

UNIVERSIDAD NACIONAL AGRARIA DE LA SELVA

FACULTAD DE INGENIERIA EN INFORMATICA Y SISTEMAS

DEPARTAMENTO ACADEMICO DE CIENCIAS EN INFORMATICA Y SISTEMAS



**HERRAMIENTAS PARA FACILITAR LA IMPLEMENTACION DE LA
ISO/IEC 29110 EN LAS PYMES. CASO: LEAD WORKING PARTNER**

Tesis

Para optar el título de:

INGENIERO EN INFORMATICA Y SISTEMAS

WILLIAM LOPEZ OBREGON

ASESORES:

ING. WALTER RUBÉN BERNUY BLANCO

ING. BRIAN CÉSAR PANDO SOTO

Tingo María - Perú

2016 - 2017

PARTE 1. FASE INICIAL

Siendo las 1:30 horas del día 30 de Mayo de 2017; en la Sala de Grados de la FIIS, se instala el jurado calificador conformado por:

- Jurado 1. MG. WILLIAM R. MARCHAND NIÑO
- Jurado 2. ING. PEDRO TRUJILLO NATIVIDAD
- Jurado 3. MSC NOEL JUIPA CAMPO

Oficializado mediante **Resolución n.º 043-2017-D-FIIS-UNAS** del 21/Marzo/2017, para el proceso de sustentación del informe final de Tesis del bachiller **William LÓPEZ OBREGÓN** titulado: **"HERRAMIENTAS PARA FACILITAR LA IMPLEMENTACIÓN DE LA ISO/IEC EN LAS PYMES. CASO: LEAD WORKING PARTNER"**

Se manifiesta que el bachiller cumple con los requisitos exigidos de Ley y se le invita a disertar su Tesis por espacio de 30 minutos, asimismo se dispondrá de igual tiempo para la absolver preguntas y sugerencias.

PARTE 2. FASE DE PREGUNTAS Y RESULTADO

Culminada la exposición se inicia la fase de preguntas por parte del jurado calificador; también se invita a los asistentes a formular preguntas sobre el tema de Tesis.

Absueltas todas las peticiones, el jurado calificador procede a deliberar en privado la calificación y resultado.

Concluida la deliberación y en presencia del público asistente, el jurado calificador anuncia que el resultado de la Sustentación de Tesis es: APROBADO POR UNANIMIDAD

(NOTA: consignar una de la siguientes: DESAPROBADO, APROBADO POR MAYORIA o APROBADO POR UNANIMIDAD)

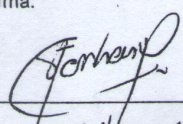
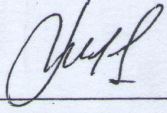
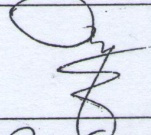

Con calificativo de: BUENO

(NOTA: consignar una de la siguientes: EXCELENTE, MUY BUENO, BUENO, DEFICIENTE, MUY DEFICIENTE)

Por lo que se comunicará a las instancias correspondientes para el trámite respectivo.

PARTE 3. CONFORMIDAD

De todo lo mencionado se firma al pie en señal de conformidad.

Firma: 	Firma: 	Firma: 
Jurado 1: <u>William Marchand Niño</u>	Jurado 2: <u>Noel Juipa Campo</u>	Jurado 3: <u>Pedro Trujillo Natividad</u>
Firma: 		
Sustentante: <u>William Luis Lopez Obregon</u>		

DEDICATORIA

A Dios, porque todo lo que soy se lo debo a él pues me ha impulsado a seguir adelante en todos los momentos de mi vida.

A mis padres, Docentes y amigos por su esfuerzo y guía de enseñanza, brindándome sus conocimientos durante mi etapa de universitario.

AGRADECIMIENTO

A Dios por brindarme la fuerza y sabiduría para culminar mi etapa universitaria, a mi Universidad por permitirme convertirme en un profesional, A mis asesores que sin su apoyo y conocimiento no hubiese podido culminar mi proyecto de tesis, a mis padres por ser mi inspiración de lucha

RESUMEN

Para que una empresa pequeña o mediana compita con calidad debe adoptar modelos de calidad existentes en el mercado, uno de estos modelos esta propuesto en la ISO/IEC 29110, pero la implementación de un estándar internacional como lo es la ISO en la mayoría de casos tiene un costo muy elevado para las medianas y pequeñas empresas además de la gran complejidad de esta, por ello la finalidad de esta investigación es determinar una combinación de herramientas de software de las empresas líderes que cubran los procesos de la ISO/IEC 29110 y que pueda ser fácilmente usado en un proyecto de software.

