callao who are enrolled in the Academic Semester 2017 - 2018. The sample was divided into two groups randomly. In one of the groups the software factory program will be applied while in the other group the traditional teaching will continue.

At the School of Systems Engineering of the National University of Callao, students are practically consumers of programs and programming languages, their experience in program development is only when they leave the University and when the company indicates their requirements. This student expertise can be advanced with the "software factory" working with real needs, developing the requested programs as a team.

KEY WORDS: SOFTWARE DEVELOPMENT WORKSHOPS PROFESSIONAL TRAINING

INTRODUCCIÓN:

La presente investigación propone el uso de Talleres de desarrollo de Software en la formación profesional de los estudiantes denominados también “fábricas académicas de software” haciendo una similitud a la fábrica de bienes en una empresa industrial. Ya algunas universidades que poseen facultades de sistemas tienen áreas dedicadas a la ingeniería de software tal es el caso de la Pontificia Universidad Católica de Chile en 1998, la Universidad de Lima en el 2008 y Universidad San Martín de Porras apoyado por la IBM en el 2010 y en otras universidades ha incentivado la realización de estudios de investigación para aclarar el significado de dicha expresión, así como la discusión sobre la posibilidad de ampliar su significado con el fin de darle un uso académico y establecer los requerimientos para implementar una fábrica académica de software en la Facultad de Ingeniería de Sistemas.

En anteriores investigaciones se realizaron algunas entrevistas a gerentes de fábricas de software de nuestro medio y se presentó un informe preliminar, que fue el punto de partida de la discusión sostenida en un focus group, en el que participaron los gerentes de las fábricas de software más representativas de Lima. Los resultados de estas investigaciones han confirmado que la expresión fábrica de software es una analogía que trata de expresar la conveniencia de copiar las mejores prácticas usadas en diferentes industrias que fabrican bienes, para utilizarlas en el proceso de desarrollo de software.

Los procesos de una fábrica de software tienen las siguientes características: configuran, adaptan y ensamblan rápidamente componentes; usan patrones de industrialización, estándares, actividades repetibles y automatizables; miden el desempeño y la productividad de las personas que participan en el desarrollo del software; utilizan la planificación para controlar el desarrollo

acelerado del software; y sus actividades son establecidas con criterios de rentabilidad, tratando de obtener presupuestos y plazos predecibles, así como una calidad aceptada en el ámbito mundial. Construir un producto software utilizando una fábrica de software implica las siguientes actividades: establecer los requerimientos del producto en coordinación con el usuario, realizar un análisis para indicar qué hará el producto, y luego especificarlo formalmente para pasar a diseñarlo y a establecer los componentes que se pueden volver a usar en su construcción. Terminado el diseño se puede pasar a la implementación, des pliegue y prueba del producto.

La presente investigación aplicada trata básicamente sobre el uso de talleres de desarrollo de Software en el proceso aprendizaje de los alumnos de la Escuela de Ingeniería de Sistemas, de cuarto y quinto ciclo. El objetivo del proyecto es desarrollar un sistema de desarrollo del talento universitario apoyado en una plataforma que permita hacer aplicativos por los alumnos y profesores. Esta metodología que consiste en lograr la formación integral en los alumnos a través de la teoría que se imparte en las aulas y la práctica en las “fabricas” o talleres de software acondicionados adecuadamente en un laboratorio de Cómputo.

Se utilizó el software libre para la elaboración de programas y sistemas que resuelvan los problemas que plantean las empresas, como son las páginas web, sistemas de facturación, sistemas de almacén, sistemas de facturación, sistemas de ventas, sistemas de producción, sistemas de contabilidad, sistemas de personal, sistemas de logística sistemas de alta gerencia o sistemas de información que integre los sistemas anteriores.

Con esta metodología los estudiantes que reciben los conocimientos teóricos pasarán a simular la solución de las situaciones más habituales que se puede presentar en una empresa, a diferencia de la formación profesional anterior, el docente utiliza solamente los

recursos teóricos y ejemplos prácticos para poder capturar y mantener la atención del estudiante con el fin de cumplir con el proceso de enseñanza-aprendizaje.

En la Escuela de Ingeniería de sistemas de la Universidad Nacional del Callao, los alumnos son prácticamente consumidores de programas y lenguajes de programación, su experiencia en el desarrollo de programas lo tiene recién cuando egrese de la Universidad y cuando la empresa le indique sus requerimientos. Esta experticia de los estudiantes se puede adelantar con las “fábrica de software” trabajando con necesidades reales, desarrollando en equipo los programas solicitados.

I. PLANTEAMIENTO DEL PROBLEMA

1.1 Descripción de la realidad problemática

Mertens, (1994) Los cambios tecnológicos hacen necesario una formación académica integral, acercándose a la realidad de las empresas y necesidades de la comunidad. En el ámbito educativo, la enseñanza - aprendizaje está pasando por una innovación que deja de lado la tradicional relación Profesor - Alumno. El Uso de Talleres de desarrollo de Software en la Educación es una nueva metodología que consiste en transmitir los conocimientos necesarios para experimentar con el desarrollo software

En la Escuela de Ingeniería de sistemas de la Universidad Nacional del Callao, los alumnos son prácticamente consumidores de tecnología extranjera en lo que se refiere al uso de programas y lenguajes de programación, su experiencia en el desarrollo de programas lo tendrá recién cuando egrese de la Universidad y cuando la empresa le indique sus requerimientos. Esta experiencia se puede adelantar si en la Escuela de Sistemas se tuviera un taller o “fábrica de software” donde con necesidades reales, tenga preparar en equipo los programas solicitados. Es un motivo para comprometer a los docentes y estudiantes para asumir retos de mejora continua en la calidad de enseñanza.

Los estudiantes de la Escuela de Ingeniería de Sistemas desarrollan asignaturas de tipo teórico-práctico cumpliendo los contenidos de sus respectivos sílabos en la parte teórica y en escasas ocasiones ven problemas reales con los detalles del caso en lo que se refiere al análisis y al diseño de los programas.

Gráfico 1 Descripción de la Realidad problemática



Fuente: Propia

Las causas del problema son básicamente tres como se observa en el Gráfico 1, no cuenta con talleres de software, no hay convenios con universidades que cuentan con talleres y no hay alianzas con empresas para desarrollar software.

Los efectos son notorios en los egresados porque no logran las competencias de la carrera y no pueden competir con egresados de otras universidades que si cuentan con talleres de software y además los egresados tendrán que invertir para lograr la experticia recién en la vida profesional a costa de cursos de capacitación en instituciones especializadas como Tecsup o la Universidad Nacional de Ingeniería. Es necesario que la Escuela Profesional de Ingeniería de Sistemas de la UNAC cuente con una “Fábrica de Software” para mejorar el nivel de preparación de los estudiantes y que sean más competitivos cuando egresen de las aulas. Esta “fábrica de software” debe ser implementada para resolver el problema de la falta de talleres.

1.2 Formulación del Problema

1.2.1 Formulación del Problema General

¿De qué manera el Uso de Talleres de desarrollo de software influye en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao en el 2017 - 2018?

1.2.2 Formulación de los Problemas Específicos:

- a) ¿De qué manera los usos de tecnologías de información en los Talleres de desarrollo de Software influyen en el rendimiento académico de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao?

- b) ¿De qué manera la gestión en los Talleres de desarrollo de Software, influye en la calidad de aprendizaje de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao?

1.3 Objetivos: General y Específicos

1.3.1 Objetivo General:

Explicar de qué manera el uso de Talleres de desarrollo de Software influye en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao en el 2017 – 2018.

1.3.2 Objetivos Específicos:

- a) Precisar de qué manera el uso de tecnologías de información en los Talleres de desarrollo de

Software influyen en el rendimiento académico de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao.

- b) Determinar de qué manera la gestión en los Talleres de desarrollo de Software, influye en la calidad de aprendizaje de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao.

1.4 Limitantes de la Investigación

1.4.1 Limitantes de la Investigación teórica:

Una limitante podemos ubicarla en la poca bibliografía encontrada en la Biblioteca Especializada de la Facultad de Ingeniería Industrial y de Sistemas y la Biblioteca Central de la Universidad Nacional del Callao.

1.4.2 Limitantes de la Investigación temporal:

La información obtenida se obtuvo corresponde a los semestres 2017 y 2018.

1.4.3 Limitantes de la Investigación espacial:

La información sobre el uso de Talleres de desarrollo de Software en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas es muy dispersa dado que las universidades particulares ya tienen talleres o “fábricas” de software como por ejemplo la Universidad de Lima, La Pontificia Universidad Católica del Perú, la Universidad de San Martín y muchas Universidades extranjeras.

II. MARCO TEÓRICO

2.1 Antecedentes:

2.1.1 Antecedentes Internacionales

En nuestra búsqueda de información vía Internet se encontró que las Universidades de España y México han desarrollado Proyectos sobre Desarrollo del Software. Estos Talleres de desarrollo, investigan los modos de crear software alterno que se asemejen con diversos grados de exactitud a un software real. Todas estas experiencias de entrenamiento con problemas reales, permiten a los profesores y estudiantes desarrollar y utilizar herramientas y estrategias para, explorar y manipular datos experimentales de casos reales. Por ejemplo, un diseñador puede utilizar el software para crear el primer prototipo y analizar la viabilidad de su producto, mientras que un fabricante quizás emplee el software porque es el único modo de fabricar un componente complejo.

Cendejas (2014), en su investigación realizada en la Universidad Popular Autónoma del Estado de Puebla titulada "Implementación del modelo integral colaborativo (MDSIC) como fuente de innovación para el desarrollo ágil de software en las empresas de la zona centro - occidente en México", buscó definir un modelo que contribuya al creación de una fábrica de software para pequeñas y medianas empresas que tienen pocos recursos en tiempo y costo; y que necesitan desarrollar sistemas de información ya sea para uso interno, o para venta, debido a que no cuentan con los procesos de producción de software bien definidos. Los resultados permitieron al investigador diseñar e implementar un modelo que brinde las bases para el desarrollo de

software de calidad y que contribuya en la realización de proyectos tecnológicos en empresas del área centro occidente de México. Así mismo uno de los aportes al estudio es haber detectado que los modelos más empleados el día de hoy hacen referencia al desarrollo ágil del software para el desarrollo rápido y en aquellos que ponen énfasis en el trabajo basado en procesos (p.19).

Cendejas (2014), también ha podido aplicar el modelo MDSIC en compañías del rubro público y privado, obteniendo así un cuadro comparativo de ventajas de su aplicación, así también realizó la comparativa de la aplicación del modelo MDSIC versus otras principales metodologías usadas en la industria de software, midiendo el impacto, costo y beneficio (p.21). La investigación realizada por Cendejas se encuentra dentro del marco de estudio referido a modelos de fábrica de software, brindando la directriz a través de su modelo para producir software de calidad siendo este aplicable en organizaciones públicas y privadas.

Guajardo, G. (2014), en el trabajo de su investigación realizada en la Universidad Andrés Bello de Chile titulada “Determinación y Cuantificación de los Factores Relevantes relacionados con el Negocio que Afectan la Elección Metodológica de Proyectos Informáticos sobre las Metodologías Ágiles”, buscó definir cuáles los factores relevantes que contribuyen en la elección de una metodología de desarrollo de software basados en las metodologías ágiles, analizó sus características y los factores que la diferencian, estableciendo cuál es la más apropiada para cada proyecto en particular. Los resultados permitieron al investigador estructurar en forma sistemática y secuencial

el proceso de selección de una metodología de desarrollo ágil para un proyecto específico, basándose en criterios objetivos y cuantificables (pp.4-5). La investigación de Guajardo se encuentra dentro del marco de estudio referido a modelos de fábrica de software que analiza el uso de metodologías de desarrollo de software acorde a las necesidades de cada proyecto informático.

Araus (2013), en su trabajo de investigación desarrollada en la Universidad Católica de Valparaíso, titulada mencionada “Desarrollo de un patrón Genexus para la construcción de Aplicaciones SOA”, buscó definir un modelo de fábrica de software que oriente la construcción de aplicaciones de servicios web (web services) a través de la herramienta Genexus, creando una arquitectura orientada a servicios (SOA), esto teniendo como principal objetivo el rápido desarrollo de sistemas de información, mediante los beneficios que ofrece el patrón Genexus como la reutilización y el bajo acoplamiento de los componentes. Los resultados le permitieron al investigador describir y comprobar las funcionalidades de la herramienta Genexus como una nueva metodología de diseño y elaboración de aplicaciones, permitiéndole la normalización y diseño de datos (estructura), creación y mantenimiento de la base de datos y de los programas existente, obteniendo los beneficios en reducción de recurso como tiempo y costo, debido a que trabaja bajo un enfoque orientado a la reutilización del conocimiento. Uno de los aportes del investigador fue exponer los beneficios que brinda las funcionalidades del modelo Genexus como la reutilización de componentes, la abstracción los servicios son autónomos, revisión, monitoreo, facilidad de escalabilidad y

mantenibilidad (pp.18-19). La investigación de Araus se encuentra enmarcado en el estudio de modelos de fábrica de software, brindándonos la opción de utilizar el patrón Genexus para 18 implementar un modelo de fábrica de software, con la cual se incentiva el desarrollo rápido basado en generación automática de código multiplataforma.

Matturro (2010), en su investigación en la Universidad Politécnica de Madrid titulada “Modelo para la gestión del conocimiento y la experiencia integrada a las prácticas y procesos de desarrollo software”, buscó definir un modelo de gestión de conocimiento (knowledgemanagement), para administrar el conocimiento y experiencias en proyectos de software, con el cual las practicas realizadas y registradas en esta base de conocimientos lleguen a ser tomadas en los proyectos software junto con mejoras prácticas y procesos software en uso en la organización. Los resultados le permitieron establecer los procedimientos de uso de las herramientas requeridas para capturar los conocimientos y las experiencias obtenidas por el equipo, en forma paralela a la realización de las actividades de los proyectos, que difieren de los modos habituales basados en el análisis posterior a la culminación de los proyectos y técnicas similares. Uno de los aportes del investigador fue presentar las fases y actividades que conforman modelo utilizado para la gestión del conocimiento, pudiendo concluir que el modelo propuesto forma las bases para administrar del conocimiento en proyectos de software (pp.11-12). Así también

Matturro (2010), con la investigación realizada le permitió concluir que el arquetipo propuesto sea viable pudiendo ser implementado e integrado a las fases de los proyectos

software, y que esta integración no implica una carga excesiva de trabajo para los integrantes de los equipos de los proyectos de software, siendo fácil identificar lecciones aprendidas, presentando propuestas de estandarización de mejores prácticas relativas a procesos software en favor de la organización (p.182). La investigación de Maturro se enmarca en una metodología para la gestión del conocimiento dentro de un modelo para una fábrica de software que tiene el propósito de utilizar esos conocimientos y aprendizajes como sustento para las actividades de mejora.

En el ámbito internacional, las empresas ven necesario desarrollar distintas aplicaciones o sistemas de información ya sea para su uso interno, atención a sus clientes, o de venta de software especializado; con el objetivo de encontrar diferenciación en sus servicios brindados que le permitan contar con ventajas competitivas comerciales. Siendo el caso de la empresa IBM con modelo de capacidad de madurez Integración Nivel 5 (CMMI5), desarrolla aplicaciones para países como EEUU, Londres, China, India o Egipto (IBM 2015, p.1).

Dinero (2011) se refirió sobre la industria del software de la siguiente manera: La industria del Software ha crecido recientemente, debido a que se requiere desarrollos de productos de software para distintos sectores como en educación, economía, salud, tecnología limpia y aplicaciones móviles, están generando la próxima ola del desarrollo del software de la India, que se opta por un mecanismo de producción de software que asegure competitividad empresarial (p.43).

2.1.2 Antecedentes Nacional

Para desarrollar el presente proyecto de investigación se investigó todo aquello relacionado con Proyectos Educativos similares que fueron desarrollados por universidades peruanas. En el caso de la Universidades e instituciones de Educación Superior del país se necesita contar con herramientas que incrementen las capacidades administrativas y de toma de decisiones de sus futuros egresados. Por medio de los Talleres de desarrollo de Software se pretende que los estudiantes participen en programas o sistemas orientados a la Producción, Comercio, Servicios y/o Finanzas. La Tecnologías de Información y la Industria de Software, se encuentran relacionada estrechamente en nuestros días, la construcción de software es un proceso muy complejo donde muchos “nuevos alquimistas” buscan la forma de industrializar el software, pero su producción es un proceso mucho más que una manufactura propia; es una combinación de ciencia, arte, ingeniería y habilidades de administración de proyectos.

Mendoza (2016) indica que; Los empresarios locales del rubro de fábrica de software, deben incorporar las nuevas tecnologías vigentes para no perder competitividad frente a fabricantes de talla regional o mundial que desde el Perú atienden contratos nacionales y regionales. Existe demanda de Sistemas de Información desde el sector privado y más en el sector público, donde el gobierno impulsa directivas para obligar a los ministerios a tomar las medidas pertinentes que permitan la continuidad de los servicios públicos (p.21).

El problema surge cuando los proyectos se entregan fuera de tiempo y el recurso económico utilizado sobrepasa los planificado, el código creado no es reutilizable, el proceso de

inversión es muy largo y los requerimientos no son realizados con mucha rapidez, y el personal solo logra dominar ciertas herramientas.

El reto de Tecnología de Información es proveer a las empresas la capacidad para responder rápidamente ante los cambios siendo un factor clave para la competitividad y el crecimiento de las organizaciones. La rapidez de éstas fue si cuestionada el área de Tecnología de Información no pueden responder de forma flexible a los cambios que afectan a la actividad de negocio y no les permite adaptarse adecuadamente a las presiones del mercado.

Barzola y Henríquez (2014), en la investigación realizada en la Universidad San Martín de Porres, titulada “Diseño de una metodología de certificación de productos de software orientado al sector público”, definieron una metodología para planificar, desarrollar, verificar, validar los productos de software construidos, con el fin de minimizar los errores de definición, construcción e implementación de productos de Software. Los resultados le permitieron al investigador contar con una metodología que le entregue el control adecuado de la planificación realizada a nivel de tareas, actividades desempeñadas, recursos y costos requeridos, permitiéndole medir el desempeño, la competencia y la calidad obtenida. Uno de los aportes del investigador fue exponer que la metodología descrita se encuentra alineada a la Norma Técnica Peruana (NTP) 12207: 2006, que hace uso de listado de comprobaciones y en el proceso de desarrollo de software: inicio, elaboración, construcción y transición (p.21).

Barzola y Henríquez (2014), también entrega plantillas de documentos para la definición funcional, técnica, de

arquitectura, de control, de definición de recursos necesarios como base de datos, archivos, tablas, servidores proponiendo planes de control para el seguimiento del proyecto y supervisión de flujos de trabajo (p.197). La investigación realizada por Barzola y Henríquez se encuentra dentro del marco de los modelos de fábrica de software y desarrollo de sistemas de información, tema de interés en la investigación realizada. Así mismo precisó la importancia del uso la NTP 12207: 2006 en proceso de desarrollo de software. Montalvo et al. (2012), en la investigación realizada en la Universidad ESAN con el título de “Modelo de una fábrica de software: caso CosapiSoft”, buscaron establecer una propuesta de modelo de fábrica de software en la empresa CosapiSoft S.A. planteando estrategias enfocados en la industrialización de software, metodología de fabricación de software basado en estándares, modelo de procesos, de gestión del talento humano. Los resultados permitieron al investigador 20 establecer, en la fabricación de software CosapiSoft S.A. los procesos definidos, repetibles y a la vez medibles, que hace uso de los métodos de producción y fases verificables y controlables, desde su creación, entrega y posterior mantenimiento, así como la constatación que la calidad del producto software depende primordialmente de la calidad de los procesos empleados en su fabricación, siendo las personas el eje primordial. Los aportes del investigador fue descubrir que una fábrica de software deber alinearse al modelo de madurez de la capacidad integrada (CMMI) haciendo uso de sus estándares en el proceso de fabricación de software, así también hace uso de una biblioteca de infraestructura de tecnologías de información (ITIL) para la gestión de servicios (pp.11-14). La

investigación de Montalvo se encuentra dentro del marco de los modelos de fábrica de software, proponiendo un modelo para la operación de una fábrica de software en la empresa CosapiSoft S.A., permitiendo brindar el servicio de fabricación de software y competir en costos y calidad.

Berta (2011), en la investigación desarrollada en la Universidad de Piura con el título de “Incorporación de la integración continua en el desarrollo de software: caso de estudio: Organismo Supervisor de la Inversión en Energía y Minería”, buscó definir la integración continua en el desarrollo de software en la Oficina de Sistemas de la institución pública Organismo Supervisor de la Inversión en Energía y Minería (OSINERGMIN), donde desarrollan software de manera constante, debiendo responder a los cambios según las disposiciones legales y técnicos relacionadas con los subsectores de electricidad, hidrocarburos y minería. Los resultados permitieron al investigador encontrar la influencia directa y significativa entre la integración continúa propuesta y el desarrollo de software en la entidad OSINERGMIN, llegando a las conclusiones que la integración, construcción, pruebas, inspección, retroalimentación de la integración continúa propuesta influye en el desarrollo de software. Uno de los aportes del investigador es recomendar realizar reuniones que permitan que fluya el intercambio de experiencias de la integración continua como parte de la mejora continua, discutiendo las mejoras en la calidad del software (pp.16-18). La investigación de Berta se encuentra dentro del marco de Desarrollo de Sistemas de Información, tema de interés en la investigación realizada, precisando la

importancia de buscar la satisfacción del cliente y generar valor al negocio.

García L. (2009), en la investigación realizada en la Universidad de Piura con el título de “MEDESOFTE: Metodología de Desarrollo de Software en Entidades de Educación Superior”, buscó definir una solución a la situación problemática que aqueja a los centros de estudio y de investigación con carreras similares a la industria del software, debido a que no se cuenta con metodologías apropiadas para la gestión y la implementación de proyectos de desarrollo de software para producir software de alta calidad, haciendo uso de varias normas y modelos de procesos tales como el modelo de capacidad de madurez integrado (CMMI), norma técnica peruana ISO/IEC 12207, la mejora de proceso del software brasileño (MPS.BR) de Brasil y el modelo de procesos para la industria del software (MOPROSOFT) de México. Los resultados permitieron al investigador exponer los beneficios obtenidos con la aplicación de la propuesta expuesta, logrando que se reduzcan el tiempo de desarrollo de un proyecto de software (pp.5-149). La investigación de García se encuentra dentro del marco de Desarrollo de Sistemas de Información, siendo tema de interés en la investigación realizada.

Bombieri (2017) Indica que una fábrica de software viene a ser un entorno de desarrollo de aplicaciones cuyo fin es construir de forma rápida cualquier tipo específico de aplicación, sistema de información o software; un modelo de fábrica de software entrega metodologías con las mejores prácticas para gestión efectiva de procesos de construcción de software de forma que se alcanza niveles altos de

productividad, brindando la capacidad de configurar, modificar y ensamblar rápidamente componentes de software desarrollados de forma independientemente; para construir así conglomerados de sistemas similares con funcionalidades comunes en todos los sistemas, pero diferentes a la vez con funcionalidades personalizadas (p.13)

Al contar con componentes de calidad nos lleva a pensar que en el desarrollo de software es posible construir software también de calidad basados en la reutilización y personalización. Se ve que los avances logramos en la asimilación correcta de conceptos de reutilización de software y transferencia de conocimiento, es el comienzo. De esta manera la construcción de software basado en 24 componentes con la utilización de metodologías específicas, se convierte en el cimiento de la revolución industrial del software y se concreta hoy en día en diferentes formas de desarrollar software de alta calidad con los costos y en tiempos que antes no eran estimables (p.14).

González (2013) explicó el objetivo de funcionalidad de una fábrica de software como: Un ambiente dedicado a construir componentes y partes de software con procesos muy completos que sirven para la ejecución de sistemas informáticos basados en especificaciones o requerimientos. Trabaja como una línea de producción de ensamblado que se basa en los modelos o planos arquitectónicos para su construcción. La empresa Nothware define un modelo de fábrica de software con el fin de conseguir ventajas competitivas (p.3).

Gonzales y Abiuso (2013), indican que un modelo de fábrica de software en una empresa tiene el propósito de: Crear software a partir de especificaciones de requerimientos entregados los usuarios o clientes interesados, implementa un proceso para el desarrollo de software maduro en cada una de las etapas del ciclo de vida (proceso disciplinado para la industrialización del proceso de desarrollo), que se fundamenta en la gestión integral del servicio realizado; y esto involucra el uso de mejores prácticas que tiene esta industria, así como gestión haciendo uso de métricas para el monitoreo y la mejora continua de los servicios ofrecidos. Un modelo de fábrica de software utiliza metodologías relacionadas a las buenas prácticas en la industrialización de software a nivel de gestión, producción y comercialización (p.2).

MSDN (2012) explicó que una fábrica de software es un grupo estructurado de componentes de software relacionados entre sí, con objetivos comunes, que ayuda a arquitectos y desarrolladores de software a crear de manera previsible y eficiente productos de software de alta calidad con características específicas, se implementa un modelo de fábrica de software estableciendo los patrones, marcos y herramientas que el mismo describe, empaquetarlos, y ponerlos a disposición. La fábrica de software ayuda a crear aplicaciones que compartan una arquitectura y un grupo de características; los activos incluyen códigos reutilizables, documentación y mapas de implementaciones de referencia; las herramientas de software incluyen asistentes o wizard, generadores de código y diseñadores visuales. Normalmente, un modelo de fábrica de software proporciona

plantillas y otras herramientas para ayudar a un equipo de desarrollo a iniciar rápidamente el desarrollo de nuevas aplicaciones (p.1).

Greenfield (2004), explica que una empresa hace uso de un modelo de fábrica de software: Para la gestión de una línea de productos de software, configurando herramientas, procesos y contenido utilizando una plantilla o plano de la fábrica de software basada en un esquema de fábrica de software para simplificar, automatizar y mejorar el desarrollo y mantenimiento de variantes de un producto adaptando, ensamblando y configurando componentes basados en framework (p.18)

Carranza Z. (2017), define a los procesos de una fábrica de software como “procedimientos susceptibles a medición del desempeño y productividad en el desarrollo del software; utiliza la planificación y sus actividades son establecidas con criterios de rentabilidad, tratando de obtener presupuestos y plazos predecibles, así como la calidad” (p.10).

Azanza, Díaz y Trujillo (2010), explican el proceso de la industria del software como el proceso de la industria del software que hace uso de la experiencia de profesionales calificados en tareas manuales intensivas en mano de obra, la creciente presión costo y tiempo de comercialización, y para mejorar la calidad del software, ocasiona que el proceso utilice a métodos más automatizados, en este ámbito, el proceso de fábricas de software produce un cambio de paradigma con la promesa de desarrollar aplicaciones con más costo efectivo a través de la reutilización sistemática. Esto fomenta el aumento de las cadenas de suministro

mientras que abre la puerta a la personalización masiva de la industrialización de software (pp.11-12).

Pérez, Mendoza y Grimán (2006), define a los procesos de una fábrica de software en las organizaciones como los procesos de una fábrica de software que se desarrollan con el objetivo de pasar de la forma tradicional de construir software a la producción en masa, buscando la eficacia, adaptado a las necesidades de los cliente; el proceso está compuesto por identificación del problema; justificación y estudio de viabilidad; análisis y determinación de requisitos; diseño; pruebas; entrega e instalación y; mantenimiento. Así también para realizar el proceso de producción de software 27 empaquetado tiene los pasos siguientes: desarrollo; pruebas; duplicación de discos; embalaje y manuales; ventas y marketing; distribución; y apoyo (p.3).

AMCIS (2010), acerca de los procesos en una fábrica de software indica que se refiere a los pasos planeados que posibilitan a la empresa sobresalir en un nivel de capacidad acorde a las exigencias para entregar servicios de calidad y en el corto plazo tener una mayor madurez según el modelo, utiliza las buenas prácticas en el ciclo de vida del software, teniendo la capacidad de brindar los servicios de mayor calidad a las empresas, permitiendo que los colaboradores de una fábrica de software puedan tener experiencias alineadas a la realidad acorde a las exigencias del mercado, que sigue un modelo de gestión de procesos creado a partir de buenas prácticas (p.6).

Sommerville (2005) indica que un proceso de software es una agrupación racional de normas, políticas, procedimiento,

estructuras organizacionales, tecnologías, y artefactos que se necesitan para comprender, idear, definir, desarrollar, probar, desplegar y mantener un producto software” (p.80)

Wallace (2014) define de forma simple a un proyecto como “un producto único, propio, peculiar, exclusivo, y original que se crea solo una vez, y los sistemas junto con las herramientas empleadas para desarrollarlo se vuelven a utilizar en otras actividades, en muchos de los casos, se utiliza para llevar a cabo otros proyectos” (p.8).

Project Management Institute (2013), define un proyecto como “un trabajo temporal que se realiza para crear un producto único no repetible, que tiene una fecha de principio y una fecha final de termino bien definidos” (p.32). 28

ETOP (2007), define un proyecto como un grupo de actividades que están planificadas, definidas, estructuradas y relacionadas entre sí, que utilizan los insumos de tiempo y recursos específicos para lograr el objetivo y las metas trazadas, un proyecto se caracteriza por su intencionalidad de mejora y cambio, busca cambiar de su estado básico inicial a un estado ideal sabido a un objetivo, en el cual el problema se resuelve de manera total aprovechándose de una oportunidad de mejora (p.13).

Llorens (2005) indica que un proyecto es “ejecutar una actividad compleja con fin de obtener un objetivo específico, que tiene una duración finita, utilizando un conjunto finito de recursos, cada proyecto es único, cruza fronteras organizativas integrando esfuerzos multidisciplinarios” (p.9).

DTyOC (2016), indica que una metodología “tiene un enfoque de la ingeniería que hace uso de principios, métodos, conceptos, de prácticas y técnicas que se basadas en la Ingeniería del Software para desarrollar productos de software de alta calidad” (p.6).

Rivas (2015), define metodología como un conjunto de métodos disciplinados coherentes, que tienen principios comunes que explican su funcionamiento; encaminan las actividades para desarrollar software de forma ordenada. Desarrollar trata las acciones de incrementar o modificar aumentar alguna característica de algo físico (concreto) o algo intelectual (abstracto); por lo tanto, concluimos que metodología de desarrollo de software es el estudio y determinación de los métodos más adecuados para dar incremento o mantenimiento a los productos de software. A la fecha el término desarrollo es utilizado para referirse a las tareas que implican la creación, procesamiento, fabricación, actualización o modificación de un software (p.182).

Virueta (2010), explica que una metodología es un grupo de técnicas, herramientas y procedimientos basados en buenas prácticas con documentación de apoyo que ayuda a los desarrolladores a realizar un nuevo producto de software. Se basa en el ciclo de vida de desarrollo de software indicando que se consigue en cada etapa del desarrollo del proyecto, pero no explica el cómo hacerlo. Dependerá de la selección de la metodología a utilizar en el proyecto, es necesario conocer a fondo cada una de las metodologías de desarrollo de software existente para conocer bondades, beneficios y desventajas para aplicar la más apropiada en cada uno de los proyectos de desarrollo de software a implementar (p.4).

García (2009) indica que la metodología de una fábrica de software “definirá procesos, interdependencias entre los procesos, metas, objetivos, actividades, métricas de proceso, así como las técnicas y herramientas específicas que permitan fundamentar los cimientos para realizar una producción de software con calidad y de forma estandarizada”(p.6).

Carrillo (2008), las metodologías aplicadas a la industria del software: Hacen uso de los procedimientos, políticas, normas técnicas, herramientas y documentos de soporte para desarrollar un producto software de forma estandarizada. Estas metodologías tienen como fin orientar a los desarrolladores de software a crear un nuevo producto software de forma ordenada. Los requisitos para el desarrollo de los sistemas de información son tan variados y cambiantes, dando lugar a que se defina una gran variedad de metodologías y modelos que gestionan la creación del software. Se clasifican en dos grupos: metodologías destinadas al control de los procesos y metodologías destinadas a la interacción con los usuarios siendo llamadas metodologías ligeras o ágiles (p.3). Características de modelos de fábrica de software.

Aragón y De Sousa (2004), indican que las características fundamentales de una fábrica de software se desarrollan con: Proceso definido y estandarizado. Revisión manejada con el cliente. Solicitudes de servicio normalizados. Facilidad de obtención de costos y estimaciones de tiempo. Control de los recursos utilizados en cada fábrica de software. Control y almacenamiento en repositorios de artículos de software. El estado y ejecución de todos los servicios. Productos

producidos según normas. Establecidos en una organización. Equipo entrenado en organización y procesos productivos. Control de calidad de los productos. Procesos para servicio al cliente. Medidas para control de acuerdos en nivel de servicio con el cliente (pp.11-12). Pilares para construir un Modelo fábrica de software

Lenz y Wienands (2006), indican que “son 4 pilares básicos en una fábrica del software: desarrollo de la línea de productos, arquitectura de los marcos de trabajo (frameworks), orientación en contexto y desarrollo orientado a modelos” (p.14).

Lenz y Wienands (2006), además definen los cuatro pilares como sigue: Desarrollo de la línea de productos: La reutilización es el objetivo de este concepto como paradigma de las fábricas de software.; significa es que necesitamos definir el alcance de la Fábrica de Software que permite la reutilización sistemática, permitiendo proporcionar puntos de extensión para personalizaciones para cada producto en una línea de productos. Arquitectura de los marcos de trabajo (frameworks): proporcionan una arquitectura de línea de base que todos los miembros de la línea de productos, definimos una arquitectura que soporta todos los productos en el ámbito de la línea de productos definida. Los frameworks incorporan las mejores prácticas y patrones para un tipo de aplicación particular, lo que permite un desarrollo más eficiente con mayor calidad y una arquitectura uniforme. Se centra en una línea de productos con un alcance limitado para proporcionar un marco arquitectónico muy eficiente y de alta calidad para la línea de productos. Orientación en Contexto: Ya hemos implementado algunos productos

similares antes de decidir ir a la reutilización sistemática, podemos ofrecer las mejores prácticas a los desarrolladores como guía, hemos extraído las mejores prácticas de nuestras implementaciones iniciales y potencialmente puede darnos la orientación que necesitamos. La orientación en contexto ayuda a los desarrolladores de aplicaciones a implementar las variabilidades de una aplicación en particular. El resultado de instruir al desarrollador en qué hacer y cómo hacerlo no sólo aumenta la calidad del producto al prevenir errores, sino que también conduce a un desarrollo más eficiente y una curva de aprendizaje reducida para desarrolladores nuevos y / o inexpertos. Desarrollo orientado a modelos (MDD): tiene el objetivo de desarrollar software a un nivel de abstracción mucho más alto utilizando los conceptos de dominio empresarial, como flujo de trabajo, actividad o mensaje. Con el uso de términos como clase, evento, biblioteca.

El objetivo final es proporcionar modelos en diferentes niveles de abstracción. Esto se puede comparar con la construcción de una casa. Imagine modelos que muestren diferentes aspectos de la casa, tales como planos eléctricos, plomería y estructurales. En Software podríamos imaginar los mismos conceptos para que diferentes partes interesadas, tales como analistas de negocio, directores de proyectos, arquitectos de infraestructuras, arquitectos de soluciones, arquitectos de bases de datos, desarrolladores y probadores. Este enfoque permite no sólo trabajar en un nivel superior de abstracción, sino también verificar y validar el sistema antes de su implementación (pp.14-15).

2.2 Bases Teóricas

La tesis: “Uso de talleres de desarrollo de software en la formación profesional de los estudiantes de la Escuela de Ingeniería Industrial y de Sistemas de la Universidad Nacional del Callao” se enmarca En la corriente filosófica de Racionalismo que se desarrolló en Europa durante los siglos XVII y XVIII, formulada por Rene Descartes, que se complementa con el criticismo de Immanuel Kant, y que es el sistema de pensamiento que acentúa el papel de la razón en la adquisición del conocimiento, en contraste con el empirismo, que resalta el papel de la experiencia, sobre todo el sentido de la percepción.

El racionalismo se identifica ante todo con la tradición que proviene del filósofo y científico francés del siglo XVII René Descartes, quien decía que la geometría representaba el ideal de todas las ciencias y también de la filosofía. Descartes aseguraba que solo por medio de la razón se podían descubrir ciertas verdades universales. A partir de aquellas verdades es posible deducir el resto de contenidos de la filosofía y de las ciencias. Manifestaba que estas verdades evidentes en sí eran innatas, no derivadas de la experiencia. Este tipo de racionalismo fue desarrollado por otros filósofos europeos, como el holandés Baruch Spinoza y el pensador y matemático alemán Gottfried Wilhelm Leibniz. Se opusieron a ella los empiristas británicos, como John Locke y David Hume, que creían que todas las ideas procedían de los sentidos.

Stiglits, (1998) indica que en esa línea de pensamiento muchos informes internacionales en educación coinciden en caracterizar la sociedad del siglo XXI como una sociedad donde el rasgo dominante fue la preeminencia del saber, asigna a la educación el rol de ser el instrumento imprescindible para dominarla; la gran novedad actual es que personas que nunca se habían ocupado de

la educación, como los empresarios, destacan su importancia y exigen mayores niveles de calidad.

Es así, que por cinco siglos el texto impreso ha constituido no una sino "la forma" casi permanente de mediación en los diversos campos de la actividad humana, ha condicionado la percepción y hasta la manera de concebir los procesos educativos. A ello la Escuela le rinde tributo cotidianamente. Sin embargo, el resultado histórico y social no es a todas luces satisfactorias por sus limitaciones formativas. Los alumnos aprenden más y mejor fuera de los "Talleres educativos".

Senge (2005) Indica que en la actualidad se pretende que los alumnos obtengan como resultado educativo: una visión objetiva, integral, proyectiva o sistémica en "la percepción de las cosas", un enfoque contextual de entorno o situacional a la vez que multidimensional en "la explicación de las cosas", un equipamiento cognitivo eficiente para el manejo de la información una capacidad adecuada para resolver problemas, mostrar un desempeño exitoso y una actuación creativa, y una habilidad para aprender a pesar de las limitaciones en los recursos.

En las últimas décadas con la aparición de la computadora, la memoria digital, los multimedia y hasta el Internet actual, se observa un avance vertiginoso de la tecnología de la información, donde el denominador común es la informática.

Mayor (1997) indica que el positivismo franquea la educación superior y la capacidad cognitiva como *"La clave de un desarrollo sostenible e independiente es la educación, educación que llega hasta todos los miembros de la sociedad, a través de nuevas modalidades, nuevas tecnologías a fin de ofrecer oportunidades de aprendizaje durante toda la vida para todo debemos estar dispuestos, en todos los países, a remodelar la educación de forma de promover actitudes, aptitudes y comportamientos conducentes a una cultura de la sostenibilidad"*, el ser competente, es ser capaz dentro de la(s) sociedad(s) y cumplir algún rol en la

actividad productiva o de servicios administrativos, entre otros y como trabajador incorporarse a la PEA (Población Económicamente activa) calificadamente y ser competitivo permanente a las actividades que nos permitan alcanzar niveles auspiciosos de bien social, organizacional y un compromiso.

En este marco de referencia, sostenemos que la educación competitiva debe orientarse a lograr que las personas dispongan de un panorama personal que lo ubique en el contexto global con capacidad de aprender en un entorno cambiante. Quienes insisten en aprender "cada vez más" en lugar de "cada vez distinto", caminan mirando al pasado.

GATLIFF, Bee; WENDEL, Federick (1998) al examinar las perspectivas generales de Simulación en la educación universitaria identifican cuatro dimensiones donde los estudiantes y profesores pueden potenciar la capacidad creativa y conseguir mayor flexibilidad mental, dentro de un proceso dinámico de enseñanza - aprendizaje. Estas dimensiones son las siguientes:

- La comunicación por computadoras
- Experiencias de interconexión educativa
- Información, investigación y Publicaciones
- Desarrollo de Proyecto educativos

Una síntesis teórica de estas dimensiones se desarrolla a continuación.

a. Comunicación por Computadoras

Internet ha facilitado la creación y difusión de diversos medios que permiten potenciar las comunicaciones por computadora. Los estudiantes que se valen de este software ven incrementada su capacidad para proveer, obtener o intercambiar información con otras personas o instituciones independientemente de su ubicación espacio temporal. Así, estudiante y profesor puede escoger una amplia gama de Software y herramientas: @email, listas de interés, los new groups y gopher, ftps, el chat, la conferencia, la vídeo conferencia, el hipertexto, y las páginas Web, los motores de

búsqueda, las bases de datos, etc. Estas herramientas valen para facilitar o incrementar la comunicación y como medios para "simular", "actuar" para "hacer cosas" y en especial para la interacción humana permitiendo nuevos géneros y modalidades de comportamiento entre personas e instituciones.

b. Experiencias de Interconexión Educativa

La clásica instrucción centrada en la enseñanza se ha convertido en el aprendizaje centrado en "quien aprende" y el aprendizaje del aula ha migrado fuera de ella, convirtiéndose en aprendizaje distribuido porque se vale de recursos (de información, personales e institucionales) que no se encuentran en ningún "centro educativo" sino fuera de ellos, y ocurre a través de experiencias e interacciones propias de otros entornos no escolarizados. Ya existen las herramientas (medios), recursos de información (contenido), entornos, y ambientes que promueven interacciones y experiencias de interconexión educativa. Internet es un entorno "constructivista" que favorece el descubrimiento, el aprendizaje, el enlace con los expertos, la interacción social, las comunidades de aprendizaje, la construcción del conocimiento, el tránsito a través de la "zona de desarrollo proximal".

Las experiencias de simulación y navegación en el Ciberespacio enlazados bajo una perspectiva sistémica favorecerán el aprendizaje y las experiencias formativas de los usuarios. Se trata de una educación en sí competitiva, orientada a tres logros fundamentales en las personas.

- . Una Cultura extensiva o amplia
- . Formación especializada intensiva
- . Una capacidad de aprendizaje para toda la vida

c) Información Investigación y Publicación

Esta dimensión de la perspectiva educativa del software de simulación define el carácter y filosofía de la informática. Los recursos de información de los que unos disponen son ofrecidos a una comunidad más amplia, en el caso de Internet el principio es

compartir, tenemos el derecho a "tener" en correspondencia con nuestra obligación de "ofrecer". Mención especial merecen en este acápite, las bases de datos, y los "motores de búsqueda". Las primeras son entidades de "consulta interactiva" que han revolucionado los tradicionales bancos de datos. Igualmente disponemos de los archivos News Groups y Listserv así como de sus respectivos FAQs (documentos de preguntas y respuestas) que permiten efectuar un seguimiento o reconstruir los itinerarios conceptuales de quienes participaron en sus debates o unirnos a la travesía cognitiva que significa participar en tales foros al expandir el espacio y resolver el problema de distancia, los Talleres de desarrollo de Software permiten ahora algunas alternativas para la ejecución de proyectos de enseñanza y de investigación, anteriormente estos proyectos representaban costos importantes fuera del alcance de cualquier Universidad que deseaba realizar educación interactiva. A propósito de nuestra Tesis, hemos conducido un proyecto exploratorio con los alumnos del sexto y octavo ciclo de Ingeniería Industrial de la Universidad del Callao sobre la Simulación de Negocios utilizando el correo electrónico y páginas Web. Los problemas planteados y su posterior solución fueron manejados "virtualmente". Esto nos hace ver las cosas de manera distinta y ser optimistas al respecto. No hay instrucción sin información ni educación sin investigación. La Simulación de problemas reales favorece estas posibilidades, permitiendo la difusión de información, opiniones o resultados de trabajos académicos, es en sí, un centro de enseñanza e investigación.

d. Desarrollo de Proyectos Educativos

La esencia del uso de herramientas informática en la educación, lo reiteramos es la actitud de las personas que forman una comunidad de intereses, comparten esfuerzos y beneficios. En Internet podemos encontrar múltiples casos bajo la denominación de proyectos colaborativos. Este concepto supone un sentido de trabajo en equipo, iniciativa y asunción de riesgos, también supone

adquirir la capacidad para organizar la información dispersa proveniente de diversos medios.

Haciendo una recapitulación de lo señalado diremos que estas cuatro dimensiones esbozadas pueden servir para sustituir los aspectos de un proceso enseñanza - aprendizaje tradicional, por otro software donde los Talleres de desarrollo de Software mejoren la calidad del actual software de aprendizaje en la Facultad de Ingeniería Industrial y de Software de la Universidad Nacional de Callao.

Las universidades modernas son los focos más desarrollados de una sociedad. Todo lo que sucede en la vida real se refleja en ella. Ahí se debate, crea, innovan y genera un nuevo conocimiento. Estas instituciones están y deben estar orientadas a resolver los problemas de la vida cotidiana. En los términos más generales deben ser bastiones de desarrollo, democracia, ciencia, tecnología y cultura.

En sus orígenes etimológicos el constructo UNIVERSIDAD procede del latín UNIVERSITAS, teniendo que ver con lo universal, integral, el todo, la suma. De ahí, que toda universidad debe ser sinónimo de pluralidad e interculturalidad. Una suma y expresión de todo lo avanzado de la época, de la historia y de lo heredado por la humanidad.

La universidad contemporánea se yergue en el centro de la sociedad. Es la institución más importante en el complejo proceso del conocimiento y no sólo alberga a las ciencias básicas, sino que también da lugar a revistas, libros, y bases de datos para comunicar el conocimiento en todo el mundo. (Altbach, 2009, pág. 39)

A todo ello debemos sumar, que las universidades como entes vivos y dinámicos tienen que estar en permanente vínculo con la sociedad, la empresa y el Estado. La primera, porque la

universidad tiene la responsabilidad de resolver los grandes problemas de la sociedad, de sus ciudadanos y grupos sociales; la segunda, la empresa, porque la universidad prepara profesionales altamente calificados, para que se desempeñen en las empresas públicas y privadas, con calidad y eficiencia, dando sostenibilidad económica a la sociedad. Y por último, con el estado que regula las actividades económicas sociales de un país, respetando los niveles de autonomía y democracia, promoviendo el bienestar general de sus ciudadanos.

Las nuevas tecnologías y avances científicos han exponenciado a las universidades, haciéndolas más eficientes, incluso se han masificando, siendo diversas las formas de acceder a ellas, teniendo como rol de formar profesionales al servicio de la nación.

Según José Ramón López y Portillo Romano (Romano, 2018) las tecnologías están transformando sustancialmente la vida humana, por ejemplo: La biotecnología y química; Tecnología digital y nanotecnología; Tecnología digital y física óptica; Tecnología digital, neurología y sensores; Tecnología digital, sensores, medicina y farmacología; Física de materiales, química, biología y medicina; Tecnología digital, almacenamiento de energía y fisiología; Neurología, nanotecnología y bioquímica; etcétera. Todas ellas y otras más están dinamizando el ciclo tecnológico en la cual nos desenvolvemos en la búsqueda permanente del conocimiento y su innovación.

Las universidades en el escenario de orden mundial, moderno, han reconfigurado su quehacer y devenir, teniendo entre sus nuevos rasgos los siguientes:

INTERNACIONALIZACIÓN. - La internacionalización es un proceso complejo que se expresa a diferentes niveles y en diferentes actores de la educación superior. En las universidades tiene un carácter dual, por una parte, implica la presencia de la

dimensión internacional al interior de sus funciones y, por otra, implica la proyección internacional de sus capacidades y productos. (Jesús, 2005).

De ahí que hoy se toma en cuenta los estándares de la internacionalización para la educación, sustentada en la informatización, interdisciplinariedad y carácter transversal en cuanto al uso de las Nuevas Tecnologías de Información y Comunicación (NTIC) y los Nuevos Entornos Virtuales de Aprendizaje (NEVA).

Otras de las características de las universidades son:

INTERCONEXIÓN. - Tiene que ver con que las universidades trabajan en torno a sistemas de redes y recursos tecnológicos, siendo empleados para los procesos de enseñanza-aprendizaje. Ello implica la modernidad y optimización de sus procesos, todo ello basado en el uso de las Nuevas Tecnologías de Información y Comunicación (NTIC).

COLABORACIÓN E INTERCAMBIO DE INFORMACIÓN. - Ya no hay ninguna universidad que pueda trabajar de forma aislada, estas necesitan de apoyo sostenido entre ellas, empleando convenios, firmas de cartas de intención, normas, tratados, etcétera.

MOVILIDAD ESTUDIANTIL Y DOCENTE. - Cada vez más se emplean políticas de intercambio estudiantil y docente. Los grados, estudios en otros países a los inicios de su formación, connotan sus conocimientos, los prestigian y empoderan para los futuros desempeños laborales.

COMPETENCIA. - Esto implica que las universidades, cualquiera sea su carácter, público o privado, están inmersas en un mundo regulado por el mercado, en donde las competencias rigen la vida académica, compartiendo los docentes y estudiantes,



experiencias, conocimientos y saberes, así como los materiales e investigaciones que realizan.

En 1958 se funda la Red de Agencias de Proyectos de Investigación Avanzada (ARPANET) dirigida por el Departamento de Defensa de Estados Unidos, esta infraestructura fue empleada con fines militares durante la guerra fría. Su objetivo era reconstruir la información en caso de una guerra nuclear. Favoreció a la creación de la Red Universitaria Americana (RUA), estableciendo relaciones entre universidades, con el fin de definir un protocolo de comunicaciones abierto, multiusuario y diversificado.

El primer lenguaje de programación tuvo fines científicos, con fines bélicos, para calcular la balística y poder impactar al enemigo, en el marco de la segunda guerra mundial. En 1960 se crea un lenguaje con fines comerciales el COBOL. A partir de 1964 se crea el lenguaje de programación BASIC con fines educativos. A partir de entonces las universidades utilizaron a las computadoras para elaboración de materiales educativos, cálculos científicos. Cuando entra el lenguaje de programación BASIC se desarrollan los sistemas operativos, que en ese instante eran almacenados en discos magnéticos (DOS).

El 17 de mayo de 1969 se transmite el primer mensaje a través del ARPANET entre la Universidad de California de los Ángeles y el Instituto de Investigaciones de Stanford (Washington), ese es el primer E-mail que se envió, motivo por el cual se celebra en el mundo el DÍA MUNDIAL DE LAS TELECOMUNICACIONES - Internet.

En enero de 1975 la Revista Popular Electronics, publica los diagramas de la computadora Altair 8800, haciendo que algunos jóvenes se entusiasmen y se reúnan en un garaje para ver la posibilidad de construir esos equipos, de manera conjunta,

asistiendo entre ellos Bill Gates, Steve Wozniak y Steve Jobs, quienes posteriormente desarrollarían las microcomputadoras, fundando así Apple. Frente a eso IBM, que ya construía Minicomputadoras, empieza a fabricar Microcomputadoras, lo mismo hace Matsushita en Japón. La oferta y demanda se expande en el mercado de las pequeñas empresas y en los hogares.

En el 80 se masifica la venta de micro computadoras y nacen los videos juegos. Para poner orden a esa producción, surge el protocolo TCP/IP, que pone una máscara o codificación a la dirección de servidor, a partir de un nombre o interés del usuario, nacen así los primeros dominios, entre ellos la nomenclatura World Wide Web.

En 1981 con la comercialización de las microcomputadoras esta se diversifica y masifica en todos los países, apareciendo los programas especializados y aplicativos para las diversas áreas del conocimiento. Aparece la carrera de diseño Gráfico con el diseño asistido por computadora (CAD), querían crear juegos, extendiéndose también al diseño y la manufactura asistida por computadora (CAM).

Las tecnologías connotan sus espacios. En 1990 el internet se masifica en Europa y Estados Unidos. Aparecen los primeros navegadores que pueden trabajar en múltiples sistemas operativos, en ese momento se democratiza el conocimiento, comienzan las personas a relacionarse con las demás culturas, exponenciando el saber y accediendo de manera libre a la información con el protocolo TCP/IP. En esos años aparece la Educación a Distancia utilizando los sistemas de plataformas virtuales.