Para determinar estas herramientas la investigación evalúa dos criterios puntuales, el primero la demanda y popularidad de herramientas en el mercado y el costo de estas herramientas. Así mismo evaluar estas herramientas sobre proceso de gestión e ingeniería de software.

La elección de herramientas pasa por el filtro del cuadrante de Gartner sobre herramientas ALM del 2015, a partir de ello se busca la información utilizando google trends, información oficial de proveedores de las herramientas y otras fuentes secundarias.

Elegido el grupo de herramientas, se aplicó a un caso de desarrollo de software de la empresa Lead Working Partner SAC, al que se realizó un seguimiento y se obtuvo información con resultados bastantes satisfactorios, concluyendo en que existen herramientas de gran demanda y bajo costo que permite a una pyme adoptar la ISO/IEC 29110.

ABSTRACT

For a small or medium enterprise compete with quality should adopt existing models of quality in the market, one of these models is proposed in ISO/IEC 29110, but the implementation of an international standard such as the ISO in most cases has a very high cost for small and medium enterprises in addition to the complexity of this, so the purpose of this research is to determine a combination of software tools of the leading companies covering processes ISO/IEC 29110 and can be easily used in a software project.

To determine these tools the research evaluates two specific criteria, the first the demand and popularity of tools in the market and the cost of these tools. Also evaluate these tools on process management and software engineering.

The choice of tools passes through the Gartner quadrant filter on ALM tools of 2015, from which information is searched using google trends, official information from vendors of tools and other secondary sources.

Once the tool group was chosen, it was applied to a software development case from the company Lead Working Partner SAC, which was monitored and obtained information with quite satisfactory results, concluding that there are high demand and low cost tools that Allows an SME to adopt ISO / IEC 29110.

INDICE

INTRODUCCIÓN	1
I. DESCRIPCION PROBLEMÁTICA.....	5
1.1. PLANTEAMIENTO DEL PROBLEMA	5
1.2. FORMULACION DEL PROBLEMA.....	6
1.3. JUSTIFICACION DEL PROBLEMA	6
1.4. IMPORTANCIA.....	7
1.5. LIMITACIONES.....	7
II. FUNDAMENTO TEÓRICO.....	9
2.1. ANTECEDENTES	9
2.1.1. En Colombia, Boyacá	9
2.1.2. En Lima, Santiago de Surco.....	10
2.1.3. En Arequipa, Cercado	11
2.2. MARCO TEÓRICO	13
2.2.1. ISO/IEC 29110.....	13
2.2.2. PEQUEÑA Y MICRO EMPRESA (PYME).....	14
2.2.3. PROCESOS DE SOFTWARE.....	15
2.2.4. FASES DE DESARROLLO DE SOFTWARE.....	17
2.2.5. HERRAMIENTAS DE INGENIERIA DE SOFTWARE	21
2.2.6. CUADRANTE DE GARTNER.....	22
2.3. DEFINICIONES OPERACIONALES.....	25
2.3.1. UTILIZACION.....	25
2.3.2. COSTOS.....	25
III. OBJETIVOS E HIPÓTESIS.....	27
3.1. OBJETIVOS	27
3.2. HIPÓTESIS	27
IV. METODOLOGÍA DE LA INVESTIGACIÓN	29
4.1. MÉTODO	29
4.2. TÉCNICA	29
4.3. UNIVERSO Y MUESTRA.....	30
V. ANÁLISIS SITUACIONAL DE LA INDUSTRIA DEL SOFTWARE.....	31
5.1. LA INDUSTRIA DEL SOFTWARE EN EL MUNDO	31
5.2. LA INDUSTRIA DEL SOFTWARE EN AMÉRICA LATINA	32
5.3. LA INDUSTRIA DEL SOFTWARE EN EL PERÚ	34