Nace el primer navegador Gráfico MOSAIC compatible para Windows y Macintosh (apple), esto revoluciona todo, las personas podían comunicarse, acceder y comprar desde su hogar. Las

ventas de computadora y ofertas de servicios por Internet se exponencian.

En ese contexto nace la primera globalización denominada como las PUNTO COM, las cuales trascendían fronteras y ofrecían sus productos a cualquier parte del mundo: amazon.com; alibaba.com, CNN, etcétera. A partir de allí surgen y mueren empresas, especialmente las que no supieron desarrollar las plataformas de páginas. Creían en los clientes y no había confiabilidad con los clientes, validación, sistema seguro, no existía la Banca por Internet (el home banking). Eso fue un atraso, tampoco estaba normado al respecto de esa dinámica, no había leyes de transacciones digitales. Toda esta industria fue desarrollada por profesionales vinculados a la informática y sistemas. No había interactividad, todo iba a una página fría. Eso termina a mediados del 2000 cuando nace la WEB 2.0.

La WEB 2.0 nace cuando se generaliza los pagos por internet y se comienza a viralizar las redes sociales, las cuales, a diferencia de las listas de interés anteriores, eran foros que relacionaban interés con contactos. Su dinámica era más interactiva, el atractivo de estas redes era que uno se podía vincular con amigos o familiares que no había visto años y te enganchabas a ese sistema. En ese contexto surge el facebook, el twitter, instagram, likending y otros. Con ello se ve el surgimiento del prosumidor (gente que consume lo que produce), los cuales se desarrollan a gran escala. La gente se relaciona según su interés, diversificado.

Otra característica de la WEB 2.0 es que se basan en sistemas automatizados en donde la gente ya no necesita saber programar, ni estar vinculados a los estudios de una carrera de informática. Cualquier persona puede ser comunicadora, difusora o gráfica. Surgen los bots (primitivas inteligencias artificiales a la par con los sistemas automatizados, entre ellos los teléfonos

inteligentes, con aplicaciones sencillas y gratuitas, que la gente empieza a descargar según sus intereses y usos. Esta Web reconoce patrones o tendencias, lo cual va marcando la predominancia en el posicionamiento de búsquedas a google.com.

En este contexto es nacen y se desarrollan los "YouTubers", gente joven e ingeniosa que aprovechan el interés de temas diversos, desarrollando producciones audiovisuales atractivas e innovadoras. Estas se financian por el número de suscriptores y vistas del video que YouTube paga por eso. Inicialmente son de orden humorístico, pero después se vuelven más completas y sensacionales.

En los documentos de "La Conferencia Mundial sobre la Educación Superior" (1998), tratadas por la UNESCO, y en las Comisiones de Seguimiento de dicha Conferencia, se hicieron sugerencias valiosas de cómo afrontar los desafíos más urgentes, en cuanto al uso de las tecnologías, como por ejemplo: la actualización permanente de los profesores, de los contenidos y del currículo; la introducción de redes electrónicas para el aprendizaje; traducción y adaptación de las principales contribuciones científicas; modernización de los sistemas de gestión y dirección; e integración y complementación de la educación pública y privada así como de la educación formal, informal y a distancia.

Estas propuestas son retos para los profesores dado sus múltiples labores que tienen que abordar, esto en cuanto a la calidad de estos educadores, especialmente en cuanto al uso y manejo de las nuevas tecnologías de comunicación e información. Sea como fuere, aunque las reformas educativas son constantes en los países latinoamericanos, los resultados siguen siendo decepcionantes (Pérez-Díaz, 2005).

No debemos desconocer que Latinoamérica presenta necesidades educativas muy amplias, incluso primarias, que abarcan desde la implementación de electricidad, tecnologías e infraestructura, pero las nuevas tecnologías apertura también nuevas propuestas y oportunidades para un desarrollo integral de los grupos humanos, cualquiera sea la geografía o el lugar en donde se encuentren.

En América Latina se está profundizando en el último tiempo en experiencias basadas por un lado en el aprendizaje cooperativo, y por otro, en la cooperación institucional para compartir los recursos de información a través de las ciber bibliotecas, para la utilización de plataformas tecnológicas comunes y centros tecnológicos regionales hasta llegar a la Gráfico integral del consorcio (Mena, 2004: 33).

Se debe reconocer que el uso desenfrenado de la tecnología puede acarrear nuevos y viejos problemas, por lo que no debemos olvidar el factor esencia de la persona humana. Todo lo contrario, creemos que hay que recuperar el enfoque de la ilustración, en el cual los valores de la dignidad humana están en íntima conexión con el desarrollo de la racionalidad científica y tecnológica, no en contradicción con ella. (Quintanilla, 2005, pág. 26).

Debemos indicar también que el Internet es un sistema informático de redes, espacios que han creados los individuos, brinda poder a quienes lo usan, permite difundir la información y los sucesos, fomentando la innovación y creatividad, incluso recreando la vida de sus usuarios. Es una herramienta esencial para el logro educativo, todo indica que a futuro dominará ese sistema masificado, gratuito y de interacción de masas. El cual ya no fue cableado, sino inalámbrico, omnipresente, un soporte fundamental para la educación masiva virtual.

De la Sociedad de la Información a la Sociedad del Conocimiento

En la Cumbre Mundial sobre la Sociedad de la Información (CMSI), realizada en Ginebra 2003-Túnez 2005, se estableció la Declaración de Principios para el Mundo, reducido en la siguiente frase: «Construir la Sociedad de la Información: un desafío global para el nuevo milenio» (Naciones Unidas, 2003-2005).

Este desafío se hace más necesario de impulsar en países como el Perú, en donde las carencias económicas, la geografía, el clima, limitaciones de infraestructuras y atenciones de servicios sanitarios y básicos, no los han alcanzado. La carencia de tecnologías nos aleja, limita el desarrollo de los individuos.

Es evidente, que, de la Sociedad del Conocimiento establecida por la ONU, se ha pasado a la Sociedad de los Conceptos y luego a la Sociedad del Aprendizaje en la cual nos encontramos actualmente.

Stephen Downes sustenta de manera clara sobre el aprendizaje en general, haciendo diferencias y distinguiendo los tipos de aprendizaje, lo cual tiene que ver con redes, comunidades, sociedad, participación o contenidos. Ello implica lo que pasamos a ser y hacer con el conocimiento, o como un mecanismo de lo que se produce o consume. Eso es necesario distinguir en cuanto a educación bajo los modernos soportes tecnológicos.

Downes es claro en plantear que el Aprendizaje nos puede hacer buenas personas, nos posibilita trabajar hacia el crecimiento de la persona como individuo. El aprendizaje abierto, compartido, contribuye en ello, diferente del aprendizaje ligado y orientado solo al consumismo.

Debemos trabajar el Aprendizaje de redes, esto implica ser más productor que consumidor. Podemos dar significancia a nuestras vidas. ¿Por qué las redes pueden dar significancia a nuestras vidas? Por lo siguiente:



- Fortalecen nuestra AUTONOMÍA, nos permite hallar nuevos y mejores caminos. Lograr y alcanzar nuestras metas, sueños y proyectos de vida.
- Amplia y contribuye con nuestra DIVERSIDAD, esto contribuye a ampliar los puntos de vista diversos, a comprendernos, entendernos e integrarnos como sociedad, grupos sociales y culturales, respetando la diversidad y creando puentes de entendimiento entre la especie humana.
- Promueve la APERTURA en nosotros, la capacidad de recibir experiencias, de compartir lo que tenemos y sabemos, apoyándonos en la experimentación y creatividad compartida. Rompe con el egoísmo y egocentrismo que tanto daño hace para el desarrollo conjunto de las personas.
- Se construye e impulsa la INTERACTIVIDAD, promueva la idea, el conocimiento diverso, el conocimiento de la persona, de lo que hacemos y somos; el conocimiento de la sociedad de sus avances y logros; todo ello creado en las interacciones infinitas y diversas que podamos hacer.

Esto y mucho más sienta las bases de nuestros principios y nueva forma de comprender la importancia que brinda las redes y tecnologías, es decir, su efectividad e impacto social y personal en quienes asumen esta nueva visión conectivista; es decir, darles un valor a las tecnologías, y no posibilitando que estas nos destruyan como sociedad.

Todo indica que las modernas tecnologías y sus aceleradas innovaciones pueden y deben convertirse en herramientas fundamentales para el cambio, la educación y desarrollo de los pueblos, especialmente para los sectores más olvidados. Ese es el norte y el compromiso que toda institución educativa deba tener, no solo de parte de los educadores y pedagogos, sino también de las instituciones públicas y privadas, para disminuir la pobreza, el atraso, la ignorancia, incluso con la violencia.

No debemos olvidar que la tecnología es expresión del desarrollo científico de una sociedad, son inherentes, no están separados. Su avance, descubrimiento e innovación se da en ese devenir. Es decir, la tecnología y ciencia para alcanzar sus máximos niveles debe tener como columna vertebral la dinámica estructural de una sociedad, en donde las universidades responden a ello y a las relaciones vinculantes y de dependencia a nivel internacional.

Metodología a utilizar para el desarrollo del software

El considerar que un proyecto de desarrollo de software podía abordarse como cualquier otro ha llevado a una serie de problemas

El identificar dichos problemas y conocer sus causas es el único método que nos puede llevar hacia su solución.

La ingeniería del software tiene tres elementos clave: métodos, herramientas y procedimientos

Los métodos indican cómo construir técnicamente el software, y abarcan una amplia serie de tareas que incluyen:

- a) La planificación y estimación de proyectos,
- b) El análisis de requisitos
- c) El diseño de estructuras de datos
- d) Programas y procedimientos
- e) La codificación
- f) Las pruebas
- g) El mantenimiento.

Las herramientas proporcionan un soporte automático o semiautomático para utilizar los métodos, se denominan CASE (Computer Assisted Software Engineering).

Los procedimientos definen la secuencia en que se aplican los métodos, los documentos que se requieren, los controles que

permiten asegurar la calidad y las directrices de evaluación de progresos.

Clasificación de software

Software de sistema. Cuyo propósito es facilitar la ejecución de otro software.

Entran en esta categoría:

- Sistema operativo.
- Compiladores

Software de aplicación. Automatiza un sistema de información para un fin concreto. Entran en esta categoría:

- Procesador de texto: Word
- Hoja de Calculo: Excel
- Gestores de base de datos: Access
- Aplicaciones Web

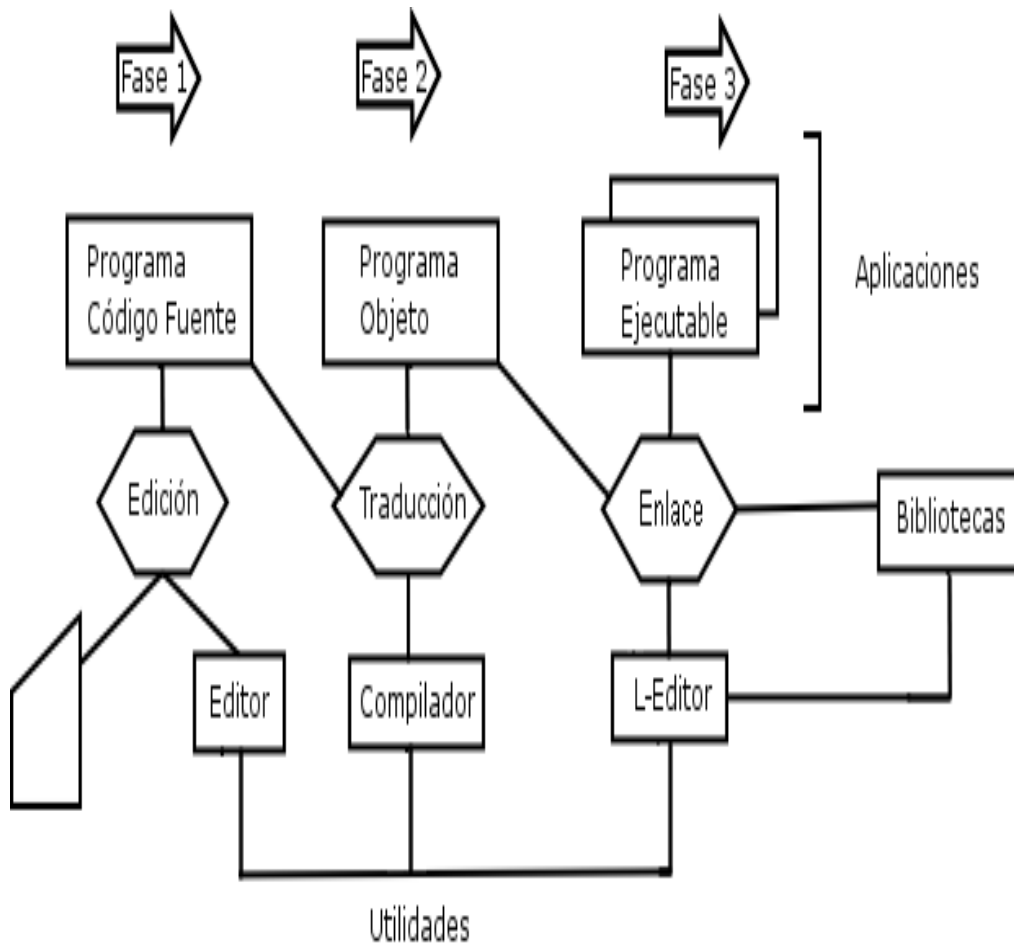
Formas de software

Código Fuente: escrito por programadores. Contiene el conjunto de instrucciones destinadas a las computadoras.

Código Objeto: resulta usando un *compilador* sobre el código fuente. Consiste en una traducción de este último. No es directamente inteligible por el ser humano, pero tampoco es directamente entendible por la computadora.

Código Ejecutable: resulta de enlazar uno o varios fragmentos de código objeto. Constituye un archivo binario con un formato tal que el sistema operativo es capaz de cargarlo en la memoria de un ordenador y proceder a su ejecución. Es directamente inteligible por la computadora.

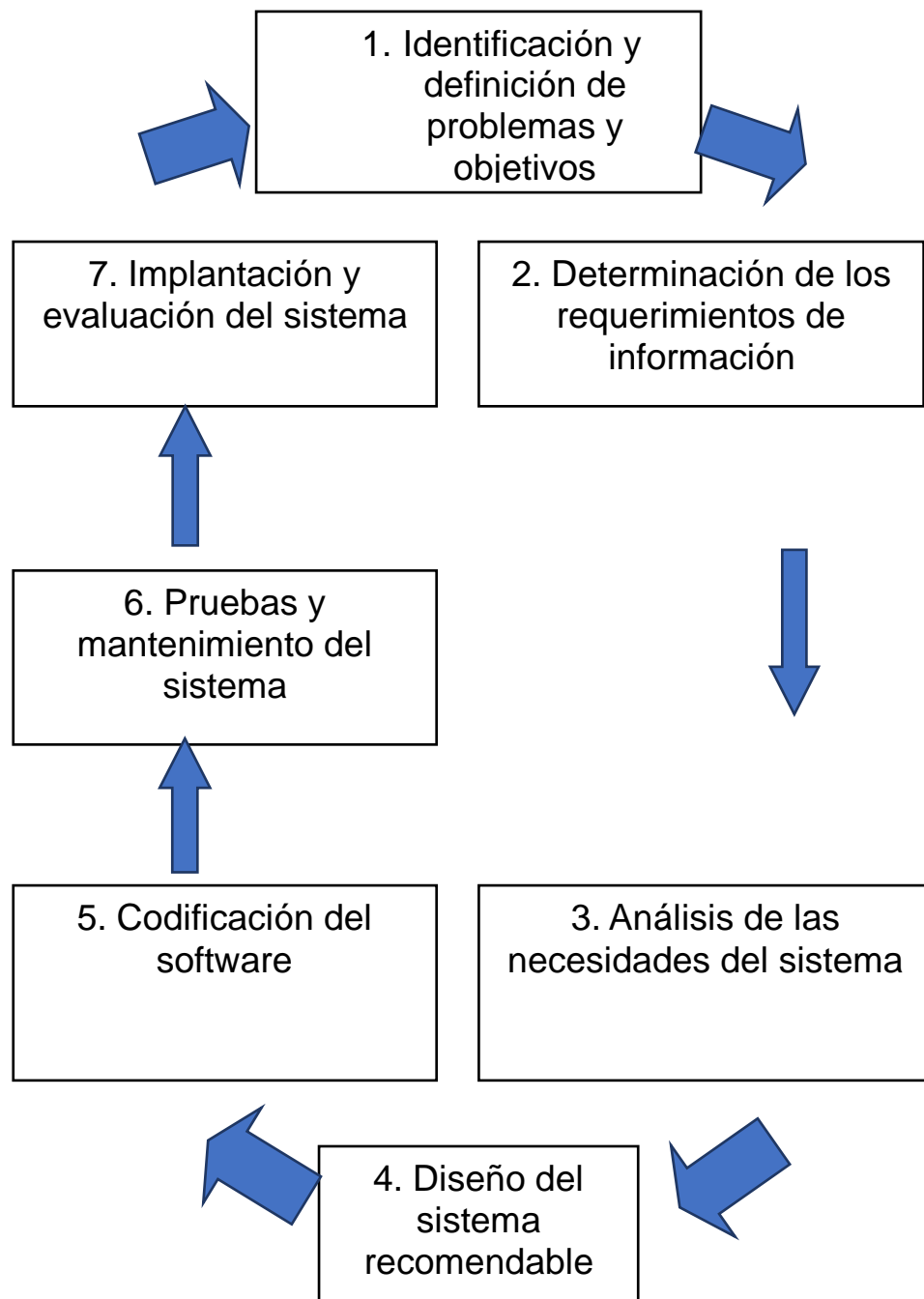
Gráfico 2 Formas de Software



Debe contemplar la elección del hardware, la configuración del software de base y las aplicaciones de software que permitan cubrir las necesidades de información del SIG.

La obtención del software adecuado constituye una gran parte del esfuerzo de automatización del sistema, ya que, aunque cada vez existe una mayor cantidad de software eficaces que se pueden comprar listos para instalar, la mayoría de las empresas requieren la construcción de software a medida, o al menos, la adaptación de aplicaciones existentes. Se habla entonces de desarrollo de software para el SIG.

Gráfico 3 Fases del desarrollo del Software



El proceso de desarrollo de software es materia de la Ingeniería del Software. Involucra diversas tareas:

- Análisis
- Diseño (produce un prototipo)
- Codificación
- Pruebas (unitarias e integradas)
- Instalación
- Administración (evaluación continua)

Intervienen muchas personas

El cliente: Es el que tiene el problema en su empresa y desea que sea solucionado.

Analista de Sistema: Es el encargado de hacerle llegar todos los requerimientos y necesidades que tiene el cliente a los programadores.

Programadores: Son las personas encargadas de realizar lo que es la codificación y diseño del sistema para después probarlo y lo instalan al cliente.

Papel del Analista de Sistemas

Como consultor:

Canaliza tópicos de la informática

Contribuye con ideas frescas sobre la empresa

Implanta metodologías para analizar y diseñar SI.

Como especialista de apoyo:

Contribuye en el Dpto. de sistemas

Dispone de experiencia profesional respecto al Hardware y Software y a sus aplicaciones en la empresa.

Como agente de cambio:

Confiere responsabilidad en las actividades del ciclo de desarrollo de sistemas

Se define como una persona que cataliza el cambio, desarrollando planes de modificar y/o diseñar nuevos sistemas.

Analista de información:

Conduce estudios de sistemas para detectar hechos relevantes relacionados con la actividad de la empresa. La función más importante en este caso es reunir información y determinar los requerimientos.

Diseñadores de sistemas, diseñadores de aplicaciones: desarrolla el estudio completo de los sistemas, tiene como responsabilidad diseñar el nuevo sistema.

Analista programador: conduce la investigación de sistemas, desarrolla las especificaciones de diseño y escribe el software necesario para implantar el diseño.

Desarrollo de software

Primer. Paso: es el análisis

Aquí el analista se pone en contacto con la empresa para ver cómo está conformada, a que se dedica, saber todas las actividades que realiza.

Según los requerimientos, se empieza a realizar el diagrama relacional todo debe de llevar una secuencia lógica de las

actividades, todo esto se realiza de manera manual para ver cómo fue su diseño lógico y diseño de pantallas.

Segundo paso: es el diseño

Aquí entra todo el diseño del sistema; es decir las pantallas, base de datos, todo esto debe de cumplir con ciertos estándares los cuales se toman en cuenta para poder desarrollar el diseño con calidad y así poder ofrecer un diseño amigable en cuestión de colores, tamaños de botones, etc.

Tercer. Paso: es la codificación,

Aquí se desarrolla todo el código del sistema por parte del programador, que tiene sus bases o formas para realizarlo, ofreciendo funcionalidad al sistema, de acuerdo a las especificaciones del cliente.

Cuarto paso: son las pruebas

Aquí al sistema se pone a prueba, para saber cuáles son los posibles errores y con ello mejorarlo para tener mejor calidad.

Prueba de unidad

Aquí se prueban los caminos de control importantes, con el fin de descubrir errores dentro del ámbito de un módulo.

Prueba de integración

Aquí se llevan a cabo pruebas para detectar errores asociados con la integración de la estructura del programa.

Prueba de validación y verificación

Es el conjunto de actividades que aseguran que el software implementa correctamente una función específica se denomina Verificación. La Validación se refiere a un conjunto diferente de actividades que aseguran que el software construido se ajusta a los requisitos y necesidades del cliente.

Prueba del sistema

Sirve para verificar que se hayan integrado correctamente cada uno de los elementos del sistema.

Prueba de Recuperación

Es una prueba que se hace al sistema forzando a que produzca fallas de software de muchas maneras y verificando que la recuperación se lleve a cabo, ya sea automáticamente o manual.

Prueba de Seguridad

Verifica la aplicación de los mecanismos de protección incorporados en el sistema.

Prueba de Resistencia

Está diseñada para enfrentar a los programas en situaciones anormales, es decir ejecutar el sistema en forma que demande recursos en cantidad, frecuencia o volúmenes anormales.

Depuración

Es el resultado de una prueba efectiva, es decir, cuando en un caso de prueba se encuentra un error, la depuración es el proceso de eliminación de un error.

Análisis de los sistemas

Es un método, plan o procedimiento de clasificación para hacer algo (¿QUE?)

- Es un conjunto de procedimientos o programas relacionados de manera que juntos forman una sola unidad.
- *Es un conjunto de hechos, principios y reglas clasificadas y dispuestas de manera ordenada* mostrando un plan lógico en la unión de las partes.

Es un conjunto o arreglos de elementos para realizar un objetivo predefinido en el procesamiento de la información.

Es conocer el sistema para:

- Determinar los objetivos y límites,
- Caracterizar su estructura y funcionamiento,
- Marcar las directrices que permitan alcanzar los objetivos propuestos y
- Evaluar sus consecuencias.

Principios

Debe presentarse y entenderse el dominio de la información de un problema.

Debe definirse las funciones que debe realizar el Software.

Debe representarse el comportamiento del Software a consecuencias de acontecimientos externos.

Debe dividirse en forma jerárquica los modelos que representan la información, funciones y comportamientos.

Función:

Es dar soporte a las actividades de un negocio para generar beneficios.

Dependiendo de los objetivos del análisis podemos estar ante dos problemáticas:

Análisis de un sistema ya existente.

Análisis como paso previo al diseño de un nuevo sistema.

Elementos de los sistemas

Software

Hardware

Personal

Base de datos

Procedimientos

Una empresa de Consultoría de Sistemas cubre toda el Ciclo de Vida de Software, desde la toma de requerimientos y diseño funcional hasta la construcción, pruebas de aceptación e implantación; Por tanto, los skills de una factoría de software son claramente diferentes: requieren de un menor conocimiento de la necesidades del futuro usuario de negocio, existe más flexibilidad respecto a la ubicación geográfica de la factoría (al no requerir cercanía con el futuro usuario del sistema a fin de definir el requerimiento) y se especializan para conseguir calidad de software a menor coste, automatizando e industrializando los procesos de desarrollo de software donde aplique.

Mientras que la **fábrica de software** inicia su tarea a partir de un diseño funcional que "otros" han realizado. Tiene ingresos adicionales por los servicios asociados que brinda a los clientes a los que les desarrolla las aplicaciones informáticas, tales como el

mantenimiento, la capacitación, la actualización, el despliegue, el soporte, etc. Una fábrica de software de una fábrica de productos de software, dado que esta última se dedica al desarrollo de software para la generación de productos propios basados en aplicaciones informáticas cuya propiedad intelectual le pertenece. Es necesario también tener en cuenta que al emprender una fábrica de software se debe capacitar muy bien al personal que va a trabajar para ser más eficientes y contribuyan al desarrollo continuo y avanzado en el transcurrir del tiempo y de las necesidades.

Desarrollo de software en los talleres

El desarrollo de software es un conjunto o disposición de procedimientos o programas relacionados de manera que juntos forman una sola unidad.

Otros autores definen como un conjunto de hechos, principios y reglas clasificadas y dispuestas de manera ordenada mostrando un plan lógico en la unión de las partes. También es un conjunto o arreglo de elementos para realizar un objetivo predefinido en el procesamiento de la Información.

La ingeniería del software tiene siete pasos:

- La planificación y estimación de proyectos,
- El análisis de requisitos
- El diseño de estructuras de datos
- Programas y procedimientos
- La codificación
- Las pruebas

- El mantenimiento.

Las herramientas proporcionan un soporte automático o semiautomático para utilizar los métodos, se denominan **CASE** (ComputerAssisted Software Engineering).

Los procedimientos definen la secuencia en que se aplican los métodos, los documentos que se requieren, los controles que permiten asegurar la calidad y las directrices de evaluación de progresos.

El desarrollo de software tiene dos tareas fundamentales el análisis y el diseño. El análisis es conocer la situación actual mientras que el diseño de software es el proceso de aplicar ciertas técnicas y principios con el propósito de definir un dispositivo, un proceso o un Sistema, con suficientes detalles como para permitir su interpretación y realización física.

El Diseño de software debe contemplar los siguientes aspectos:

El diseño de los datos.

Trasforma el modelo de dominio de la información, creado durante el análisis, en las estructuras de datos necesarios para implementar el Software.

El Diseño Arquitectónico.

Define la relación entre cada uno de los elementos estructurales del programa.

El Diseño de la Interfaz.

Describe como se comunica el Software consigo mismo, con los sistemas que operan junto con él y con los operadores y usuarios que lo emplean.

El Diseño de procedimientos.

Esto se lleva a cabo teniendo en cuenta ciertos principios:

- Debe presentarse y entenderse el dominio de la información de un problema.
- Defina las funciones que debe realizar el Software.
- Represente el comportamiento del software a consecuencias de acontecimientos externos.
- Divida en forma jerárquica los modelos que representan la información, funciones y comportamiento

El proceso debe partir desde la información esencial hasta el detalle de la Implementación

Un Análisis de Sistema se lleva a cabo teniendo en cuenta los siguientes objetivos en mente:

- Identifique las necesidades del Cliente.
- Evalúe que conceptos tiene el cliente del sistema para establecer su viabilidad.
- Realice un Análisis Técnico y económico.
- Asigne funciones al Hardware, Software, personal, base de datos, y otros elementos del Sistema.

- Establezca las restricciones de presupuestos y planificación temporal.
- Cree una definición del sistema que forme el fundamento de todo el trabajo de Ingeniería.

Identificación de problema.

Consiste en identificar el problema que se solucionará con el desarrollo del software en mención.

Especificaciones para solución del problema

Para solucionar el problema diagnosticado se desarrolla una aplicación informática que contemple módulos:

Estudio de viabilidad

La viabilidad y el análisis de riesgos están relacionados de muchas maneras, si el riesgo del proyecto es alto, la viabilidad de producir software de calidad se reduce.

Cuatro Áreas Principales de Interés:

Viabilidad Económica

Viabilidad Técnica

Viabilidad Legal

Viabilidad del Mercado

Viabilidad Económica. Es una evaluación de los costos de desarrollo, comparados con los ingresos netos o beneficios obtenidos del Sistema desarrollado.

Viabilidad Técnica. Es un estudio de los principios técnicos del sistema, las funciones, rendimiento, mantenimiento, productividad y restricciones.

Viabilidad Legal. Es determinar cualquier posibilidad de infracción, violación o responsabilidad legal en que se podría incurrir al desarrollar el Sistema.

En esta parte se verá si el desarrollo de la aplicación es viable técnicamente ya que existe el soporte teórico para desarrollar los procedimientos que se requieren, en cuanto a las herramientas de desarrollo seleccionadas, permitirán realizar los módulos de interfaz con los usuarios.

Herramientas de desarrollo

Las herramientas disponibles que se puede usar son

Microsoft Visual Basic 6.0: Para la creación de la interfaz del usuario.

Microsoft SQL Server: Como motor de base de datos

Microsoft Internet Información Server: Como servidor web

Tecnología ASP: Para interactuar con el Motor de Base de Datos, Generar preguntas automáticamente y para calificar las evaluaciones tomadas por los estudiantes

HTML: para estructurar la información que se presentará en las páginas

Tecnología ADO: Para el acceso, sincronización e interacción con el Motor de Base de datos

Modelado de arquitectura

El modelo debe representar todas las funciones y sub funciones de un Sistema. El modelo puede incluir notación gráfica, información y comportamiento del Sistema. Todos los sistemas pueden modelarse como transformación de la información empleando una arquitectura del tipo entrada y salida.

Especificaciones del sistema

Es un documento que sirve como fundamento para la ingeniería Hardware, Software, base de datos e Ingeniería Humana.

Describe la función y rendimiento de un sistema basado en computadoras y las dificultades que estarán presentes durante su desarrollo.

Tareas que constituyen el análisis

Conceptualización

Consiste en obtener una visión de muy alto nivel del sistema, identificando sus elementos básicos y las relaciones de éstos entre sí y con el entorno.

Análisis funcional

Describe las acciones o transformaciones en el sistema.

Análisis de condiciones

Debe reflejar todas aquellas limitaciones impuestas al sistema que restringen el margen de las soluciones posibles.

Construcción de modelos

Se debe construir un prototipo (modelo definitivo) del mismo.

Validación del análisis

El análisis debe ser consistente y completo. Si el análisis se plantea como un paso previo para realizar un diseño, habrá que comprobar además que los objetivos propuestos son correctos y realizables.

Gráfico 4 Ciclo de Vida de un software



Fuente: Propia

Modelado del software con UML

Para el análisis y diseño se podrá utilizar el lenguaje de modelado UML, Unified Modeling Language (Lenguaje Unificado de Modelado) por sus siglas en inglés. UML, es una herramienta que permite modelar software orientado a objetos a través de un amplio vocabulario Gráfico enfocado a la representación conceptual y física de los sistemas de software. Actualmente es un estándar adoptado por el Grupo de Desarrollo de Objetos OMG (Object Management Group). [Abud, M., 2002]

El UML es un lenguaje estándar que puede usarse para escribir especificaciones de software, para visualizar, especificar, construir y documentar las entidades que forman un sistema de software.

UML permite modelar sistemas que van desde sistemas de información para la empresa, hasta aplicaciones distribuidas basadas en el web, de ahí que hemos escogido dicho lenguaje de modelado para el diseño del prototipo.

Diagrama de casos

Los diagramas de casos de uso muestran la funcionalidad del sistema desde la perspectiva que tienen los usuarios y lo que el sistema debe de hacer para satisfacer los requisitos propuestos. Pueden mostrar el comportamiento de un sistema completo o, de una parte.

Los elementos básicos que se utilizan son:

1. Actores: Son los diferentes usuarios y el papel que representan dentro del sistema.
2. Caso de uso: Representan todo lo que el usuario puede realizar dentro del sistema.
3. Relaciones: Para asociar los elementos anteriores

Diagrama de datos físicos (entidad relación)

El modelo físico de datos, permite definir la estructura de la base de datos, desde el punto de vista de su implementación física. Es decir, que su implementación toma lugar en los recursos físicos, tales como: el DBMS, estructuras de

almacenamiento de datos y software, para describir la estructura de la base de datos.

Diseño de la interfaz de usuario

Para los módulos de Generación de Preguntas y Definición de Evaluaciones se utiliza una Interfaz con las siguientes características:

- Interfaz de múltiples documentos (MDI)
- Formularios de Tipo Explorador
- Formularios y objetos de formularios Estándar de Visual Basic 6.0

En el módulo de Presentación y Evaluación de Test se utiliza páginas web dinámicas en donde se tomó en cuenta las siguientes características

- Se optó por no utilizar imágenes en las páginas que despliegan las preguntas ya que estas páginas tienen que ser bastante livianas, además en una aplicación de este tema hay que tener en cuenta la velocidad y sencillez de la página para que el estudiante no tenga distracciones.
- Todos los elementos de una página, sean elementos de formularios o no, estarán alineados a una tabla.
- El tamaño de los campos de texto debe estar de acuerdo con la extensión máxima que permiten los mismos.

- En las páginas de visualización de preguntas se debe visualizar el nombre del alumno que rinde la prueba y un reloj que controle el tiempo que se demora en contestar una pregunta o una evaluación.

2.3 Marco conceptual

2.3.1 Talleres de desarrollo de Software

Son centros de entrenamiento que utiliza recursos informáticos y de telecomunicaciones¹. En la investigación, esta técnica se utiliza para realizar experimentos y entender el comportamiento del software productivo y financiero. Son medios tecnológicos que utilizan equipos y programas para el análisis diseño e implantación de programas, sistemas, páginas web, etc. y entender el comportamiento de los softwares productivos, software de información y software financieros para organizaciones.

Incluye: Software (programas), hardware (computadoras y equipos), Internet, base de datos, y Procedimientos.

El uso de estos Talleres, permite que los alumnos tengan la experiencia de los casos reales, logrando así las capacidades de las principales asignaturas de la especialidad de Ingeniería de Sistemas.

2.3.2 Formación profesional

Es el logro de las competencias de las asignaturas tanto en la teórica como en la práctica, de manera permanente, para que cuando egrese, pueda ejercer su profesión.

Proceso que facilita la transformación permanente del pensamiento, las actitudes y comportamiento de los estudiantes. La enseñanza es un proceso mediante el cual el

¹ Alberto Prieto, Antonio Lloris, Juan Carlos Flores, Introducción a la Informática, 2da. Edición, 1995, McGraw-Hill Interamericana de España, Madrid, Paginas 505-506...

profesor selecciona un material que debe ser aprendido y realiza una serie de operaciones para que el estudiante adquiera esos conocimientos. La enseñanza es el uso de técnicas que ayudan al estudiante en su esfuerzo por el aprendizaje. El aprendizaje es el acto por el cual un estudiante intenta captar y elaborar los contenidos expuestos por el profesor o por cualquier otra fuente de información. El Alumno aprende a través de medios (técnicas de estudio o de trabajo intelectual). El proceso aprendizaje se realiza en función de unos objetivos, que pueden o no identificarse con los del profesor y lo lleva a cabo dentro de un determinado contexto.

Es un proceso mediante el cual el profesor selecciona los temas y realiza una serie de operaciones para que el estudiante adquiera habilidades y destrezas

El estudiante por su parte intenta captar los contenidos expuestos por el profesor o por cualquier otra fuente de información. Para ello el estudiante utiliza medios, técnicas de estudio o de trabajo intelectual. Este proceso se realiza en función de unos objetivos, que pueden o no identificarse con el profesor y se lleva a cabo dentro de un determinado contexto.

2.4 Definición de términos básicos:

Actitud	Conducta personal que se expone en trabajos de equipos y eventos sociales.
Acuerdo de Nivel de Servicio	Los compromisos asumidos como parte de la calidad, que piensa cumplir con los clientes, especifican las acciones que son necesarias si no se cumple el compromiso
Arquitectura	Grupo de estándares que brindan un contexto de referencia obligatorio para definir la construcción de un nuevo producto de software, permitiendo a los desarrolladores, analistas y todo el conjunto de programadores del software compartir una única línea de trabajo
Aptitud	Capacidad de discernir, operar y proyectarse.
Capacidad	Cuantifica la disponibilidad y empleabilidad de los recursos que cuenta una fábrica de software para su utilización óptima y rentable.
Capacitar	Formar, preparar, hacer apto a uno para realizar algo.
Competencia	Capacidad crítica que demuestran los alumnos en el desarrollo óptimo de sus actividades académicas.
Desarrollo	Es el proceso manual o automatizado de traducir a lenguaje de la computadora lo establecido en el análisis y diseño de un sistema como alcance, aplica técnicas y herramientas para el diseño de prototipo del

	sistema
Destreza	Habilidad de un operario.
Eficiencia	Hacer las cosas con el menor uso de recursos, es decir, optimizándolos.
Eficacia	Lograr o alcanzar los objetivos que se han planeado.
Fábrica	Es un espacio conformado por una infraestructura, herramientas y empleados encargados de la elaboración de productos determinados, convirtiendo materia prima en un producto de acuerdo a la demanda del mercadeo, estos productos son colocados en un mercado para ser adquiridos por muchos clientes; utiliza procesos que se puede realizar a través de máquinas industrializadas o algún trabajo manual.
Framework	Son marcos de trabajo de importancia clave para la construcción de sistemas de software orientados a objetos a gran escala, mayor productividad y menor tiempo de lanzamiento al mercado a través del diseño y la reutilización de código. 57
Fiabilidad	Probabilidad de un buen funcionamiento de la plataforma.
Habilidad	Capacidad y disposición para una cosa.
Información	Es un conjunto de datos, ya procesados y están ordenados para su lectura y comprensión, aporta conocimientos a un individuo o a un sistema acerca de una

materia en específico.

Ingeniería de software	Es la puesta en práctica de los conocimientos científicos en el análisis, diseño y construcción de programas para computadora con la documentación asociada necesaria para operar, producir software y mantenerlos en el tiempo.
Implementar	Es proveer de medios, recursos y métodos antes de ejecutar un proceso.
Modelo	Es un prototipo referente, diseñado en base a planos y esquemas contiene la forma y los pasos para crear productos de las mismas características. Una fábrica de software utilizará un modelo para la producción de software.
Proceso	Es la sucesión de acciones realizadas en un orden establecido en el tiempo, que llevan a la obtención de un objetivo.
Productividad	Es la cantidad producida teniendo en cuenta el trabajo efectuado o el capital invertido.
Proyecto	Es la intención que se analiza y se desea realizar y para lo cual se establece un modo determinado (esfuerzo) y un conjunto de medios necesarios (recursos).
Satisfacción	Es una reacción subjetiva de cumplimiento con los requisitos establecidos en fabricación de un producto.

Simulador	Software que realiza o ayuda a realizar una simulación, por lo general se trata de una réplica de tamaño real o modelo de una situación de control, por ejemplo: la cabina de un avión, las transacciones bursátiles de una Agencia de Bolsa, la sala de control de una Planta Nuclear.
Software	Hace referencia a programas para una computadora, que incluye datos, procedimientos, reglas, con documentación asociada a un sistema; un producto de software se ha construido para un usuario específico.
Sistemas	Es el conjunto de piezas que interactúan entre sí para obtener un objetivo común, pudiendo ser cada pieza un sistema que recibe y entrega productos.
Tiempo	Es la magnitud física fundamental que se utiliza en indicadores con el fin de medir un proceso en específico; en una fábrica de software se gestiona el tiempo.

III HIPÓTESIS Y VARIABLES

3.1 Hipótesis

3.1.1 Hipótesis General:

El Uso de Talleres de desarrollo de Software influye significativamente en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao en el 2017 – 2018

3.1.2 Hipótesis Específicas:

- a) La aplicación de tecnologías de información en los Talleres de desarrollo de Software influye positivamente en el rendimiento académico de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao
- b) La gestión en los Talleres de desarrollo de Software, influye significativamente en la calidad de aprendizaje de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao.

3.2 Definición conceptual de las Variables:

Variable Independiente

(X): Talleres de desarrollo de software

Variable Dependiente

(Y): Formación profesional

3.2.1 Operacionalización de variables

Tabla 1 OPERACIONALIZACIÓN

Variables	Definición teórica	Definición operacional	Dimensi ones	Indicadores	Unidades
Variable Independiente X Talleres de desarrollo de software	Son centros de entrenamiento que utiliza recursos informáticos y de telecomunicaciones	Técnica para realizar experimentos y entender el comportamiento del software productivo y financiero.	X1. Tecnologías de información	- Tipo de computadoras - Internet - Simuladores - Programas de Simulación	Número Nivel Número Número
			X2. Gestión	- Planeación - Organización - Dirección - Control	Grado Grado Grado Grado
Variable Dependiente Y Formación profesional	Proceso que facilita la transformación permanente del pensamiento, las actitudes y comportamiento de los estudiantes	La enseñanza es el uso de técnicas que ayudan al estudiante en su esfuerzo por el aprendizaje. El aprendizaje es el acto por el cual un estudiante intenta captar y elaborar los contenidos expuestos por el profesor o por cualquier otra fuente de información.	Y1. Rendimiento	- Notas	Número
			Y2. Calidad de aprendizaje	-Relevancia -Reflexión -Interactividad -Apoyo de los tutores -Apoyo de los compañeros -Interpretación	Porcentaje Porcentaje Porcentaje Porcentaje Porcentaje Porcentaje

Fuente: Propia

IV: DISEÑO METODOLÓGICO

4.1 Tipo y diseño de la investigación

Por el tipo de investigación, el presente estudio reúne las condiciones metodológicas de una investigación aplicada. De acuerdo a la naturaleza del estudio de la investigación, reúne por su nivel las características de un estudio correlacional.

Se tomó la muestra en la cual los operadores de la función estadística se detallan de la siguiente manera:

$$\left. \begin{array}{ccc} G_1 & O X_1 & O Y_1 \\ & & Y_1 \end{array} \right\} \cong \neq Y_2$$
$$\left. \begin{array}{ccc} & G_2 & O X_2 & O Y_2 \\ & & & Y_2 \end{array} \right\}$$

Dónde:

G: Grupo de estudio

G₁: Estudiantes del IV y V del grupo experimental.
(Participan en la fábrica de Software)

G₂: Estudiantes del IV y V del grupo de control.
(No participan en la fábrica de Software)

O: Observación o medición de la variable

X: Variable independiente

X₁: Uso de Fábrica de desarrollo de Software

X₂: Uso del Método tradicional.

Y: Variable dependiente

Y₁: Formación profesional del grupo 1.

Y₂: Formación profesional del grupo 2

$\cong \neq$:Comparación

4.2 Método de investigación

De acuerdo a los objetivos del estudio el método de investigación es deductivo. Porque al analizar el caso particular de la Universidad Nacional del Callao, podemos generalizar hacia otras universidades.

4.3. Población y muestra

4.3.1 Población:

La población en estudio fue 120 estudiantes del IV y V ciclo de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao que están matriculados en el Semestre académico 2017 - 2018.

4.3.2 Determinación de la Muestra:

La muestra estará conformada por 92 estudiantes del IV y V ciclo de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao que están matriculados en el Semestre académico 2017 - 2018. La muestra fue dividida en dos grupos de manera aleatoria. En uno de los grupos se aplicó el programa de fábrica de software mientras que en el otro grupo seguirá la enseñanza tradicional.

Muestreo

El muestreo es no probabilístico intencionado, porque responde a la naturaleza de la investigación. Para el tamaño de la muestra se utilizó la fórmula de muestreo aleatorio simple, a fin de estimar las proporciones correspondientes.

Fórmula:

$$n = \frac{N \cdot Z_{\alpha}^2 \cdot p \cdot q}{d^2 \cdot (N - 1) + Z_{\alpha}^2 \cdot p \cdot q}$$

α = Nivel de significación = 5%

Z = Valor crítico de la estadística de prueba para el nivel de confianza del 95 %.

Z= 1,96

p = Proporción (prevalencia) de la variable.

De literatura, Prueba Piloto o maximizar con p = 0,5.

q = Proporción del complemento de la variable.

q = 1 – p = 0,5

d = Precisión o margen de error que depende del Investigador, Costo y tiempo. d= 5%.

n = Tamaño óptimo de la muestra.

N = Tamaño de la Población.

Reemplazando:

4.4. Lugar de estudio y periodos desarrollado

La investigación se realizó en la Facultad de Ingeniería Industrial y de sistemas de la Universidad Nacional del Callao. El periodo desarrollado fue entre 2017 y 2018.

4.5 Técnicas e instrumentos para la recolección de la información documental

En el desarrollo de la Investigación se han utilizado las siguientes técnicas de recolección de la información:

Análisis documental

El análisis documental permite tener un sólido respaldo demostrativo, recurriendo a las fuentes que tienen que ser medidos por su validez, entre lo que se ha escogido en la realidad y lo que se plantea en la teoría. En el análisis documental como instrumento más utilizado es el matricial, o cuadros de contenido de matrices, que también son tratados mediante procesadores estadísticos o paquetes estadísticos de orden cualitativo y/o cuantitativo.

Encuestas

La encuesta o cuestionario es para la presente investigación una de las técnicas de recolección de información para la investigación, debido a que técnicamente construido la encuesta o cuestionario registra con veracidad la problemática existente, pues son los propios actores los que emiten la información que se realiza posteriormente y que permite incluso la validación de las hipótesis.

Entrevistas

Esta técnica la consideramos complementaria y resulta útil sobre todo para recoger información de las personas que conocen y tienen experiencia en las estrategias de inversión, también es útil para tomar información de los especialistas o personas enteradas de la temática, preferentemente en el plano de la aplicación o de la ejecución, más que en el plano puramente teórico, en diversas instancias y en diversos niveles, tienen conocimiento cercano e inquietudes de la realidad problemática. Las entrevistas, para su tratamiento y validez además de ser ordenadas y sistemáticas se efectuarán recogiendo de forma que sean los aspectos generales y luego los específicos los recogidos de quienes hayan sido seleccionados para ser entrevistados. El instrumento de recolección de información de las entrevistas fue los protocolos de entrevista, confeccionados en forma expresa para tal fin. El tratamiento de la información se efectuó, mediante los paquetes estadísticos que se mencionan en el aspecto respectivo. Además, y en menor grado se usará:

Softwareización bibliográfica

Observación ordinaria (bitácora)

Observación participante

Búsqueda en archivos usando Internet

Instrumentos para recibir información

Las principales técnicas fueron:

Análisis documental, que se encargará de la recolección y del estudio de documentos que tienen que ver con las estrategias de inversión y la rentabilidad.

Además, en menor grado se usará

Ficha de trabajo bibliográfico

Guía de observación ordinaria (Registro de Notas, test)

Guía de observaciones participante (Bitácora)

Grabadora y cámara fotográfica y de vídeo

Lista de Chequeo para evaluar el centro de simulación.

4.6 Análisis y procesamiento de datos

Los datos recogidos mediante los métodos y técnicas antes indicados fue interpretados descriptivamente teniendo en cuenta sobre todo los porcentajes obtenidos, con respecto a las respuestas obtenidas, especialmente en el análisis documental, encuestas y entrevistas, que servirán a su vez para contrastar o comprobar las hipótesis que hemos planteado.

Las respuestas a las preguntas del cuestionario fueron clasificadas de tal manera que el procesamiento de los datos se haga a través del software computarizado, y para el análisis de la información se utilizará la inferencia estadística, usando distribución normal.

Procedimiento

Los participantes desarrollaron un programa de fábrica de software. El procedimiento fue el siguiente:

- A. Aplicación de la prueba exploratoria a los estudiantes de la Escuela de Ingeniería de Sistemas del IV Y V ciclo.

- B. Se formará dos grupos equivalentes elegidos aleatoriamente a partir de los estudiantes de la Escuela de Ingeniería de Sistemas del IV Y V ciclo.

Grupo Experimental

Son los estudiantes que desarrollaron el Programa de Fabrica de Software durante un ciclo. Este programa incluye el asesoramiento permanente al alumno, el uso de los talleres de desarrollo de software y un acercamiento a las empresas a través de las visitas.

Grupo de Control

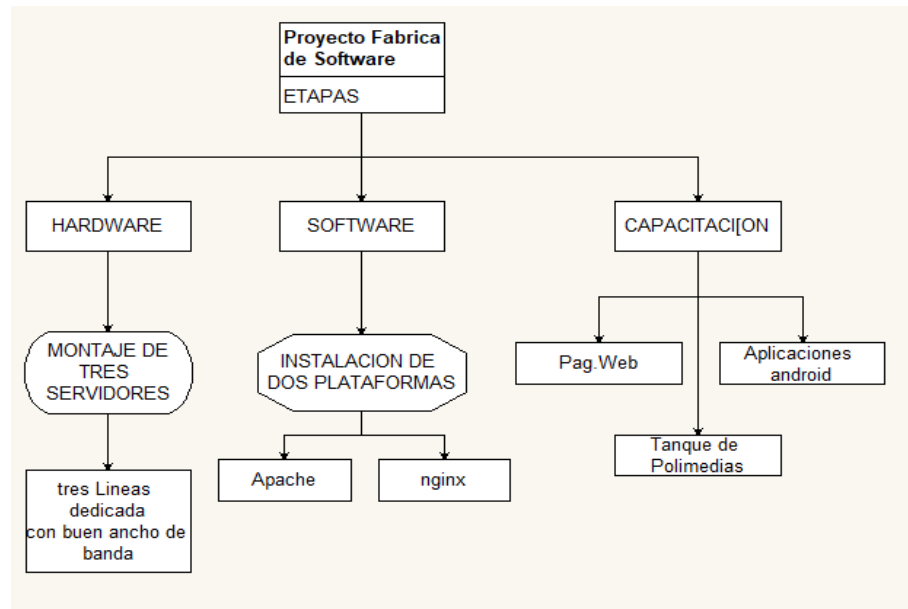
Son los estudiantes que no desarrollaron el Programa de Fabrica de Software durante un ciclo, con quienes se desarrolló las asignaturas con la enseñanza tradicional, resolverán los mismos problemas que el grupo experimental utilizando sus conocimientos previos.

- C. Se evaluó a ambos grupos y los resultados se utilizó en la contratación de las hipótesis planteadas.

Materiales y métodos

Para implementar la fábrica de software se creó una plataforma tecnológica donde los profesores y alumnos participaron de las labores de desarrollo de software en sus diferentes etapas.

Gráfico 5 Actividades de Implementación de la fábrica de software



Hardware

La fábrica de software requiere de tecnología especializada que consiste en:

- servidores físicos
- Líneas telefónicas *dedicadas
- Rack de soporte para los servidores
- Espacio de almacenamiento
- El Rack de metal mecánico
- Dos IP públicas para las plataformas.
- Tres servidores streaming...

Gráfico 6 Rack de Servidores streaming



Gráfico 7 Servidor streaming



Software

Son distintos procedimientos y con ordenamientos lógicos que ayudan a que las tareas se realicen de manera rápida. Es el conjunto de instrucciones que permite la utilización del ordenador. Sin el Software el Hardware no serviría absolutamente para nada. Es la parte intangible de la computadora.

Para el proyecto Fábrica de Software se utilizará HTML, PHP, CSS, servidores Linux, aplicaciones móviles, video en móviles.

El considerar que un proyecto de desarrollo de software podía abordarse como cualquier otro ha llevado a una serie de problemas

El identificar dichos problemas y conocer sus causas es el único método que nos puede llevar hacia su solución. La ingeniería del software tiene tres elementos clave: métodos, herramientas y procedimientos **Capacitación**

Los docentes y alumnos se capacitarán en los ambientes de Educación a distancia, los sábados de 3 a 5 pm

Gráfico 8 Propaganda para la Capacitación

UNACREA
UNIÓN DE CREATIVOS

PRESENTA EL CURSO GRATUITO
PROGRAMACION WEB

INICIO DE CLASES
SÁBADO 16 DE MARZO
WWW.UNAC.WS

INSCRIPCIONES OFICINA DE EDUCACIÓN A DISTANCIA - OBU 2DO PISO VACANTES LIMITADAS

PROYECTO FABRICA DE SOFTWARE

HTML

¿Qué es HTML?