5.4.	MODELOS DE MEJORA DE PROCESOS PARA PYMES	38
VI.	SELECCIÓN DE HERRAMIENTAS	41
6.1.	IDENTIFICACIÓN DE HERRAMIENTAS	41
6.2.	CARACTERÍSTICAS DE LAS HERRAMIENTAS	42
6.2.1.	TEAM FOUNDATION SERVICE (MICROSOFT)	42
6.2.2.	VERSIONONE	47
6.2.3.	ATLASSIAN (JIRA)	50
6.2.4.	IBM (RATIONAL)	53
6.2.5.	RALLY	56
6.3.	COMPARACIÓN DE HERRAMIENTAS	61
VII.	COBERTURA DE LAS HERRAMIENTAS SELECCIONADA EN LOS PROCESOS DE LA ISO/IEC 29110	63
7.1.	MAPEO DE ARTEFACTOS	63
7.2.	MAPEO DE PROCESOS	66
7.3.	DEFINICIÓN DE LA COMBINACIÓN DE HERRAMIENTAS	67
VIII.	CASO DE ESTUDIO	69
8.1.	CARACTERÍSTICAS DEL PROYECTO DE APLICACIÓN	69
8.1.1.	DEL PROYECTO	69
8.1.2.	DEL SOFTWARE	70
8.2.	DISEÑO Y PLANIFICACION DEL CASO	70
8.2.1.	Exposición de la ISO/IEC 29110 y sus respectivas Herramientas	70
8.2.2.	Diseño de la ejecución encuesta	71
8.2.3.	Análisis estadístico	71
8.3.	RESULTADOS DEL CASO	71
IX.	DISCUSIÓN	79
9.1.	ARTEFACTOS CUBIERTOS	79
9.2.	VALORACIÓN DE HERRAMIENTAS EN LOS ARTEFACTOS	82
9.3.	VALORACIÓN DE LAS HERRAMIENTAS EN LOS PROCESOS	84
9.4.	ANALISIS DE MEDIAS DE ARTEFACTOS Y PROCESOS DE LA ISO/IEC 29100	85
X.	CONCLUSIONES	86
XI.	RECOMENDACIONES	88
XII.	BIBLIOGRAFÍA	89
XIII.	ANEXO	92
13.1.	GRUPO DE PROCESOS DE GESTION DE LA ISO/IEC 29110	92
13.2.	GRUPO DE PROCESOS DE INGENERIA DE LA ISO/IEC 29110	93

13.3.	RESULTADO DE EVALUACIÓN DEL GESTOR DE PROYECTO	94
13.4.	RESULTADO DE EVALUACIÓN DEL ANALISTA	95
13.5.	RESULTADO DE EVALUACIÓN DEL DISEÑADOR	95
13.6.	RESULTADO DE EVALUACIÓN DEL PROGRAMADOR.....	96
13.7.	RESULTADO DE EVALUACIÓN DEL LIDER TÉCNICO	96
13.8.	RESULTADO DE EVALUACIÓN DEL CLIENTE	97
13.9.	MATRIZ DE CONSISTENCIA	98

INDICE DE FIGURA

Figura 1: Fases del desarrollo de proceso de software.....	21
Figura 2: Cuadrante de Gartner.....	24
Figura 3: Demanda de software.....	32
Figura 4: Pronósticos de adquisición de software en Latino América.....	33
Figura 5: Empresas de TI en Brasil.....	33
Figura 6: Empresas de Software en Argentina.....	34
Figura 7: Destinos de Venta de Software 2007.....	35
Figura 8: Mercado de Software Peruano al 2009.....	37
Figura 9: Cuadrante de Gartner ALM.....	41
Figura 10: Screenshot de TFS.....	43
Figura 11: Utilización de TFS.....	46
Figura 12: Resultados de búsquedas en Google.....	47
Figura 13: Vistas de Versión ONE.....	47
Figura 14: Utilización de VERSIONONE.....	49
Figura 15: Búsquedas de Versión ONE.....	50
Figura 16: Vistas de JIRA.....	50
Figura 17: Tendencia de JIRA.....	52
Figura 18: Búsquedas de JIRA.....	53
Figura 19: Vistas de Team Concert.....	53
Figura 20: Tendencia de Team Concert.....	55
Figura 21: Búsquedas de Team Concert.....	56
Figura 22: Vista 1 de RALLY.....	56
Figura 24: Vista 3 de RALLY.....	57
Figura 23: Vista 2 de RALLY.....	57
Figura 25: Vista 4 de RALLY.....	58
Figura 26: Tendencias de RALLY.....	60
Figura 27: Búsquedas de RALLY.....	61
Figura 28: Vistas del Drive del proyecto.....	75
Figura 30: Vista 2 de TFS en el proyecto.....	76
Figura 29: Vista 1 de TFS en el proyecto.....	76
Figura 32: Vista 4 de TFS en el proyecto.....	77
Figura 31: Vista 3 de TFS en el proyecto.....	77
Figura 33: Vistas del código e integración del proyecto.....	78
Figura 34: Vistas del código e integración del proyecto.....	78
Figura 35: Artefactos cubiertos en el proyecto.....	79
Figura 36: Procesos cubiertos en el proyecto.....	81
Figura 37: Valoración de artefactos de gestión ISO/IEC 29110.....	82
Figura 38: Valoración de artefactos de implementación ISO/IEC 29110.....	83
Figura 39: Valoración de procesos de gestión ISO/IEC 29100.....	84
Figura 40: Valoración de procesos de implementación ISO/IEC 29110.....	84

INDICE DE TABLAS

Tabla 1: Operación de variables	28
Tabla 2: Herramientas ALM	42
Tabla 3: Características <i>TEAM FOUNDATION SERVICE (TFS)</i>	44
Tabla 4: Licencias TFS	45
Tabla 5: Características Versión ONE	48
Tabla 6: Licencias Versión ONE	48
Tabla 7: Características JIRA	51
Tabla 8: Licencias JIRA	51
Tabla 9: Características Team Concert	54
Tabla 10: Pagos de Licencias Team Concert	54
Tabla 11: Características RALLY	58
Tabla 12: Licencias RALLY	59
Tabla 13: Comparación de herramientas	61
Tabla 14: Definición de herramientas	62
Tabla 15: Mapeo de Artefactos	63
Tabla 16: Herramienta de almacenamiento de archivos	65
Tabla 17: Herramientas complementarias	65
Tabla 18: Herramientas de Diseño e Integración	66
Tabla 19: Mapeo de procesos	66
Tabla 20: Roles según la ISO/IEC 29110	69
Tabla 21: Requisitos de software	70
Tabla 22: Resultados de caso	72
Tabla 23: Calificación de procesos	73
Tabla 24: Artefactos de la ISO/IEC 29110 Cubiertos en el proyecto	80
Tabla 25: Procesos de la ISO/IEC 29100 cubiertos en el proyecto	81
Tabla 26: Análisis de Medias de procesos y artefactos de la ISO/IEC 29110 cubiertos en el proyecto	85

INTRODUCCIÓN

Las empresas dedicadas al software deben competir en el mercado usando procesos estándares y apostando por usar procesos de calidad para generar productos de calidad; los estándares propuestos inicialmente por el estado peruano no se ajustaban a las necesidades de una empresa pequeña o mediana, por lo que se propuso la NTP-RT-ISO/IEC TR 29110, la cual puede considerarse una interpretación de la ISO/IEC 29110 orientada para pequeñas organizaciones, así, esta norma sugiere los procesos y los artefactos que deben conseguirse en un proyecto de software. Basada en esta interpretación, esta investigación propone una guía de herramientas combinadas que permitirá a las pequeñas y medianas empresas desarrolladoras, cubrir los procesos y artefactos que refiere la norma.

Esta investigación es necesaria para las micro-empresas que están en la industria del desarrollo de software, porque, comprendiendo los resultados que se esperan obtener, las empresas tendrán la posibilidad de recurrir a estas herramientas para la implementación de la norma ISO/IEC 29110, lo que les permitirá lograr la mejora de procesos y con ello optimizar su desempeño en el mercado.

La presente tesis, presenta la propuesta de herramientas para el desarrollo de Software de empresas líderes, que cubran los procesos de la implementación de la ISO/IEC 29110, teniendo en cuenta los costos y tendencia de uso de estas. Para este propósito se tomó como caso de estudio el desarrollo de un software en la empresa Lead Working Partner SAC.

El informe de tesis está estructurado en nueve capítulos, cuyo contenido es el siguiente:

En el capítulo 1, denominado descripción problemática, se describe el planteamiento del problema, justificación del problema, importancia y limitaciones de la presente investigación de tesis.

En el capítulo 2, contiene los fundamentos teóricos que sustentan la investigación, como lo son antecedentes de investigaciones similares al proyecto a nivel local, regional e internacional; como también el marco teórico de conceptos relacionados con la ISO/IEC 29110 y la Ingeniería de Software.