<html>

cat is very gr...
Más imágenes
Ejem...

7 conceptos básicos que debes saber de HTML

HTML

V. RESULTADOS

5.1 Resultados descriptivos

En este capítulo, presentamos los resultados obtenidos de nuestra investigación al haber aplicado el Cuestionario.

El orden lógico seguido ha sido el siguiente: Se elaboró un cuestionario de 16 preguntas. Las 8 primeras es para la variable independiente y los 8 restantes para la variable dependiente.

Para valorar los resultados se elaboró cinco alternativas:

1=Muy malo

2=Malo

3=Regular

4=Bueno

5=Excelente

La aplicación del cuestionario a 92 estudiantes del IV y V ciclo de la facultad de Ingeniería de Sistemas, permitió construir la siguiente Tabla resumen de resultados que se presenta a continuación:

Tabla 2 RESUMEN DE RESULTADOS

N°	Preguntas	Muy malo	Malo	Regular	Buena	Excelente
VARIABLE INDEPENDIENTE: TALLERES DE DESARROLLO DE SOFTWARE						
Tecnologías de Información y comunicación						
1	¿Qué opina las Tics que maneja la Facultad?	4	18	29	24	17
2	¿Se está aplicando las Tics en la Facultad?	2	18	25	29	18
3	¿Las Tics son confiables?	2	9	34	22	25
4	¿Cómo está el nivel de implementación de las Tics?	3	10	25	31	23
Gestión						
5	¿Qué opina de la Gestión en la Facultad?	1	22	32	20	17
6	¿Qué opina de la creación de los talleres de software?	2	19	24	28	19
7	¿Cree que la creación de talleres de software es una buena alternativa?	2	22	29	20	19
8	¿El estatuto favorece la creación de los talleres de software?	2	18	31	26	15
VARIABLE DEPENDIENTE: FORMACIÓN PROFESIONAL						
Rendimiento académico						
9	¿Los talleres permiten comprender mejor los conceptos teóricos?	3	21	26	24	18
10	¿Los talleres permiten lograr la experticia procedimental en el desarrollo de software?	3	21	27	22	19
11	¿Los talleres permiten aplicar los conocimientos a la solución de problemas locales, de la región y nacionales?	3	10	18	33	28
12	¿Los talleres permiten que la carrera sea pertinente en la sociedad?	3	12	16	38	23
Calidad						
13	¿Los participantes aprenden trabajar en equipo?	3	18	24	25	22
14	¿Los participantes desarrollan el liderazgo?	4	9	29	31	19
15	¿Los participantes conocen las necesidades insatisfechas de software de los clientes potenciales?	2	19	25	30	16
16	¿Los participantes desarrollan actitudes positivas como la responsabilidad y puntualidad?	3	12	19	37	21

Fuente: Propia

Tabla 3 RESULTADOS DE LA EVALUACIÓN A LOS DOS GRUPOS

Grupo sin Taller de Software		Grupo con Taller de Software	
ALUMNO	NOTAS	ALUMNO	NOTAS
Alumno 01	14	Alumno 47	11
Alumno 02	11	Alumno 48	9
Alumno 03	8	Alumno 49	10
Alumno 04	17	Alumno 50	13
Alumno 05	18	Alumno 51	12
Alumno 06	8	Alumno 52	16
Alumno 07	14	Alumno 53	10
Alumno 08	17	Alumno 54	11
Alumno 09	14	Alumno 55	14
Alumno 10	12	Alumno 56	13
Alumno 11	8	Alumno 57	10
Alumno 12	14	Alumno 58	17
Alumno 13	12	Alumno 59	15
Alumno 14	13	Alumno 60	17
Alumno 15	11	Alumno 61	14
Alumno 16	17	Alumno 62	15
Alumno 17	11	Alumno 63	18
Alumno 18	11	Alumno 64	13
Alumno 19	18	Alumno 65	18
Alumno 20	7	Alumno 66	10
Alumno 21	13	Alumno 67	15
Alumno 22	16	Alumno 68	18
Alumno 23	9	Alumno 69	17
Alumno 24	10	Alumno 70	13
Alumno 25	12	Alumno 71	13
Alumno 26	14	Alumno 72	10
Alumno 27	12	Alumno 73	17
Alumno 28	15	Alumno 74	15



Alumno 29	15	Alumno 75	13
Alumno 30	12	Alumno 76	11
Alumno 31	15	Alumno 77	10
Alumno 32	9	Alumno 78	13
Alumno 33	9	Alumno 79	17
Alumno 34	18	Alumno 80	13
Alumno 35	10	Alumno 81	12
Alumno 36	10	Alumno 82	16
Alumno 37	13	Alumno 83	18
Alumno 38	8	Alumno 84	17
Alumno 39	15	Alumno 85	9
Alumno 40	10	Alumno 86	17
Alumno 41	12	Alumno 87	11
Alumno 42	18	Alumno 88	16
Alumno 43	13	Alumno 89	13
Alumno 44	7	Alumno 90	18
Alumno 45	14	Alumno 91	14
Alumno 46	7	Alumno 92	13
PROMEDIO	12,4130435	PROMEDIO	13,8043478
DESVIACIÓN ESTÁNDAR	3,22069454	DESVIACIÓN ESTÁNDAR	2,78676659

Fuente: Propia

INTERPRETACIÓN: Las notas obtenidas de los alumnos que participan en ambos grupos, experimentales y de control presentan una diferencia a favor del grupo que participaron en el programa de Talleres de Software. Esta diferencia puede ser significativa o no, para ello se sometió a la prueba de hipótesis.

VI. DISCUSIÓN DE RESULTADOS

6.1 Contrastación y demostración de la hipótesis con los resultados

6-1.1 Prueba de la primera hipótesis secundaria

a. Formulación de las hipótesis

H₀: “La aplicación de tecnologías de información en los Talleres de desarrollo de Software no influyen positivamente en la experticia de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao”

H₁: “La aplicación de tecnologías de información en los Talleres de desarrollo de Software influyen positivamente en la experticia académico de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao”

El propósito de esta prueba de hipótesis es demostrar la independencia o dependencia de las variables. Teniendo en cuenta que H₁ se anticipa a la dirección de la prueba, se realizó una prueba unilateral de cola derecha.

b. Nivel de significación

Asumimos el nivel de significación del 5%,

c. Estadística de prueba

Se utilizó la prueba Chi-cuadrada:
$$\chi^2 = \sum \frac{(fo - fe)^2}{fe}$$

Donde fo =frecuencia observada

fe =frecuencia esperada

La frecuencia observada y la frecuencia esperada se muestran en las siguientes tablas:

Tabla 4 FRECUENCIA OBSERVADA

					TOTAL
	Malo	Regular	Bueno	Excelente	
Malo	2				2
Regular		24	5		29
Bueno		7	47	2	56
Excelente			1	4	5
TOTAL	2	31	53	6	92

Fuente: Propia

Tabla 5 FRECUENCIA ESPERADA

					TOTAL
	Malo	Regular	Bueno	Excelente	
Malo	0,0	0,7	1,2	0,1	2
Regular	0,6	9,8	16,7	1,9	29
Bueno	1,2	18,9	32,3	3,7	56
Excelente	0,1	1,7	2,9	0,3	5
TOTAL	2	31	53	6	92

Fuente: Propia

d. Valor crítico de la estadística de prueba

Primero determinamos el valor crítico del estadístico χ^2 . En la tabla de distribución de Chi-cuadrada para la prueba de una sola cola con $\alpha = 5\%$ y Grados de libertad = $(4-1)(4-1) = 9$

$$\chi_{\alpha}^2 = 16,9$$

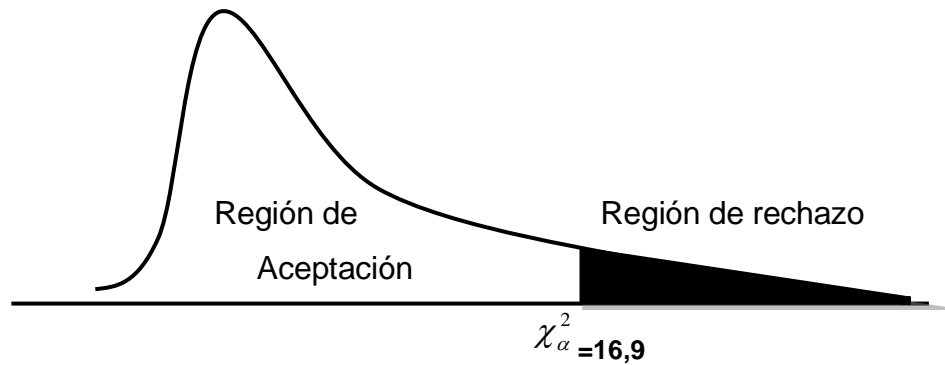


Gráfico 9 curva chi cuadrado

Fuente: Propia

e. Valor de la estadística de prueba

Tabla 6 VALOR CHI-CUADRADA

					TOTAL
	Malo	Regular	Bueno	Excelente	
Malo	88,0435	0,6739	1,1522	0,1304	90,0000
Regular	0,6304	20,7172	8,2029	1,8913	31,4419
Bueno	1,2174	7,4663	6,7339	0,7474	16,1651
Excelente	0,1087	1,6848	1,2276	41,3928	44,4138
TOTAL	90,0000	30,5423	17,3166	44,1619	182,0208

Fuente: Propia

f. Toma de decisión

Como el valor de la estadística de prueba $\chi^2_{calculado}$ está en la región de rechazo, se rechaza H_0 . Se acepta H_1

Por lo tanto, la primera hipótesis

“La aplicación de tecnologías de información en los Talleres de desarrollo de Software influyen positivamente en la experticia académico de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao”.

Queda verificada.

6.1.2 Prueba de la segunda hipótesis secundaria

a. Formulación de las hipótesis

H₀: “La gestión en los Talleres de desarrollo de Software, no influye significativamente en la calidad de aprendizaje de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao”

H₁: “La gestión en los Talleres de desarrollo de Software, influye significativamente en la calidad de aprendizaje de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao”

El propósito de esta prueba de hipótesis es demostrar la independencia o dependencia de las variables. Teniendo en cuenta que H₁ se anticipa a la dirección de la prueba, se realizó una prueba unilateral de cola derecha.

b. Nivel de significación

Asumimos el nivel de significación del 5%,

c. Estadística de prueba

Se utilizó la prueba Chi-cuadrada:
$$\chi^2 = \sum \frac{(fo - fe)^2}{fe}$$

Donde fo =frecuencia observada

fe =frecuencia esperada

La frecuencia observada y la frecuencia esperada se muestran en las siguientes tablas:

Tabla 7 FRECUENCIA OBSERVADA

					TOTAL
	Malo	Regular	Bueno	Excelente	
Muy Malo				1	1
Malo			2		2
Regular	1	11	32		44
Bueno		15	24	1	40
Excelente			5		5
TOTAL	1	26	63	2	92

Fuente: Propia

Tabla 8 FRECUENCIA ESPERADA

					TOTAL
	Malo	Regular	Bueno	Excelente	
Muy Malo	0,0	0,3	0,7	0,0	1
Malo	0,0	0,6	1,4	0,0	2
Regular	0,5	12,4	30,1	1,0	44
Bueno	0,4	11,3	27,4	0,9	40
Excelente	0,1	1,4	3,4	0,1	5
TOTAL	1	26	63	2	92

Fuente: Propia

d. Valor crítico de la estadística de prueba

Primero determinamos el valor crítico del estadístico χ^2 . En la tabla de distribución de Chi-cuadrada para la prueba de una sola cola con $\alpha = 5\%$ y Grados de libertad = $(5 - 1)(4 - 1) = 12$

$$\chi_{\alpha}^2 = 21,0$$

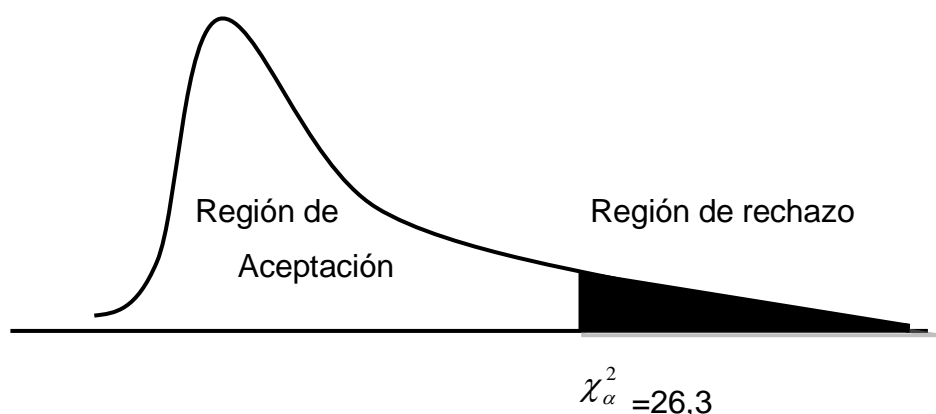


Gráfico 10 Curva chi-cuadrado

Fuente: Propia

e. Valor de la estadística de prueba

Tabla 9 VALOR CHI-CUADRADA

					TOTAL
	Malo	Regular	Bueno	Excelente	
Muy Malo	0,0109	0,2826	0,6848	44,0217	45,0000
Malo	0,0217	0,5652	0,2902	0,0435	0,9206
Regular	0,5692	0,1656	0,1160	0,9565	1,8072
Bueno	0,4348	1,2082	0,4199	0,0196	2,0824
Excelente	0,0543	1,4130	0,7255	0,1087	2,3016
TOTAL	1,0909	3,6346	2,2364	45,1500	52,1119

Fuente: Propia

f. Toma de decisión

Como el valor de la estadística de prueba $\chi^2_{calculado}$ está en la región de rechazo, se rechaza H_0 . Se acepta H_1

Por lo tanto, la segunda hipótesis

“La gestión en los Talleres de desarrollo de Software, influye significativamente en la calidad de aprendizaje de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao” Queda verificada.

6.1.3 Prueba de la hipótesis general

a) Formulación de las hipótesis

H_0 : “El uso de los Talleres de desarrollo de Software, no influye significativamente en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao en el 2018”

$$u_1 - u_2 = 0$$

H_1 : “El uso de los Talleres de desarrollo de Software, influye significativamente en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao en el 2018”

$$u_1 - u_2 < 0$$

El propósito de la prueba de hipótesis es determinar si existe diferencia significativa entre las medias de los grupos experimental y de control con base en evidencias muestrales. Teniendo en cuenta que H_1 se anticipa a la dirección de la prueba, deberá realizarse una prueba unilateral de cola a la derecha.

b) Nivel de significación

Asumimos el nivel de significación del 5%. Dado que la desviación estándar de la diferencia de medias de la población se calcula en base a la desviación estándar de la diferencia de medias de la muestra.

c) Estadística de prueba

La estadística adecuada al problema en la prueba de hipótesis es la Distribución Normal porque el tamaño de las muestras es mayor que 30. La fórmula corresponde a la diferencia de medias.

d) Valor crítico de la estadística de prueba

En la tabla de Distribución Normal para la prueba de una sola cola con un nivel de significación de $\alpha = 5\%$, el estadístico $Z_{\alpha} = -1,645$.

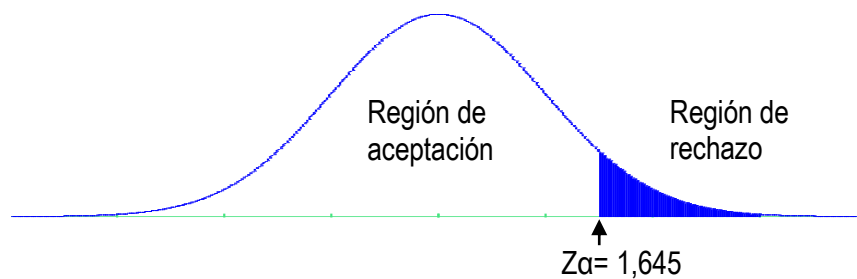


GRAFICO N° 1 Curva normal con el valor crítico

e) Valor de la estadística de prueba

$$Z_{\text{calculado}} = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}}$$

$$Z_{\text{calculado}} = \frac{13,89043478 - 12,41304352}{\sqrt{\frac{3,22069454^2}{46} + \frac{2,78676659^2}{46}}} = 2,22$$

f) Toma de decisiones

Como el valor de la estadística de prueba $Z_{\text{calculado}} = 2,22$ se ubica en la región de rechazo, se rechaza H_0 y se acepta H_1 .

De esta manera la Hipótesis general:

“El uso de los Talleres de desarrollo de Software, influye significativamente en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao en el 2018”.

Queda verificada.

6.2 Contrastación de los resultados con otros similares

(GATLIFF, Bee; WENDEL, Federick, 1998), al examinar las perspectivas generales de Simulación en la educación universitaria identifican cuatro dimensiones donde los estudiantes y profesores pueden potenciar la capacidad creativa y conseguir mayor flexibilidad mental, dentro de un proceso dinámico de enseñanza - aprendizaje. Estas dimensiones son las siguientes: La comunicación por computadoras, Experiencias de interconexión educativa, Información, investigación y Publicaciones y Desarrollo de Proyecto educativos. En la presente investigación se está poniendo en práctica lo sugerido anteriormente, logrando la interconexión con entidades como las municipalidades, empresas y la misma comunidad, en el marco de la Ley Universitaria vigente, donde la universidad tiene que cumplir la responsabilidad social.

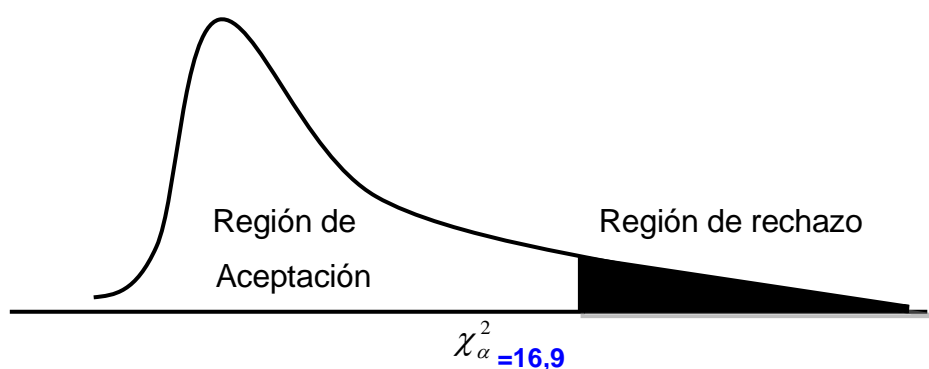
El profesor Iván Álvarez Navia de la Universidad de Salamanca, indica que el objetivo fundamental del taller consiste en adquirir soltura en la instalación, administración y utilización de Software Libre disponible en el ámbito educativo. Así mismo se proporciona una visión general de aspectos relacionados con metodologías, temas legales (licencias) y distribución del Software Libre.

Estamos completamente de acuerdo con el profesor Iván Álvarez que el objetivo fundamental del taller consiste en adquirir soltura en la instalación, administración y utilización de Software Libre disponible con el agregado que los estudiantes no solo van consumir sino van a desarrollar programas o sistemas con la tutoría de los docentes especialistas

Contrastación de la hipótesis con estudios similares:

En esta investigación los resultados obtenidos comprueban lo propuesto por el objetivo general ya que busca explicar de qué manera la aplicación de Talleres de desarrollo de Software influye en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao en el 2018.

Gráfico 11 Curva chi cuadrado



Fuente: Elaboración Propia

Tabla 10 VALOR CHI-CUADRADA

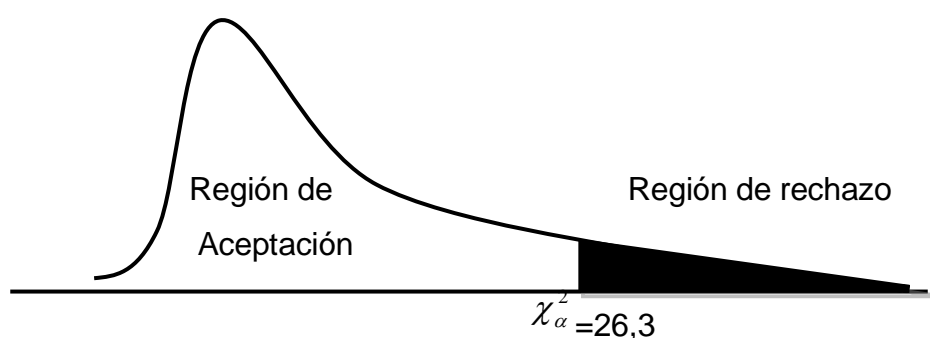
TIC	Experticia				TOTAL
	Malo	Regular	Bueno	Excelente	
Malo	88,0435	0,6739	1,1522	0,1304	90,0000
Regular	0,6304	20,7172	8,2029	1,8913	31,4419
Bueno	1,2174	7,4663	6,7339	0,7474	16,1651
Excelente	0,1087	1,6848	1,2276	41,3928	44,4138
TOTAL	90,0000	30,5423	17,3166	44,1619	182,0208

Fuente: Elaboración Propia

Se observa en la Tabla 6 que el Como el valor de la estadística de prueba $\chi^2_{calculado} = 182$ está en la región de rechazo, se rechaza H_0 . Se acepta H_1 . Por lo tanto, la primera hipótesis “La aplicación de tecnologías de información en los Talleres de desarrollo de Software influyen positivamente en la experticia de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad

Nacional del Callao”. Queda verificada. Este resultado concuerda con los encontrados por Cendejas (2014), quien en su investigación concluye que el modelo integral colaborativo propuesto le permite desarrollar software de una manera más ágil, así mismo concuerda con lo investigado por Araus (2013), quien concluye que se produjo resultados eficientes en el desarrollo de sistemas de información a través creación de componentes de software basado en un patrón de fábrica de software.

Gráfico 12 Curva chi-cuadrado



Fuente: Elaboración Propia

Valor de la estadística de prueba

Tabla 11 VALOR CHI-CUADRADA

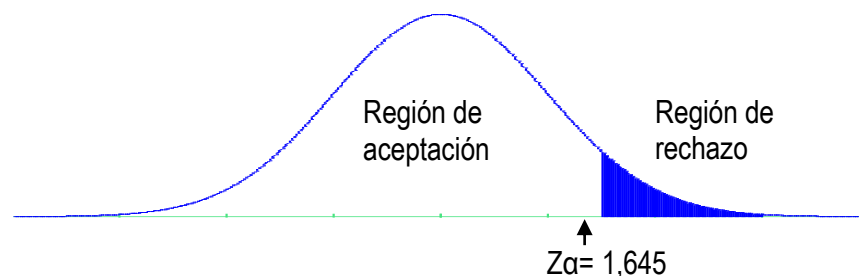
Gestión	Calidad				TOTAL
	Malo	Regular	Bueno	Excelente	
Muy Malo	0,0109	0,2826	0,6848	44,0217	45,0000
Malo	0,0217	0,5652	0,2902	0,0435	0,9206
Regular	0,5692	0,1656	0,1160	0,9565	1,8072
Bueno	0,4348	1,2082	0,4199	0,0196	2,0824
Excelente	0,0543	1,4130	0,7255	0,1087	2,3016
TOTAL	1,0909	3,6346	2,2364	45,1500	52,1119

Fuente: Elaboración Propia

Se observa en la Tabla 9 que el Como el valor de la estadística de prueba $\chi^2_{calculado}$ está en la región de rechazo, se rechaza H_0 . Se acepta H_1 . Por lo tanto, la segunda hipótesis “La gestión en los Talleres de desarrollo de Software, influye

significativamente en la calidad de aprendizaje de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao” Queda verificada. Este resultado concuerda con los encontrados según Guajardo (2014), concluye que el modelo propone factores que llevan a mejorar los indicadores de gestión del tiempo en el desarrollo de proyectos informáticos, así también Montalvo et al. (2012) concluye que una fábrica de software los procesos deben ser automatizados a fin de reducir recursos como tiempo y esfuerzo

Gráfico 13 Curva Normal



Fuente: Elaboración Propia

Se observa en Grafico 1 de la Curva Normal que Como el valor de la estadística de prueba $Z_{calculado} = 2,22$ se ubica en la región de rechazo, se rechaza H_0 y se acepta H_1 . De esta manera la Hipótesis: “La aplicación de los Talleres de desarrollo de Software, influye significativamente en el rendimiento de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao”. Queda verificada. Dicho resultado concuerda con los encontrados según Maturro (2010) quien concluye la capacidad operativa de una empresa no es suficiente para atender todas las necesidades de los clientes, y se requiere enfocar en temas claves de gestión de conocimiento, e infraestructura para sustentar las actividades en el desarrollo de software; así mismo Araus (2013),

que el modelo de fábrica de software incorpora patrones que hace que los sistemas desarrollados sean más completo y escalable, pero en contraposición también indica que una de falencias es encontrar poca documentación existente.

Como las hipótesis secundarias han sido verificadas, la hipótesis principal: “La aplicación de Talleres de desarrollo de Software influye significativamente en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao en el 2018”. También queda verificada. Estos resultados concuerdan con los encontrados por Barzola y Henríquez (2014), ya que el modelo define actividades para el desarrollo de software, realizando la verificación, validación y captura de información para el aseguramiento de la calidad.