En el capítulo 3, Se presenta brevemente los objetivos de la investigación además de formular la hipótesis: “La implementación de las herramientas de ingeniería de Software seleccionadas, permitirá cumplir los procesos contemplados en la ISO/IEC 29110, atendiendo los criterios de costo y utilización para el desarrollo de software en una PYME.”.

En el capítulo 4, denominado metodología de la investigación, menciona las características de la investigación, como el nivel y tipo de investigación, técnicas, universo y muestra.

En el capítulo 5, se detalla cómo se encuentra la situación actual de la industria del software en el mundo, América latina y en el Perú, demostrando ser un sector económico muy importante en la importación y exportación de software.

El Perú cuenta con un activo capital humano de desarrolladores de software capacitados, con empresas pequeñas y microempresas, dedicadas al desarrollo del software.

En el capítulo 6, denominado selección de herramientas, detalla la identificación y características de las herramientas para el desarrollo de software de empresas líderes en el mercado, obtenidos del cuadrante Gartner del 2015.

En el capítulo 7, denominado cobertura de las herramientas seleccionada en los procesos de la ISO/IEC 29110, se detalla los artefactos, procesos que contiene la ISO/IEC 29110 con su respectiva evaluación, para la selección de las herramientas de Ingeniería de software.

En el capítulo 8, denominado caso de estudio, se menciona las características del proyecto del software, donde se aplicó las herramientas seleccionadas para verificar el cumplimiento con los artefactos y procesos de la ISO/IEC 29100 mediante un cuestionario de satisfacción por parte del equipo del proyecto.

Finalmente en el capítulo 9, se menciona sobre la discusión de los resultados obtenidos durante la investigación, en los que podemos mencionar la valoración de los artefactos, procesos de la ISO/IEC 29110, que han sido cubiertos por las herramientas de ingeniería de software seleccionadas.

Por otro lado en el aspecto específico del caso de estudio de la empresa Lead Working Partner SAC, ya cuenta con un método ágil de desarrollo de Software así como la flexibilidad de poder adaptarse a nuevas herramientas de desarrollo de software, además es importante mencionar que la empresa es considerada una PYME en la industria del software.

De hecho, el determinar la combinación de herramientas que cubran los procesos y artefactos exigidos por la ISO/IEC 29110 es importante para la obtención de procesos de software de calidad en las empresas dedicadas a este sector, aunque es necesario tener en cuenta aspectos como tendencia de uso y costo de estas herramientas, los cuales sean accesibles para las PYMES.

I. DESCRIPCION PROBLEMÁTICA

1.1. PLANTEAMIENTO DEL PROBLEMA

La implementación de un estándar internacional como lo es la ISO en la mayoría de casos tiene un costo muy elevado para las medianas y pequeñas empresas además de la gran complejidad de esta. Asimismo, si alguien accede a leer el documento NTP-RT-ISO/IEC TR 29110 – Perfiles del ciclo de vida para las pequeñas organizaciones (PO) de INDECOPI, se observa que existe un punto “Herramientas” en el documento y, sin embargo, en este punto solo se muestra un alcance que las herramientas deberían tener para cumplir con los objetivos de la norma, más no brinda mayores detalles de la herramienta que se debería emplear para lograr realizar los pasos definidos en cada proceso de la norma.

En el caso particular de la empresa *Lead Working Partner – LWP SAC* el cual es una pyme orientada al desarrollo de software que está en proceso de implementación de la ISO/IEC 29110 – “Ingeniería de Software — Perfiles de ciclo de vida para pequeñas organizaciones (VSEs)”, se ha verificado que el problema principal es la ausencia de herramientas de soporte para los procesos de la norma ISO/IEC 29110. Por consiguiente, la experiencia demuestra que se

hace necesario e indispensable buscar las herramientas que faciliten la adopción de la ISO.

1.2. FORMULACION DEL PROBLEMA

La adopción de la ISO/IEC 29110 es realmente preocupante en el aspecto financiero en las empresas pequeñas; que apuntan al mercado internacional y a competir con otras empresas de esta industria del desarrollo de software, por lo que se debe buscar herramientas de software de empresas líderes que faciliten la adopción de este estándar, en esta búsqueda de soluciones se generan las preguntas:

- **Problema General**

¿Cuáles son las herramientas de ingeniería de software producidas por las empresas líderes del mercado que cubran los procesos de la ISO/IEC 29110 en su implementación, atendiendo a criterios de costo y utilización?