(TEOFILO ABROCIO, 2017) en su investigación indica que los resultados obtenidos comprueban lo propuesto por el objetivo general ya que ratifican que el modelo de fábrica de influye de manera positiva en el desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C. Utilizando tablas de contingencia se observa en la Tabla 16 que el 42.55% (40 respuestas) considera que el nivel del desarrollo de sistemas de información en UCL Global Perú S.A.C. es de “Alta prevalencia” y solo el 19.15% (18 respuestas) considera que el nivel es de “Baja prevalencia”; asimismo se tiene que la mayor frecuencia se encuentra en el cruce del nivel “Optimo” de la variable modelo de fábrica de software y el nivel “Alta prevalencia” de la variable desarrollo de sistemas de información en UCL Global Perú S.A.C. con 28 respuestas representando el 29.79% del total. En el plano inferencial el resultado de la Tabla 20 matriz de información de ajuste de los modelos, se obtuvo un $p\text{-valor} = 0.000 < 0.05$, se concluyó con un de nivel de significación del 5%, que el modelo de regresión logístico ordinal estimado es adecuado para analizar la influencia de la variable modelo de fábrica de software sobre la

variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C. Con respecto al resultado de la Tabla 21 matriz de bondad de ajuste, se obtuvo un p-valor del estadístico, el chi-cuadrado de Pearson y el chi-cuadrado sobre la desviación, mayor al α igual a 0.05, se concluyó que el modelo de regresión logístico ordinal estimado es compatible con los datos de la variable modelo de fábrica de software y la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C; Con respecto al resultado de la Tabla 22 resultado pseudo R-cuadrado, se obtuvo un valor del coeficiente de determinación, indicando que el 45.2% de la variación de la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C. es explicado por la variable modelo de fábrica de software.

En función de los resultados anteriores se concluyó aceptando la hipótesis alternativa, que el sistema de la variable modelo de fábrica de software influye de manera positiva en la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C. Dichos resultados concuerdan con los encontrados por Cendejas (2014), quien en su investigación concluye que el modelo integral colaborativo propuesto le permite desarrollar software de una manera más ágil, así mismo concuerda con lo investigado por Araus (2013), quien concluye que se produjo resultados eficientes en el desarrollo de sistemas de información a través creación de componentes de software basado en un patrón de fábrica de software. De acuerdo a los resultados encontrados, el uso del modelo de desarrollo de software eficiente en UCL Global Perú S.A.C. requiere el incremento de desarrollo de componentes de software.

En esta investigación los resultados obtenidos comprueban lo propuesto por el objetivo específico 1 ya que ratifican que el modelo de fábrica de influye de manera positiva en la dimensión gestión de capacidad en el desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C. Utilizando tablas de



contingencia se observa en la Tabla 17 que el 51.06% (48 respuestas) considera que el nivel de la dimensión gestión de capacidad en la variable desarrollo de sistemas de información en UCL Global Perú S.A.C. es de “Alta prevalencia” y solo el 11.70% (11 respuestas) considera que el nivel es de “Baja prevalencia”; asimismo se tiene que la mayor frecuencia se encuentra en el cruce del nivel “Óptimo” de la variable modelo de fábrica de software y el nivel “Alta prevalencia” de la dimensión gestión de capacidad de la variable desarrollo de sistemas de información en UCL Global Perú S.A.C. con 39 respuestas representando el 41.49% del total. En el plano inferencial el resultado de la Tabla 23 matriz de información de ajuste de los modelos, se obtuvo un p-valor = 0,000 < 0.05, se concluyó con un nivel de significación del 5%, que el modelo de regresión logístico ordinal estimado es adecuado para analizar la influencia de la variable modelo de fábrica de software sobre la dimensión gestión de capacidad de la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C. Con respecto al resultado de la Tabla 24 matriz de bondad de ajuste, se obtuvo un p-valor del estadístico, el chi-cuadrado de Pearson y el chi-cuadrado sobre la desviación, mayor al α igual a 0.05, se concluyó que el modelo de regresión logístico ordinal estimado es compatible con los datos de la variable modelo de fábrica de software y la dimensión gestión de capacidad de la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C; Con respecto al resultado de la Tabla 25 resultado pseudo R-cuadrado, se obtuvo un valor del coeficiente de determinación, indicando que el 56.8% de la variación de la dimensión gestión de capacidad de la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C. es explicado por la variable modelo de fábrica de software.

En función de los resultados anteriores se concluyó aceptando la hipótesis alternativa, que la variable modelo de fábrica de software influye de manera positiva en la dimensión gestión de capacidad



de la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C., dichos resultados concuerdan con los encontrados según Maturro (2010) concluye la capacidad operativa de una empresa no es suficiente para atender todas las necesidades de los clientes, y se requiere enfocar en temas claves de gestión de conocimiento, e infraestructura para sustentar las actividades en el desarrollo de software; así mismo Araus (2013), que el modelo de fábrica de software incorpora patrones que hace que los sistemas desarrollados sean más completo y escalable, pero en contraposición también indica que una de falencias es encontrar poca documentación existente. De acuerdo a los resultados encontrados, el uso del modelo de desarrollo de software eficiente en UCL Global Perú S.A.C. requiere contar con la gestión de conocimientos y experiencias.

En esta investigación los resultados obtenidos comprueban lo propuesto por el objetivo específico 2 ya que ratifican que la variable modelo de fábrica de influye de manera positiva en la dimensión gestión del tiempo de la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C. Utilizando tablas de contingencia se observa en la Tabla 18 que el 44.68% (42 respuestas) considera que el nivel de la dimensión gestión del tiempo en la variable desarrollo de sistemas de información en UCL Global Perú S.A.C. es de “Alta prevalencia” y solo el 11.70% (11 respuestas) considera que el nivel es de “Baja prevalencia”; asimismo se tiene que la mayor frecuencia se encuentra en el cruce del nivel “Optimo” de la variable modelo de fábrica de software y el nivel “Alta prevalencia” de la dimensión gestión del tiempo de la variable desarrollo de sistemas de información en UCL Global Perú S.A.C. con 32 respuestas representando el 34.04% del total. En el plano inferencial el resultado de la Tabla 26 matriz de información de ajuste de los modelos, se obtuvo un p-valor = 0,000 < 0.05, se concluyó con un de nivel de significación del 5%, que el modelo de regresión logístico ordinal estimado es



adecuado para analizar la influencia de la variable modelo de fábrica de software sobre la dimensión gestión del tiempo de la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C. Con respecto al resultado de la Tabla 27 matriz de bondad de ajuste, se obtuvo un p-valor del estadístico, el chi-cuadrado de Pearson y el chi-cuadrado sobre la desviación, mayor al α igual a 0.05, se concluyó que el modelo de regresión logístico ordinal estimado es compatible con los datos de la variable modelo de fábrica de software y la dimensión gestión del tiempo de la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C; Con respecto al resultado de la Tabla 28 resultado pseudo R-cuadrado, se obtuvo un valor del coeficiente de determinación, indicando que el 41.6% de la variación de la dimensión gestión del tiempo de la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C. es explicado por la variable modelo de fábrica de software.

En función de los resultados anteriores se concluyó aceptando la hipótesis alternativa, que la variable modelo de fábrica de software influye de manera positiva en la dimensión gestión del tiempo de la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C.; dichos resultados concuerdan con los encontrados según Guajardo (2014), concluye que el modelo propone factores que llevan a mejorar los indicadores de gestión del tiempo en el desarrollo de proyectos informáticos, así también Montalvo et al. (2012) concluye que una fábrica de software los procesos deben ser automatizados a fin de reducir recursos como tiempo y esfuerzo. De acuerdo a los resultados encontrados, el uso del modelo de desarrollo de software eficiente en UCL Global Perú S.A.C. requiere medir la productividad.

En esta investigación los resultados obtenidos comprueban lo propuesto por el objetivo específico 3 ya que ratifican que la variable modelo de fábrica de software influye de manera positiva en la dimensión satisfacción del cliente de la variable desarrollo de



sistemas de información en la empresa UCL Global Perú S.A.C. Utilizando tablas de contingencia se observa en la Tabla 19 que el 44.68% (42 respuestas) considera que el nivel de la dimensión satisfacción del cliente en la variable desarrollo de sistemas de información en UCL Global Perú S.A.C. es de “Alta prevalencia” y solo el 12.77% (12 respuestas) considera que el nivel es de “Baja prevalencia”; asimismo se tiene que la mayor frecuencia se encuentra en el cruce del nivel “Optimo” de la variable modelo de fábrica de software y el nivel “Alta prevalencia” de la dimensión satisfacción del cliente de la variable desarrollo de sistemas de información en UCL Global Perú S.A.C. con 32 respuestas representando el 34.04% del total. En el plano inferencial el resultado de la Tabla 29 matriz de información de ajuste de los modelos, se obtuvo un p-valor = 0,000 < 0.05, se concluyó con un nivel de significación del 5%, que el modelo de regresión logístico ordinal estimado es adecuado para analizar la influencia de la variable modelo de fábrica de software sobre la dimensión satisfacción del cliente de la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C. Con respecto al resultado de la Tabla 28 matriz de bondad de ajuste, se obtuvo un p-valor del estadístico, el chi-cuadrado de Pearson y el chi-cuadrado sobre la desviación, mayor al α igual a 0.05, se concluyó que el modelo de regresión logístico ordinal estimado es compatible con los datos de la variable modelo de fábrica de software y la dimensión satisfacción del cliente de la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C; Con respecto al resultado de la Tabla 28 resultado pseudo R-cuadrado, se obtuvo un valor del coeficiente de determinación, indicando que el 36.8% de la variación de la dimensión satisfacción del cliente de la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C. es explicado por la variable modelo de fábrica de software.



En función de los resultados anteriores se concluyó aceptando la hipótesis alternativa, indicando que la variable modelo de fábrica de software influye de manera positiva en la dimensión satisfacción del cliente de la variable desarrollo de sistemas de información en la empresa UCL Global Perú S.A.C.; estos resultados concuerdan con los encontrados por Barzola y Henríquez (2014), ya que el modelo define actividades para el desarrollo de software, realizando la verificación, validación y captura de información para el aseguramiento de la calidad. De acuerdo a los resultados encontrados, el uso del modelo de desarrollo de software eficiente en UCL Global Perú S.A.C. requiere incrementar el grado de cumplimiento de requisitos del cliente.

6.3 Responsabilidad ética de acuerdo a los reglamentos vigentes

Como autor de la presente investigación asumo mi responsabilidad de respeto a los reglamentos vigentes de la Universidad Nacional del Callao. También a todo ello, acompaño valores de respeto al derecho de autor de otros investigadores que me antecedieron y que han sido citados en la presente investigación.



CONCLUSIONES

Primero La presente investigación comprueba la hipótesis principal que indica que el Uso de Talleres de desarrollo de Software influye significativamente en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao en el 2017 – 2018 determinado la distribución Normal, donde el valor Z calculado sale en la región de rechazo.

Segundo La presente investigación realiza la comprobación que la aplicación de tecnologías de información en los Talleres de desarrollo de Software influye positivamente en la experticia de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao en el rendimiento de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao.

Tercero La presente investigación realiza la comprobación que la gestión en los Talleres de desarrollo de Software, influye significativamente en la calidad de aprendizaje de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao es explicado por la prueba de hipótesis usando la distribución chi-cuadrado, donde el valor chi calculado sale en la región de rechazo.

RECOMENDACIONES

Primero Para incrementar el Uso de Talleres de desarrollo de Software en todas las universidades donde haya Escuela de Ingeniería de sistemas, Escuela de Informática o Escuela de computación, se debe de realizar capacitaciones de sensibilización sobre la adopción completa de las fábricas de software en las universidades y como esto va a contribuir con la consecución de los objetivos estratégicos como la obtención de ventajas competitivas, diferenciación de la competencia, incremento de valor entregado a los clientes y por consiguiente mayor competitividad.

Segundo Para incrementar la aplicación de tecnologías de información en los Talleres de desarrollo de Software para mejorar el rendimiento académico por cuanto el proceso enseñanza aprendizaje es más efectiva y se obtiene mejores resultados, se debe de sensibilizar al personal que a través de la gestión de capacidad, todos los servicios de desarrollo de sistemas de información se vean respaldados por una capacidad de proceso y almacenamiento suficiente, estando correctamente dimensionado, que garantice que los procesos críticos tengan la capacidad suficiente para poder llevar a cabo sus funciones.

Tercero Para incrementar la gestión en los Talleres de desarrollo de Software, la calidad de aprendizaje de los estudiantes de la implementación de los Talleres de desarrollo de Software, por cuanto integra a la Universidad con la sociedad, la empresa y con los gobiernos locales y regionales, se debe de sensibilizar al personal para tomar conciencia de la importancia del tiempo como recurso y que a través de la gestión del tiempo, adoptar las mejoras prácticas sugeridas por el modelo de fábrica de software, como el trabajo en equipo, la delegación de funciones y actividades, la motivación al personal; programando eficazmente el tiempo con el fin de hacer más productiva el área de desarrollo y conseguir los resultados de acuerdo

a lo planificado. Se debe aplicar políticas de gestión de expectativas de los clientes, asegurándose su participación en el equipo de proyecto para conocer sus necesidades y procurar atenderlo de la mejor manera.

REFERENCIAS BIBLIOGRAFICAS

- ALLES, M. (2007). *Desarrollo del Talento humano basado en competencias*. Argentina: Ediciones Granice.
- ALTBACH, P. G. (2009). *Educación Superior Comparada*. Buenos Aires: UP Universidad de Palermo.
- ALVARADO OYARCE, D. (2000). *Teoría de la calidad total*. Perú: Editorial Universal.
- BENAVIDES ESPINDOLA, O. (2001). *Competencias y competitividad*. Colombia: McGraw Hill.
- BERNAL, C. A. (2010). *Metodología de la Investigación, 3ed*. Colombia: Prentice Hall.
- BOHLANDER, Snell y SHERMAN. (2001). *Administración de Recursos Humanos*. España: Thomson.
- CASTILLO, J. (2006). *Administración de Personal*. Colombia: Ecoe.
- CHIAVENATO, I. (2009). *Gestión del Talento humano*. España: McGraw Hill.
- DI LLIONARDO, G. y. (2009). *El ABC del gerenciamiento*. Argentina: Osmar DBuyatti.
- FAIRLLEY, R. (1987). *Ingeniería del Software*. España: Mc Graw Hill.
- GARETH, Jones y Jennifer M. GEORGE . (2006). *Administración contemporánea*. México: McGraw Hill.
- GATLIFF, Bee; WENDEL, Federick. (1998). *The American Journal of Distance Education, Vol. 12 N° 1*. New York: Institutional Colaboration and Team Teaching.
- HERNÁNDEZ, S. (2006). *Metodología de la Investigación Científica*. México: McGraw Hill.
- JESUS, S. (2005). *La internacionalización de las universidades como estrategia para el desarrollo institucional*. México: Innovación Educativa.
- KENDAL, K. &. (1996). *Analisi y Diseño de Sistemas*. Mexico: Prentice Hall.

- KENNETH C. LAUDON, J. (2004). *Sistemas de Informacion Gerencial*. Mexico: Pearson rentice Hall.
- KOONTZ, H. y c. O'DONNELLII, C. (1986). *Curso de Administración Moderna*. México: McGraw Hill.
- MICHAEL, Hitt, Black STEWART y Lyman PORTER. (2006). *Administración, 9ed.* México: Pearson Educación.
- Naciones Unidas. (2003-2005). *Declaracion de Principios de la ONU*. Ginebra-Tùnez: ONU.
- O`BRIEN, J. A. (2003). *Sistemas de Información Gerencial*. Colombia: Mc Graw Hill.
- QUINTANILLA, M. Á. (2005). *FILOSOFÍA DE LA TECNOLOGÍA*. Lince. Lima. Perú: Nuevos tiempos. Nuevas ideas.
- RAYMOND, M. (2000). *Sistemas de Información Gerencial* (Septima ed.). Mexico: Prentice Hall.
- ROMANO, J. R. (2018). *La Gran transición - rets y oportuniaddes del cambio tecnológivco exponencial*. México: Fondo de Cultura Económica.
- SIEMENS, G. (2004). *Conectivismo: Una teoria de aprendizaje para la era digital*. Colombia: Creative Commons 2.5.
- TEOFILO ABROCIO, D. (2017). *Modelo de fábrica de software para el desarrollo de sistemas de información en la empresa UCL Global Perú*. Lima: Universidad Cesar Vallejo.
- TORO SUAREZ, L. Y. (2015). *Tesis: LA IMPORTANCIA DEL TRABAJO EN EQUIPO EN LAS ORGANIZACIONES*. Bogotá: Univeridad Militar Nueva Granada.
- VALLE CABRERA, Ramón J. Raquel SANZ VALLE, Antonio ARAGON SANCHEZ, Fernando Martín ALCAZAR, Maria Luz FERNANDEZ ALLES. (2006). *La gestión de los recursos humanos*. España: Prentice Hall.

ANEXOS

MATRIZ DE CONSISTENCIA

TÍTULO: “USO DE TALLERES DE DESARROLLO DE SOFTWARE EN LA FORMACIÓN PROFESIONAL DE LOS ESTUDIANTES DE LA ESCUELA DE INGENIERÍA DE SISTEMAS DE LA UNIVERSIDAD NACIONAL DEL CALLAO”

Problema	Objetivo	Hipótesis	Operacionalización			Métodos
			Variabl e	Definición conceptual	Indicadores	
<p>General ¿De qué manera el uso de Talleres de desarrollo de software influye en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao en el 2017 – 2018?</p> <p>Específicos 1. ¿De qué manera la aplicación de tecnologías de información en los Talleres de desarrollo de Software influyen en el rendimiento académico de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao? 2. ¿De qué manera la gestión en los Talleres de desarrollo de Software, influye en la calidad de aprendizaje de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao?</p>	<p>General Explicar de qué manera el uso de Talleres de desarrollo de Software influye en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao en el 2017 – 2018</p> <p>Específicos 1. Precisar de qué manera la aplicación de tecnologías de información en los Talleres de desarrollo de Software influyen en el rendimiento académico de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao. 2. Determinar de qué manera la gestión en los Talleres de desarrollo de Software, influye en la calidad de aprendizaje de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao.</p>	<p>General el uso de Talleres de desarrollo de Software influye significativamente en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao en el 2017 – 2018</p> <p>Específicos 1. La aplicación de tecnologías de información en los Talleres de desarrollo de Software influyen positivamente en el rendimiento académico de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao 2. La gestión en los Talleres de desarrollo de Software, influye significativamente en la calidad de aprendizaje de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao.</p>	<p>Variable independiente: X Talleres de desarrollo de software</p> <p>Variable dependiente: Y Formación profesional</p>	<p>Son medios tecnológicos que utilizan equipos y programas para el análisis diseño e implantación de programas, sistemas, páginas web, etc.</p> <p>Es el logro de las competencias de las asignaturas tanto en la teórica como en la práctica, de manera permanente, para que cuando egrese, pueda ejercer su profesión de manera eficiente y eficaz.</p>	<p>X1. Tecnologías de información y comunicación X2. Gestión programa Contenido Estrategia Equipamiento Métodos Y1 Rendimiento Y2. Calidad de aprendizaje</p>	<p><u>Tipo y Nivel:</u> investigación Aplicada, Correlacional. <u>Población y muestra.</u> La población en estudio fue 120 estudiantes del IV y V ciclo de la Escuela de Ingeniería de Sistemas de la UNAC que están matriculados en el Semestre académico 2017 - 2018. La muestra estará conformada por 92 estudiantes del IV y V ciclo de la Escuela de Ingeniería de Sistemas de la UNAC que están matriculados en el Semestre académico 2017 - 2018. La muestra fue dividida en dos grupos de manera aleatoria.</p>

INSTRUMENTO DE RECOLECCIÓN DE DATOS

Tabla 12 CUESTIONARIO

Solicitamos su colaboración para responder el presente cuestionario en forma anónima, sobre una investigación sobre “Uso de talleres de desarrollo de software en la formación profesional de los estudiantes de la Escuela de Ingeniería de Sistemas de la Universidad Nacional del Callao”

1=Muy malo , 2=Malo, 3=Regular, 4=Bueno, 5=Excelente

N°	Preguntas	Muy malo	Malo	Regular	Buena	Excelente
----	-----------	----------	------	---------	-------	-----------

VARIABLE INDEPENDIENTE: TALLERES DE DESARROLLO DE SOFTWARE

Tecnologías de Información y comunicación

- 1 ¿Qué opina las Tics que maneja la Facultad?
- 2 ¿Se está aplicando las Tics en la Facultad?
- 3 ¿Las Tics son confiables?
- 4 ¿Cómo está el nivel de implementación de las Tics?

Gestión

- 5 ¿Qué opina de la Gestión en la Facultad?
- 6 ¿Qué opina de la creación de los talleres de software?
- 7 ¿Cree que la creación de talleres de software es una buena alternativa?
- 8 ¿El estatuto favorece la creación de los talleres de software?

VARIABLE DEPENDIENTE: FORMACIÓN PROFESIONAL

Rendimiento académico

- 9 ¿Los talleres permiten comprender mejor los conceptos teóricos?
- 10 ¿Los talleres permiten lograr la experticia procedimental en el desarrollo de software?
- 11 ¿Los talleres permiten aplicar los conocimientos a la solución de problemas locales, de la región y nacionales?
- 12 ¿Los talleres permiten que la carrera sea pertinente en la sociedad?

Calidad

- 13 ¿Los participantes trabajan en equipo?
- 14 ¿Los participantes desarrollan el liderazgo?
- 15 ¿Los participantes conocen las necesidades insatisfechas de software de los clientes potenciales?
- 16 ¿Los participantes desarrollan actitudes positivas como la responsabilidad y puntualidad?

Fuente: Propia

VALIDACIÓN DEL INSTRUMENTO

La confiabilidad usando Alfa de Cronbach se realizó tomando los 15 ítems que contiene el instrumento y haciendo un ensayo con 9 entrevistados. El resultado se presenta en la siguiente Tabla:

Tabla 13 ALFA DE CRONBACH

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14	Q15	Q16	Total
1	3	3	2	2	2	3	3	2	2	2	2	3	3	3	2	2	39
2	1	2	1	2	2	2	2	1	2	1	2	1	1	1	2	1	24
3	4	3	2	2	4	1	2	4	2	3	2	3	2	4	3	4	45
4	3	2	2	4	4	4	3	2	3	3	2	2	3	2	2	2	43
5	2	2	2	2	2	2	1	3	2	2	2	5	1	3	4	1	36
6	3	3	2	2	2	3	3	2	2	2	2	3	3	3	2	2	39
7	2	2	2	2	2	2	1	2	2	2	2	2	2	2	2	2	31
8	4	3	2	2	4	1	2	4	2	3	2	3	2	4	3	4	45
9	3	2	2	4	4	4	3	2	3	3	2	2	3	2	2	2	43
10	2	2	2	2	2	2	1	3	2	2	2	5	1	3	4	1	36
11	1	4	4	1	3	4	3	4	1	1	3	4	3	3	4	4	47
12	1	2	1	2	2	2	2	1	2	1	2	1	1	1	2	1	24
13	4	3	2	2	4	1	2	4	2	3	2	3	2	4	3	4	45
14	3	2	2	4	4	4	3	2	3	3	2	2	3	2	2	2	43
15	2	2	2	2	2	2	1	3	2	2	2	5	1	3	4	1	36
Total	38	37	30	35	43	37	32	39	32	33	31	44	31	40	41	33	576
Var	1.05	0.38	0.40	0.76	0.92	1.18	0.65	1.04	0.25	0.56	0.06	1.66	0.73	0.89	0.73	1.36	12.61

Fuente: Propia

Número de Ítems

k= 16

Sumatoria de las varianzas de los Ítems

$\sum \text{var} = 12.61$

Varianza de la suma de los ítems

var= 50.37

ALFA DE CRONBACH

$\alpha = 0.80$

$$\alpha = \frac{K}{K-1} \left[1 - \frac{\sum \text{var}}{\text{var}} \right]$$

El coeficiente Alfa (α) de Cronbach nos indica que entre más cerca de 1 esté, más alto es el grado de confiabilidad, en este caso, el resultado nos da un valor de 0.80, entonces se puede determinar que el instrumento empleado tiene un alto grado de confiabilidad.

BASE DE DATOS

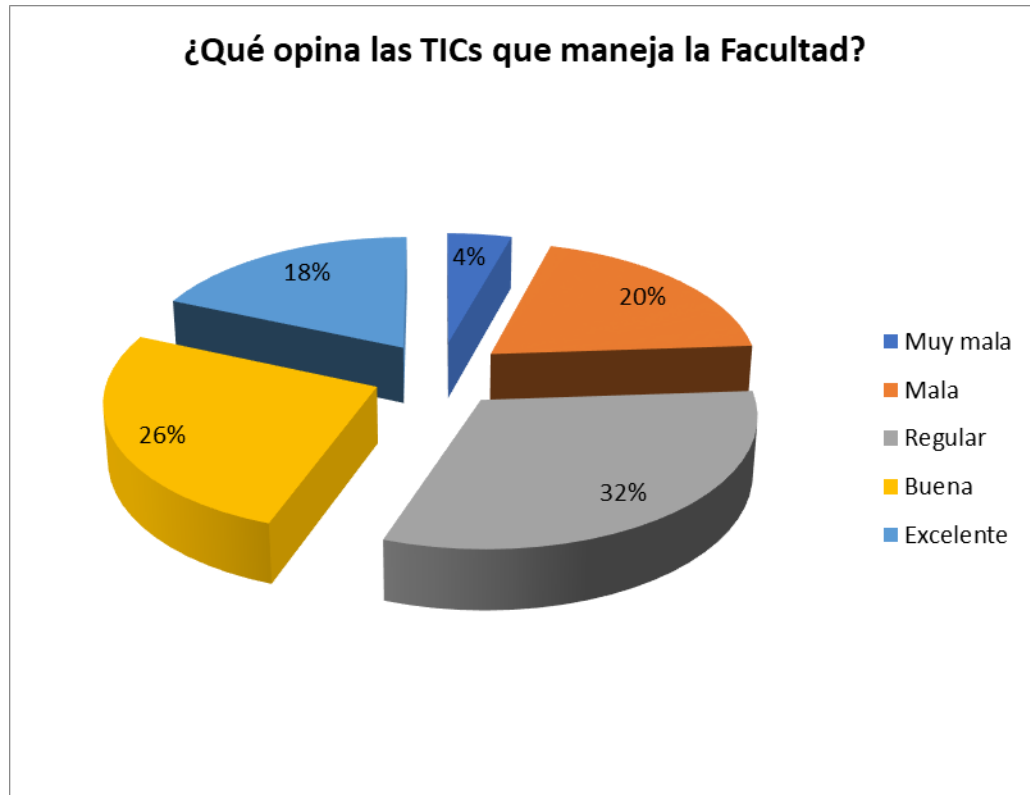
Tabla 14 RESUMEN DE RESULTADOS

N°	Pregunta	Muy malo	Malo	Regular	Buena	Excelente
VARIABLE INDEPENDIENTE: TALLERES DE DESARROLLO DE SOFTWARE						
Tecnologías de Información y comunicación						
1	¿Qué opina las Tics que maneja la Facultad?	4	18	29	24	17
2	¿Se está aplicando las Tics en la Facultad?	2	18	25	29	18
3	¿Las Tics son confiables?	2	9	34	22	25
4	¿Cómo está el nivel de implementación de las Tics?	3	10	25	31	23
Gestión						
5	¿Qué opina de la Gestión en la Facultad?	1	22	32	20	17
6	¿Qué opina de la creación de los talleres de software?	2	19	24	28	19
7	¿Cree que la creación de talleres de software es una buena alternativa?	2	22	29	20	19
8	¿El estatuto favorece la creación de los talleres de software?	2	18	31	26	15
VARIABLE DEPENDIENTE: FORMACIÓN PROFESIONAL						
Rendimiento académico						
9	¿Los talleres permiten comprender mejor los conceptos teóricos?	3	21	26	24	18
10	¿Los talleres permiten lograr la experticia procedimental en el desarrollo de software?	3	21	27	22	19
11	¿Los talleres permiten aplicar los conocimientos a la solución de problemas locales, de la región y nacionales?	3	10	18	33	28
12	¿Los talleres permiten que la carrera sea pertinente en la sociedad?	3	12	16	38	23
Calidad						
13	¿Los participantes trabajan en equipo?	3	18	24	25	22
14	¿Los participantes desarrollan el liderazgo?	4	9	29	31	19
15	¿Los participantes conocen las necesidades insatisfechas de software de los clientes potenciales?	2	19	25	30	16
16	¿Los participantes desarrollan actitudes positivas como la responsabilidad y puntualidad?	3	12	19	37	21

Fuente: Propia



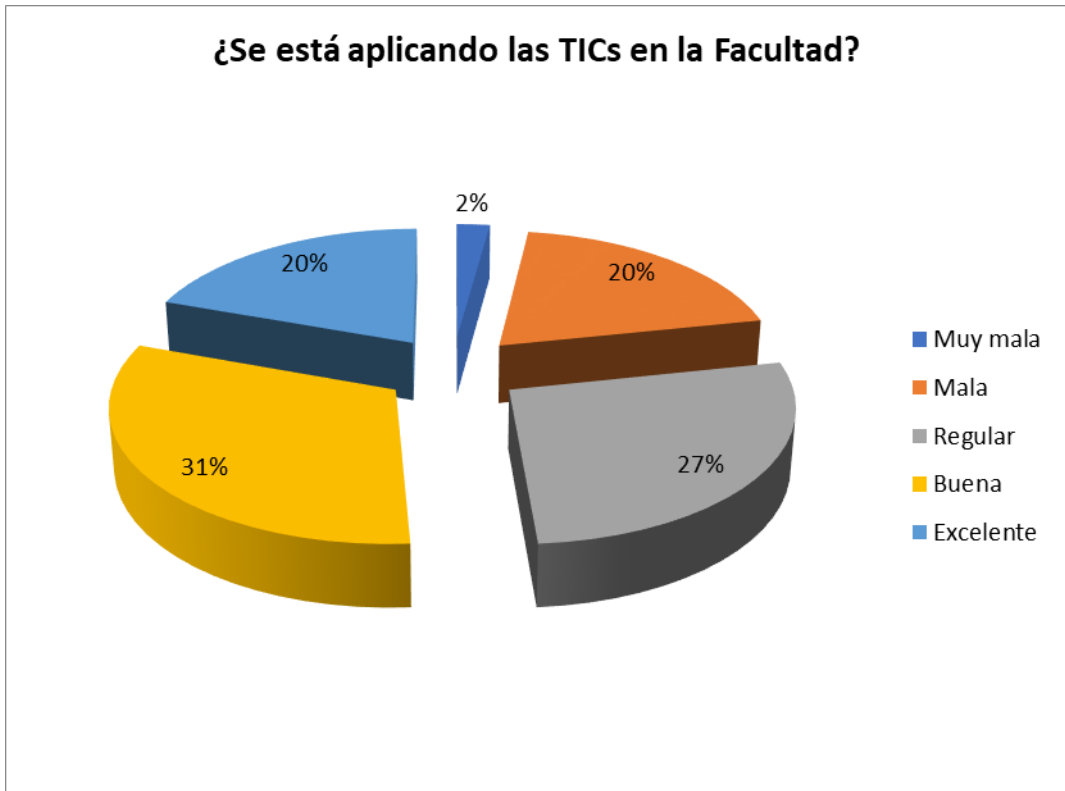
Gráfico 14 Tics en la Facultad de Ingeniería Industrial y de Sistemas



Fuente: Propia

INTERPRETACIÓN: Los estudiantes al ser encuestados sobre las Tics que maneja la facultad, el 44% indicaron que es excelente o buena, 32% indicaron que es regular, el 20% indicaron que es mala y solo el 4% indicaron que es muy mala.

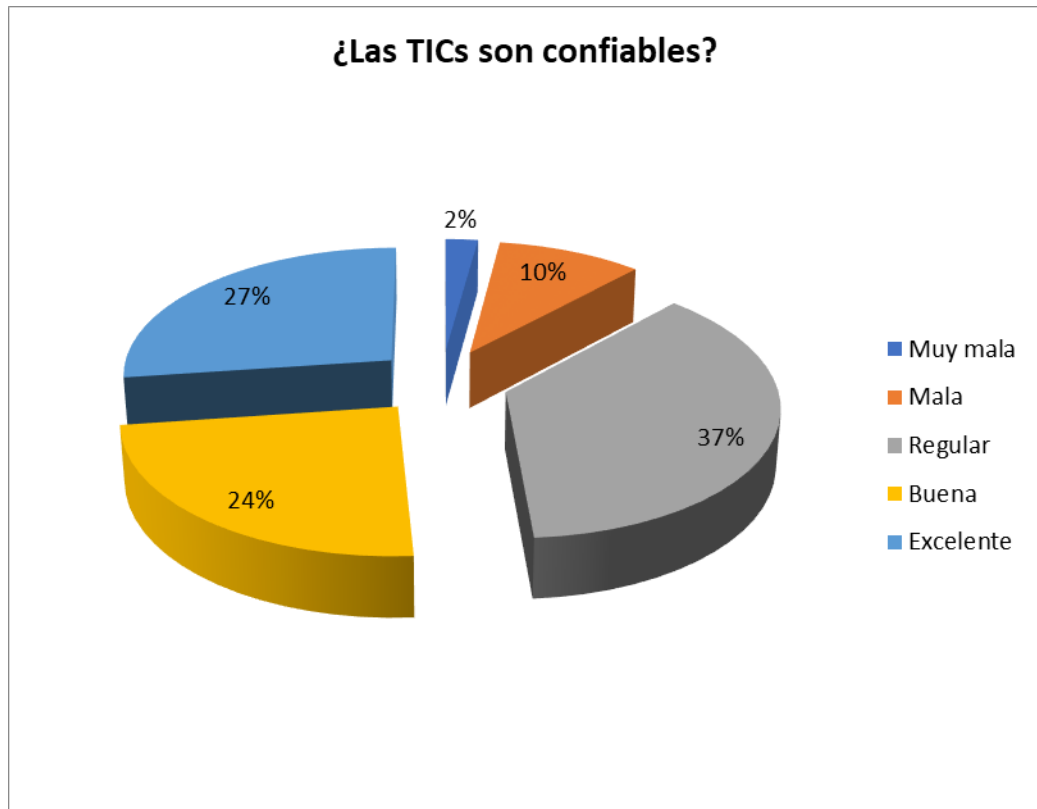
Gráfico 15 Aplicación de las Tics



Fuente: Propia

INTERPRETACIÓN: Los estudiantes al ser encuestados sobre si se está aplicando las Tics, el 51% indicaron que es excelente o buena, 27% indicaron que es regular, el 20% indicaron que es mala y solo el 2% indicaron que es muy mala.

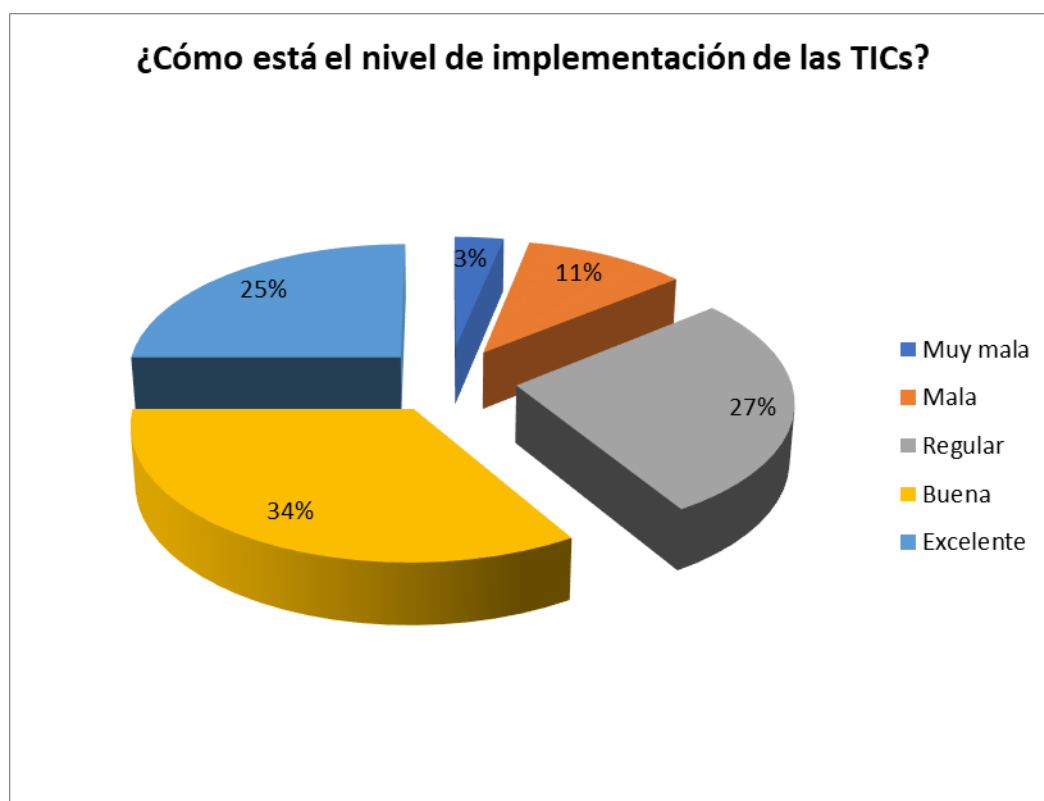
Gráfico 16 Confiabilidad de los Tics



Fuente: Propia

INTERPRETACIÓN: Los estudiantes al ser encuestados sobre si las Tics son confiables, el 51% indicaron que es excelente o buena, 37% indicaron que es regular, el 10% indicaron que es mala y solo el 2% indicaron que es muy mala.

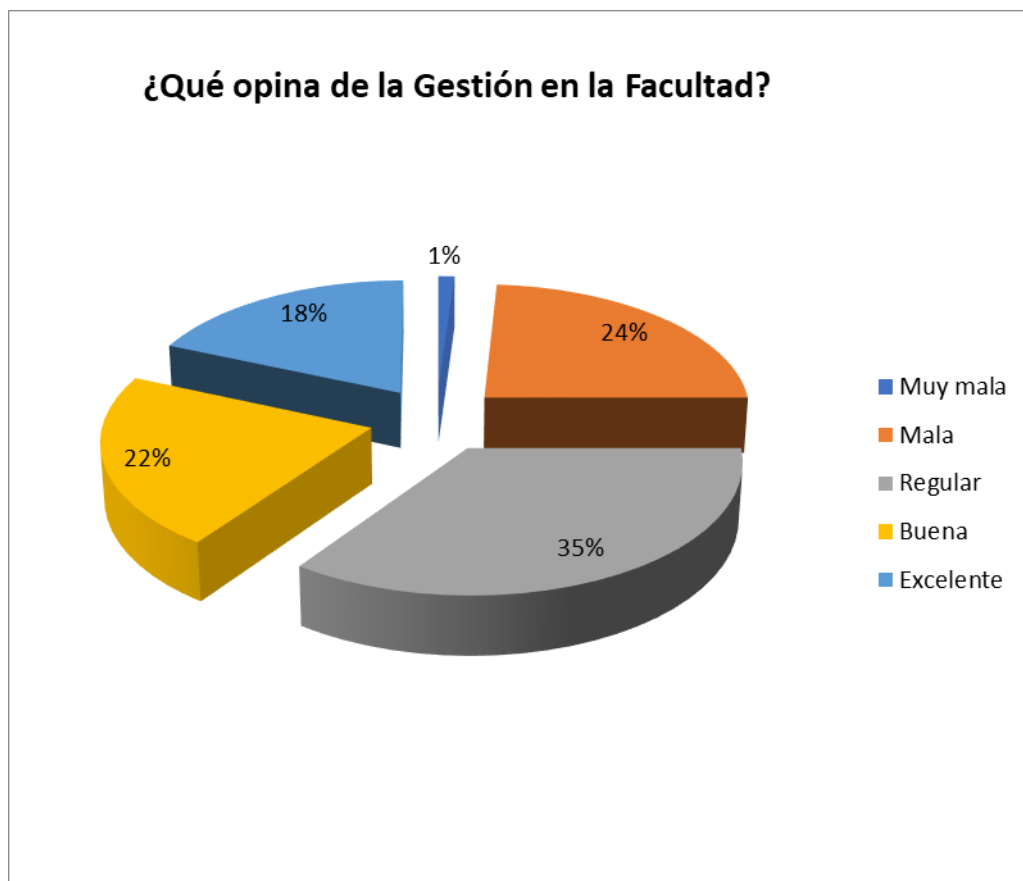
Gráfico 17 Nivel de Implementación de las Tics



Fuente: Propia

INTERPRETACIÓN: Los estudiantes al ser encuestados sobre el nivel de implementación de las Tics, el 59% indicaron que es excelente o buena, 27% indicaron que es regular, el 11% indicaron que es mala y solo el 3% indicaron que es muy mala.

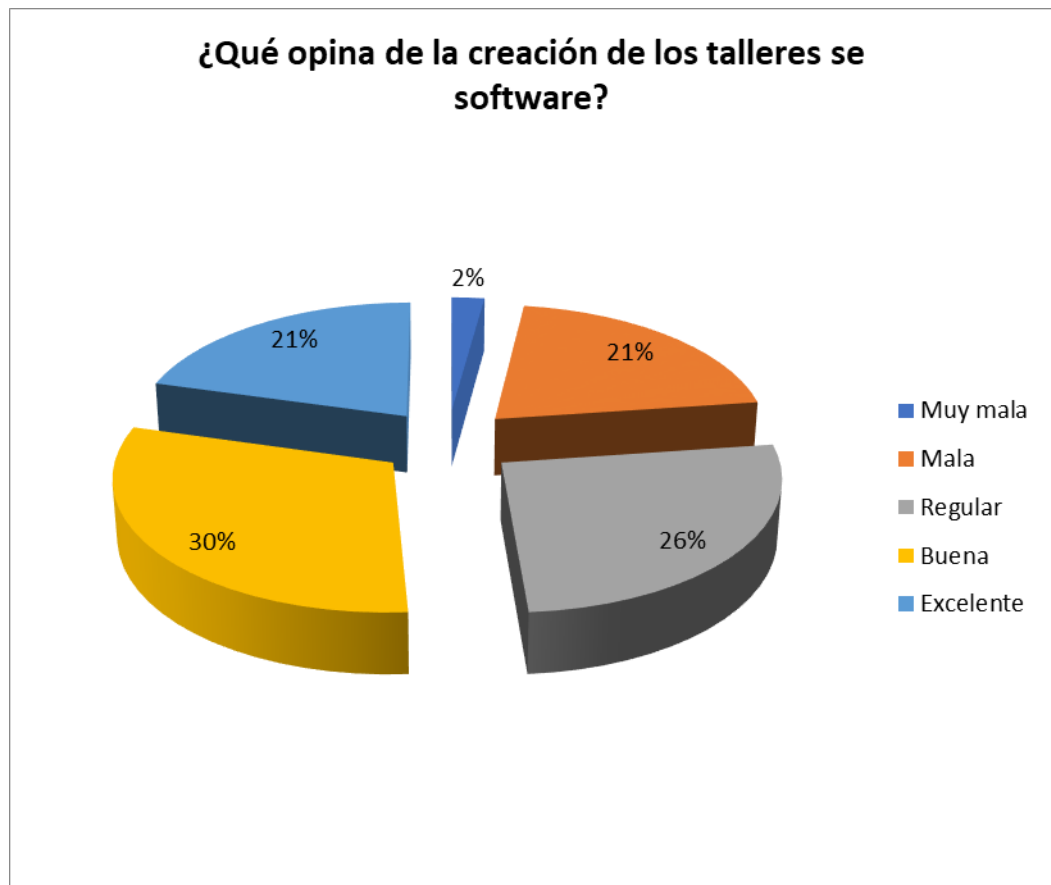
Gráfico 18 Gestión en la facultad



Fuente: Propia

INTERPRETACIÓN: Los estudiantes al ser encuestados sobre la gestión en la Facultad, el 40% indicaron que es excelente o buena, 35% indicaron que es regular, el 24% indicaron que es mala y solo el 1% indicaron que es muy mala.

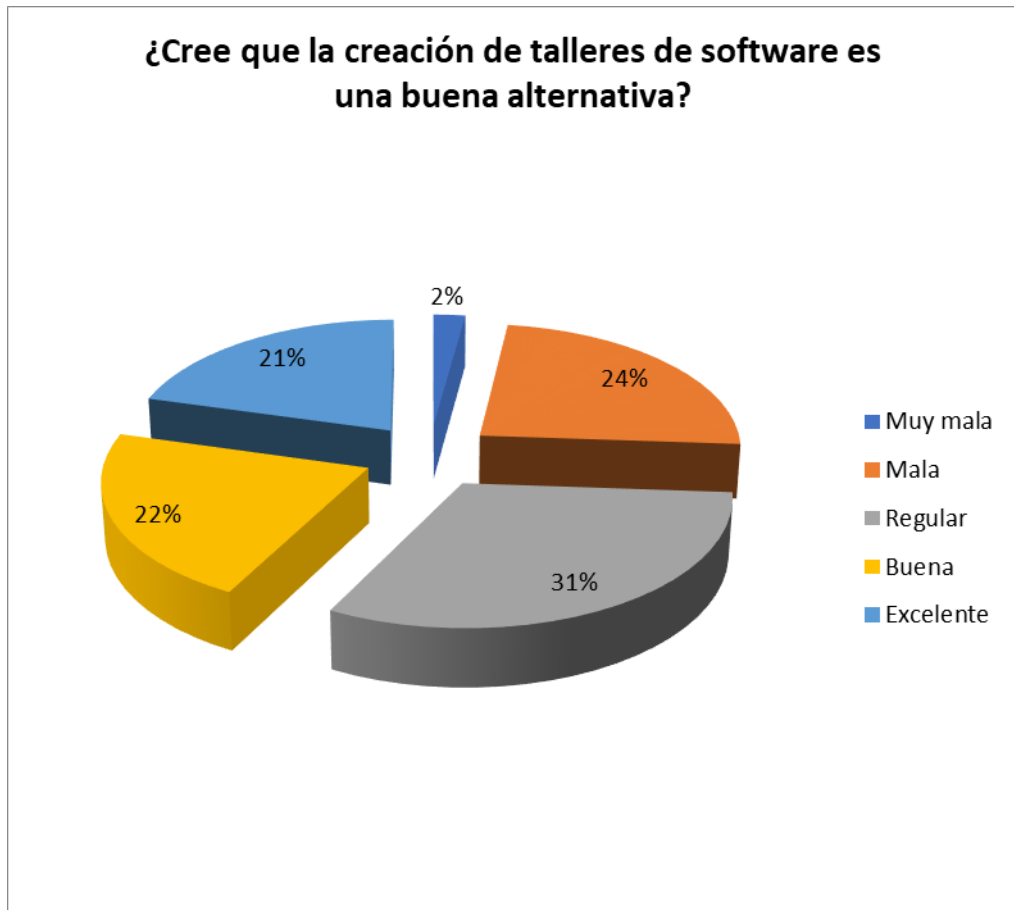
Gráfico 19 Opinión sobre creación de Talleres de Software



Fuente: Propia

INTERPRETACIÓN: Los estudiantes al ser encuestados sobre la creación de los talleres de software, el 51% indicaron que es excelente o buena, 26% indicaron que es regular, el 21% indicaron que es mala y solo el 2% indicaron que es muy mala.

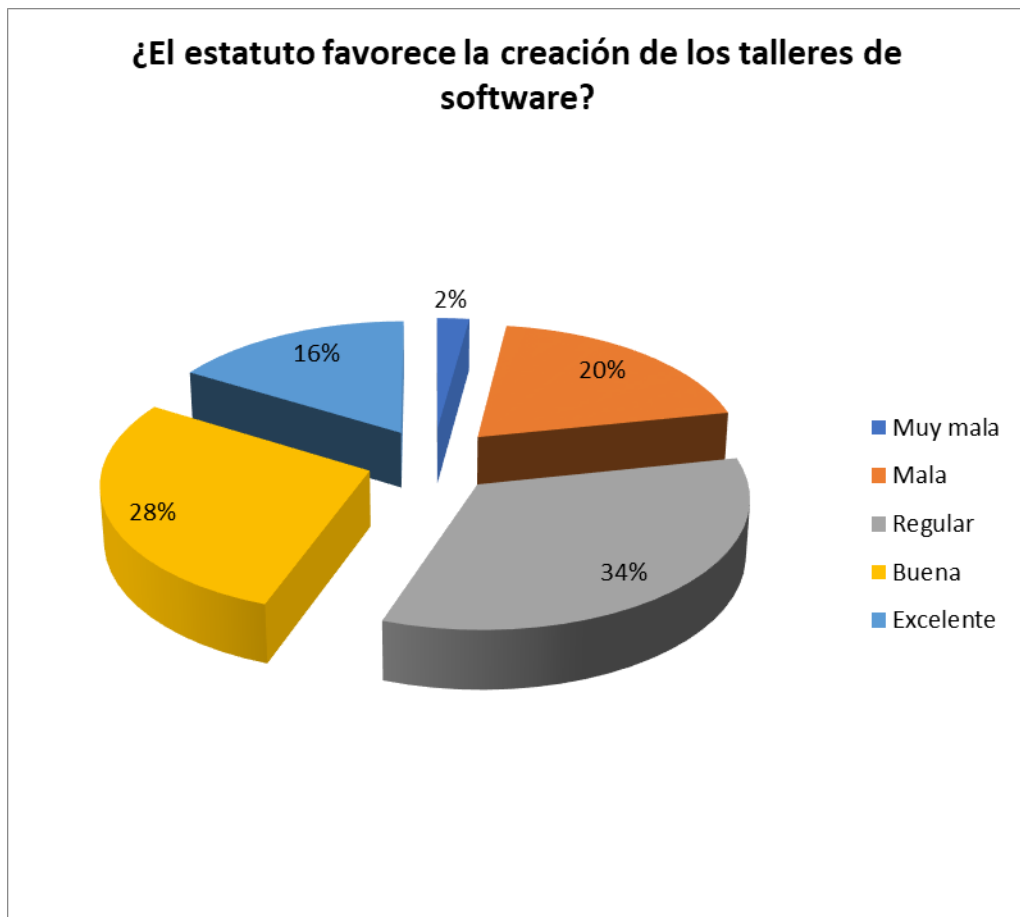
Gráfico 20 Opinión sobre si los Talleres es una buena alternativa



Fuente: Propia

INTERPRETACIÓN: Los estudiantes al ser encuestados sobre si los talleres de software es una buena alternativa, el 43% indicaron que es excelente o buena, 31% indicaron que es regular, el 24% indicaron que es mala y solo el 2% indicaron que es muy mala.

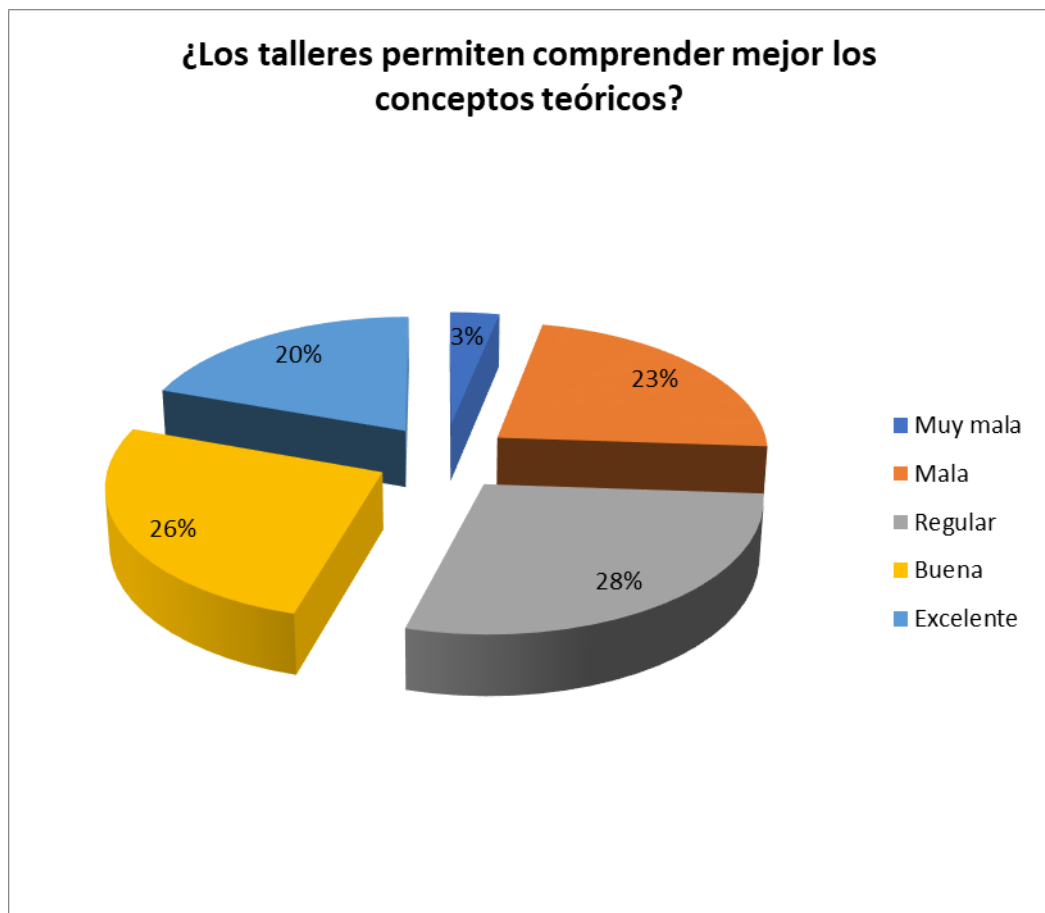
Gráfico 21 El Estatuto y los talleres de Software



Fuente: Propia

INTERPRETACIÓN: Los estudiantes al ser encuestados sobre si el estatuto de la Universidad favorece la creación de los talleres de software, el 44% indicaron que es excelente o buena, 34% indicaron que es regular, el 20% indicaron que es mala y solo el 2% indicaron que es muy mala.