1.3. JUSTIFICACION DEL PROBLEMA

- **Justificación Técnica**

La principal justificación de la presente investigación es el aporte en el ámbito de la Ingeniería de Software con la determinación de herramientas de software para la implementación de la ISO/IEC 29110 en las PYMES (Pequeñas y medianas empresas) del sector de la Industria del Software, para lo cual se aprovechará la disponibilidad de un caso real donde se podrá evaluarlos y

seleccionarlos. Esta propuesta nace a partir de la necesidad de contar con herramientas de software fabricadas o producidas por empresas líderes en TI y/o Desarrollo Software que nos permita cubrir los procesos en la implementación de la ISO/IEC 29110, a la vez teniendo en cuenta los costos y tendencias de uso de estas herramientas.

1.4. IMPORTANCIA

Esta investigación es importante para las micro-empresas que están en la industria del desarrollo de software porque a través de un caso de estudio se podrán determinar las herramientas más apropiadas, desde el punto de vista financiero y de la demanda, y por consecuencia, las empresas podrán recurrir a herramientas apropiadas y eficientes para la implementación de la norma ISO/IEC 29110 en el desarrollo de software y podrán lograr mejorar sus procesos y con ello su rendimiento y desempeño en el mercado nacional e internacional.

1.5. LIMITACIONES

No existen, en el mercado inmediato muchas empresas que apliquen completamente la norma ISO/IEC 29110 para asegurar la calidad de sus procesos, como consecuencia hay poca disponibilidad de información que pudiera servir como referencia que apoye esta investigación, ante ello, la recopilación de información se hace a través de internet y, como aplicación, se desarrolla un caso de estudio en un proyecto pequeño.

Debe considerarse también como limitación la actitud del empresariado en relación con los conocimientos generados en sus empresas, es decir el *know how* de las empresas que no es fácilmente distribuido, sino por el contrario, se guarda celosamente como tesoro tecnológico que le permitirá alcanzar o mantener la vanguardia empresarial.

II. FUNDAMENTO TEÓRICO

2.1. ANTECEDENTES

2.1.1. En Colombia, Boyacá

Título: Guía para pymes desarrolladoras de software, basada en la norma ISO/IEC 155041. (Alarcón Aldana, González Sanabria, & Rodríguez García, 2011)

Autor(es): Alarcón Aldana, Andrea Catherine; González Sanabria, Juan Sebastián; Rodríguez Torres, Sandra Lucía.

Año: 2011.

Clasificación: Artículo de Revista Virtual

Institución: Universidad Pedagógica y Tecnológica de Colombia.

Resumen: Esta investigación, mediante un método analítico y sintético, describe la norma ISO/IEC 15504 y algunos aspectos por tener en cuenta para implementarla, así como las fases de certificación de acuerdo con la misma, el proceso de auditoría, los criterios de evaluación, entre otros ítems relevantes. Como resultado, se presenta una guía orientada a las empresas desarrolladoras de software, con el fin de enfocarlas en un marco de trabajo que les permita iniciar y desarrollar un proceso de certificación bajo la norma ISO/IEC 15504.

2.1.2. En Lima, Santiago de Surco

Título: Implementación de herramientas de soporte para los procesos de la norma ISO/IEC 29110 – perfil básico. (Cuentas Morant & Ramos Castro, 2015)

Autor(es): Cuentas Morant, Ricardo M.; Ramos Castro, José L.

Año: 2015

Clasificación: Informe de Tesis Pregrado

Institución: Universidad Peruana de Ciencias Aplicadas - UPC

Resumen: El propósito de este proyecto, es proponer un conjunto de herramientas de soporte para los procesos de la norma ISO/IEC 29110, para que sean implementadas por las Pequeñas Organizaciones que desarrollen proyectos de software, ya que en la actualidad se reconoce el aporte que realizan estas con respecto a la producción de software, pues cumplen el rol de proveedores de software para medianas y grandes empresas. Se concluyó que para la proposición de herramientas aplicables en la Norma ISO/IEC, se hizo una investigación comparativa de las diferentes herramientas para la gestión de los proyectos de Software. Las empresas elegidas para la comparación fueron extraídas del cuadrante de Gartner del 2012. Ante lo expuesto se compararon las herramientas de las empresas Microsoft, IBM y Atlassian, ya que eran las tres primeras empresas líderes según el cuadrante de Gartner. Después de realizar las comparaciones se eligió como herramienta principal a Jira de la empresa Atlassian.