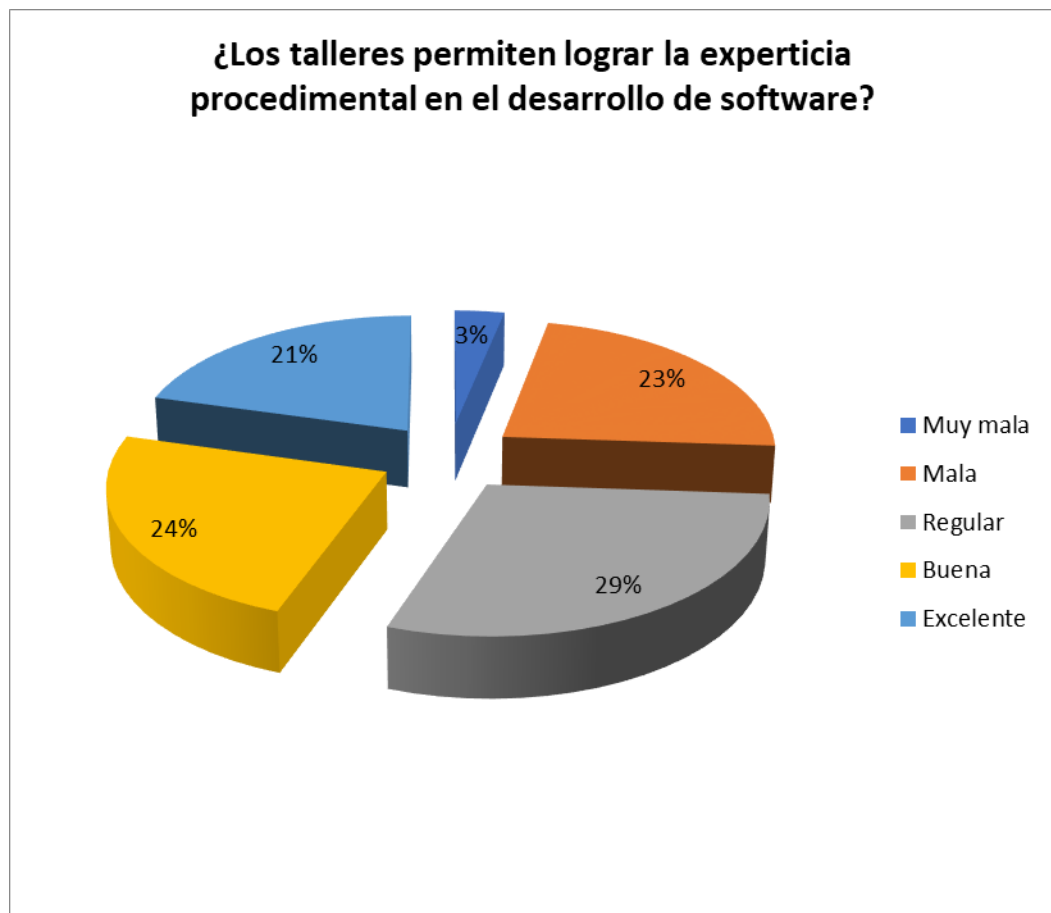
Gráfico 22 Los talleres y los conceptos teóricos



Fuente: Propia

INTERPRETACIÓN: Los estudiantes al ser encuestados sobre si los talleres de software permiten comprender mejor los conceptos teóricos, el 46% indicaron que es excelente o buena, 28% indicaron que es regular, el 23% indicaron que es mala y solo el 3% indicaron que es muy mala.

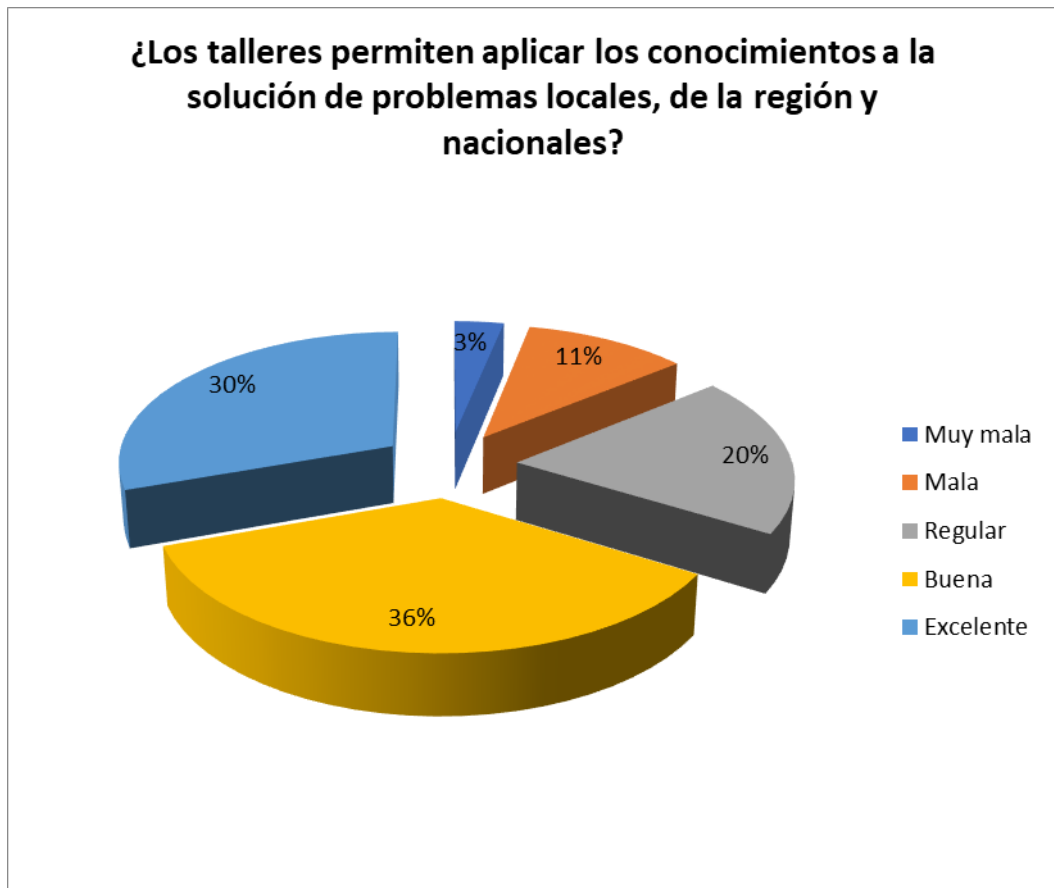
Gráfico 23 Los talleres de software y la experticia



Fuente: Propia

INTERPRETACIÓN: Los talleres de software permiten lograr la experticia procedimental en el desarrollo de software, el 45% indicaron que es excelente o buena, 29% indicaron que es regular, el 23% indicaron que es mala y solo el 3% indicaron que es muy mala.

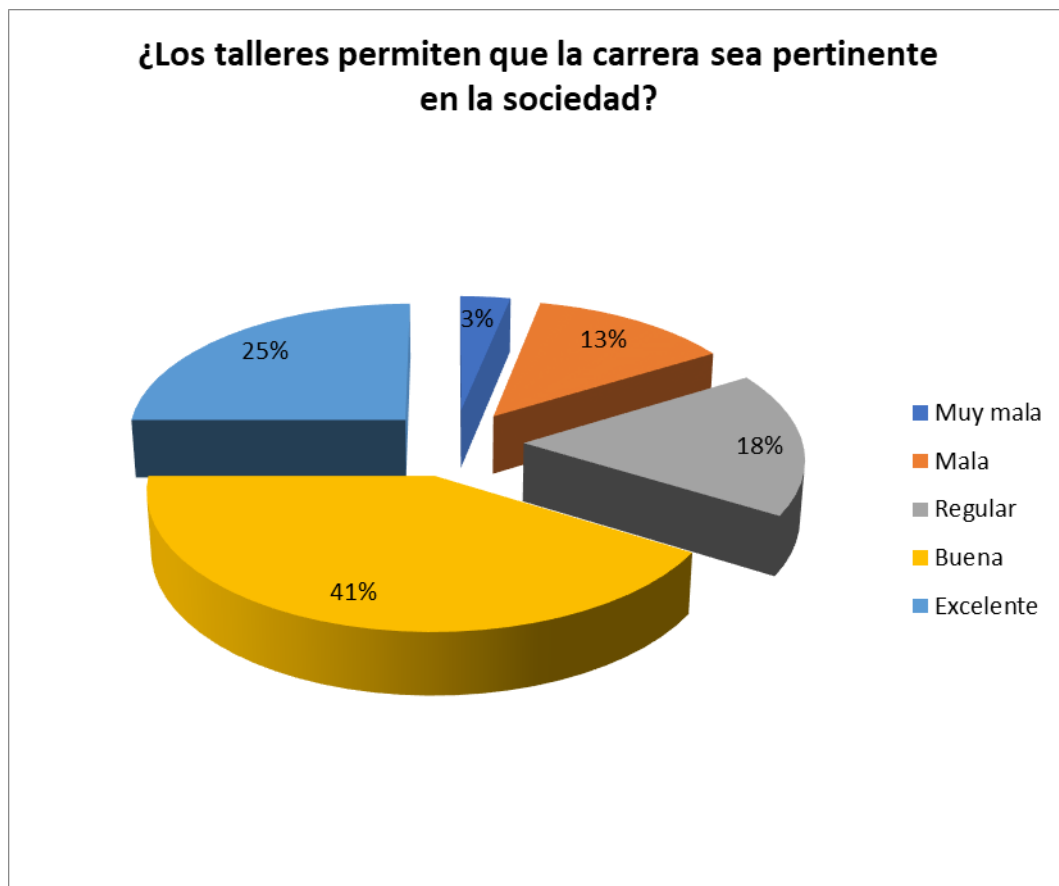
Gráfico 24 Los talleres y la solución de problemas



Fuente: Propia

INTERPRETACIÓN: Los estudiantes al ser encuestados sobre si los talleres de software permiten aplicar los conocimientos a la solución de problemas locales, de la región y nacionales, el 66% indicaron que es excelente o buena, 20% indicaron que es regular, el 11% indicaron que es mala y solo el 3% indicaron que es muy mala.

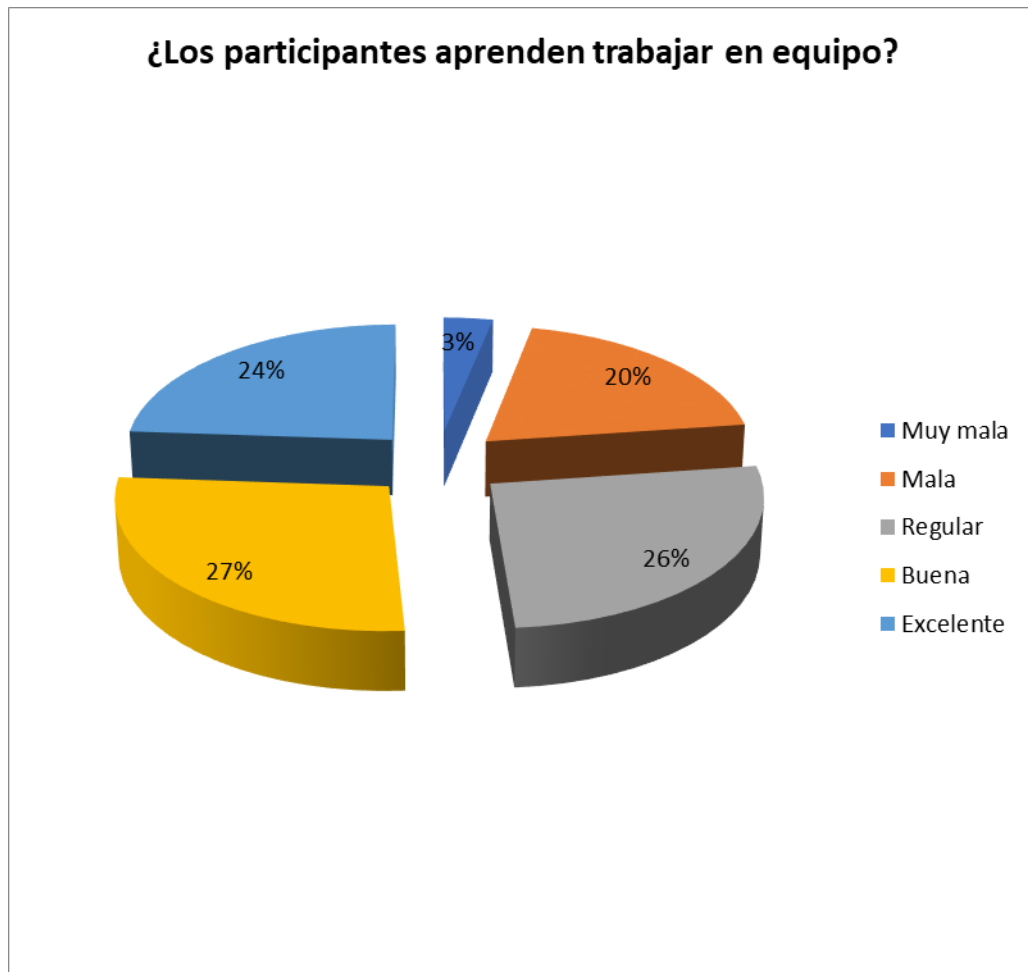
Gráfico 25 Los talleres y la pertinencia



Fuente: Propia

INTERPRETACIÓN: Los estudiantes al ser encuestados sobre si los talleres de software permiten que la carrera sea pertinente en la sociedad, el 66% indicaron que es excelente o buena, 18% indicaron que es regular, el 13% indicaron que es mala y solo el 3% indicaron que es muy mala.

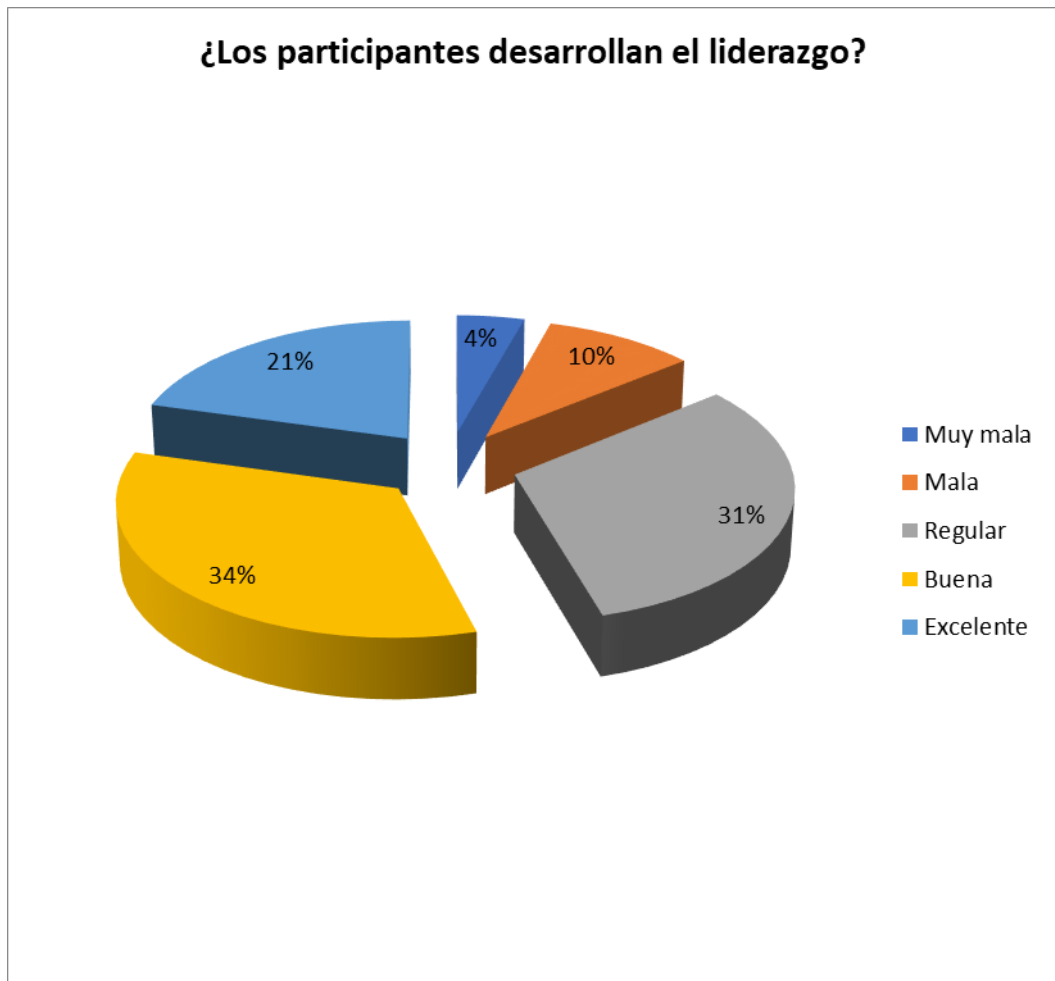
Gráfico 26 Trabajo en equipo



Fuente: Propia

INTERPRETACIÓN: Los estudiantes al ser encuestados sobre si los participantes aprenden trabajar en equipo, el 51% indicaron que es excelente o buena, 26% indicaron que es regular, el 20% indicaron que es mala y solo el 3% indicaron que es muy mala.

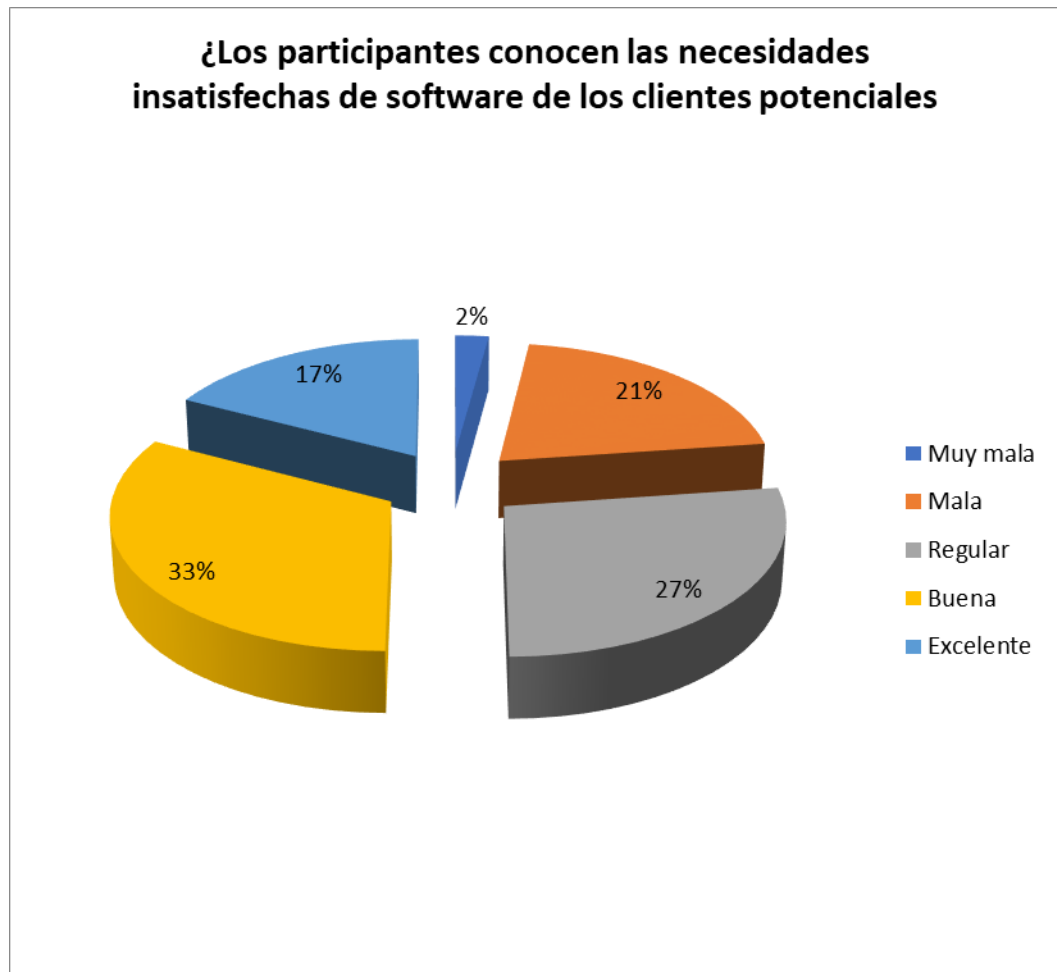
Gráfico 27 El Liderazgo



Fuente: Propia

INTERPRETACIÓN: Los estudiantes al ser encuestados sobre si los participantes desarrollan el liderazgo, el 55% indicaron que es excelente o buena, 31% indicaron que es regular, el 10% indicaron que es mala y solo el 4% indicaron que es muy mala.

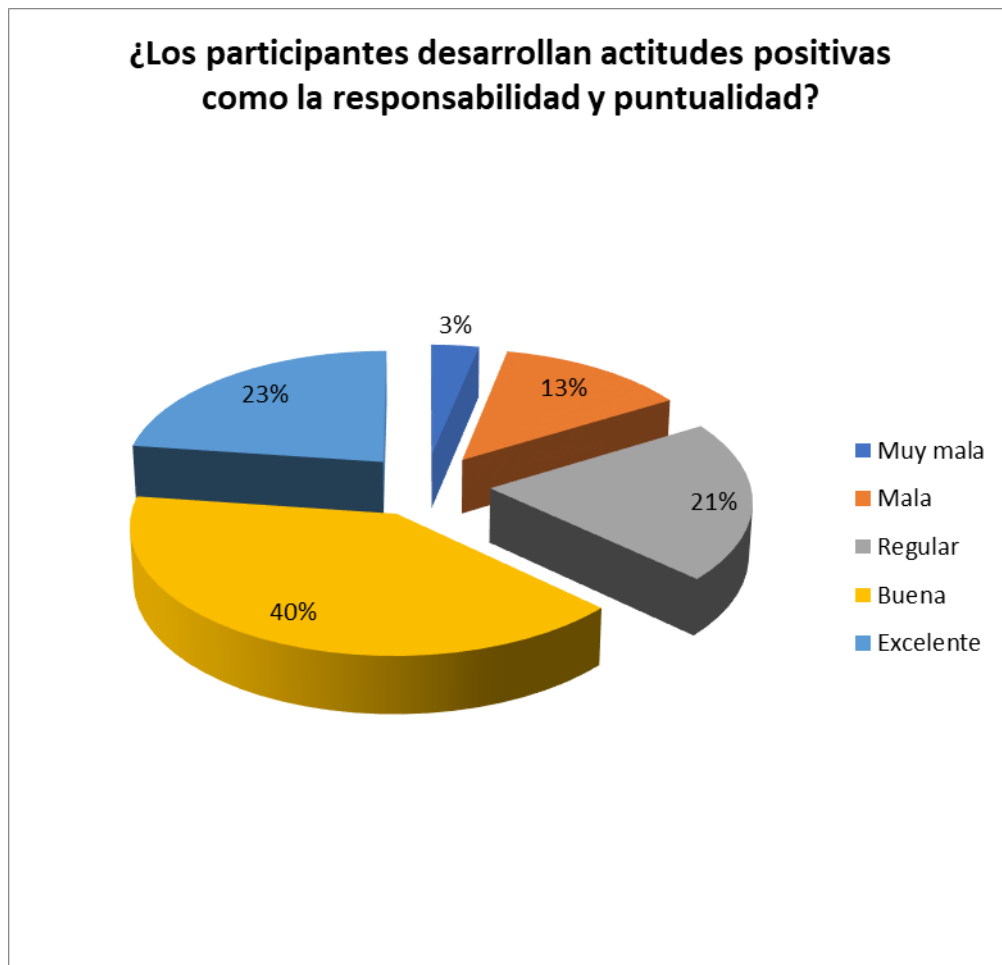
Gráfico 28 Necesidades de los clientes



Fuente: Propia

INTERPRETACIÓN: Los estudiantes al ser encuestados sobre si los participantes conocen las necesidades insatisfechas de software de los clientes potenciales, el 50% indicaron que es excelente o buena, 27% indicaron que es regular, el 21% indicaron que es mala y solo el 2% indicaron que es muy mala.

Gráfico 29 Actitudes positivas



Fuente: Propia

INTERPRETACIÓN: Los estudiantes al ser encuestados sobre si los participantes desarrollan actitudes positivas como la responsabilidad y puntualidad, el 53% indicaron que es excelente o buena, 21% indicaron que es regular, el 13% indicaron que es mala y solo el 3% indicaron que es muy mala.