2.1.3. En Arequipa, Cercado

Título: Implementación del Perfil Básico de la ISO/IEC 29110 de una Pequeña Empresa Desarrolladora de Software: Lecciones Aprendidas. (Abarca Romero, Arisaca Mamani, & Dávila Ramón, 2015)

Autor(es): Abarca Romero, Melissa; Arisaca Mamani, Robert; Dávila Ramón, Abraham.

Año: 2015

Clasificación: Artículo de Revista Virtual

Institución: Universidad Nacional de San Agustín; Pontificia Universidad Católica del Perú.

Abstract: En este artículo se presenta la experiencia de implementación y lecciones aprendidas en la mejora de los procesos del Perfil Básico de la ISO/IEC 29110 de una pequeña empresa desarrolladora de software de Arequipa-Perú. En este trabajo se utilizó Investigación-Acción dentro del marco de un programa de pruebas controladas de adopción del dicho estándar. Los principales resultados de este trabajo son una lista de problemas identificados y acciones realizadas para resolverlas, así como las lecciones aprendidas que representan el conjunto de conocimiento que por experiencia se ha logrado consolidar para el caso específico del Perfil Básico en desarrollo de software de la ISO/IEC 29110.

Luego de revisado las investigaciones anteriores en Colombia se describen los aspectos importantes a tener en cuenta para la implementación de la ISO, en Lima se definen herramientas basado en el cuadrante de Gartner del 2012, y en Arequipa se describen las lecciones aprendidas al aplicar las herramientas en un escenario controlado.

La presente propuesta se basa en la definición actual del cuadrante, apareciendo nuevas alternativas de herramientas que deben ser parte de la comparación, sin embargo debe verificarse si las nuevas herramientas pueden satisfacer las necesidades de las Pymes.

Por otro lado las implementaciones realizadas en las investigaciones realizadas ocurren en ciudades grandes como Lima y Arequipa, por ello se necesita validar cual es el impacto cuando se aplica a una cultura de una ciudad pequeña, de manera que se permita descubrir si en cualquier entorno estas herramientas son satisfactorias.

2.2. MARCO TEÓRICO

2.2.1. ISO/IEC 29110

Esta Norma publicada en mayo del 2011, está dirigido a las organizaciones pequeñas de la industria de Software, que según el documento del estándar indica textualmente:

“La Industria del Software reconoce el valor de las aportaciones de productos y Servicios de las Organizaciones Pequeñas (Very Small Entities (VSEs)). Una Organización Pequeña (VSE) es una entidad (empresa, organización, departamento o proyecto) conformada hasta 25 personas. Las Organizaciones Pequeñas (VSEs) también desarrollan y/o mantienen software que es usado en Sistemas más grandes; por ello, el reconocimiento de las Organizaciones Pequeñas como proveedores de Software de Calidad es requerido”.

La ISO/IEC 29110 es un estándar internacional con el título Ingeniería de Software - Perfiles de ciclo de vida para pequeñas organizaciones (VSEs) que contiene un conjunto de documentos (estándares y reportes técnicos) y que propone perfiles de procesos (entrada, básico, intermedio y avanzado) que las VSEs pueden certificarse, exceptuando el de entrada, y que según (Abarca Romero, Arisaca Mamani, & Dávila Ramón, 2015), la ISO/IEC 29110 es un modelo que se basa en la ISO/IEC 12207, la ISO/IEC 15289 y toma de MoProSoft aspectos fundamentales en su organización en detalle. A nivel de perfiles, se define uno denominado Perfil de Entrada que corresponde con lo esperado por una organización pequeña que inicia sus operaciones. Para el

Perfil Básico se ha establecido dos procesos principales: Gestión de Proyecto e Implementación de Software.

De la misma forma en el Artículo (Abarca Romero, Arisaca Mamani, & Dávila Ramón, 2015), se describe los problemas y lecciones aprendidas en el proceso de implementación y certificación a nivel del Perfil Básico en desarrollo de software del estándar ISO/IEC 29110, y donde se menciona que entre los principales beneficios esperados al aplicar la ISO/IEC 29110, según lo establecido en el propio estándar, se tienen: (i) mejorar la calidad del software producido por la organización que adopta el modelo; (ii) elevar la capacidad de las organizaciones para ofrecer servicios con calidad y alcanzar niveles internacionales de competitividad; (iii) permitir la fácil adopción del modelo pues se brinda mayor detalle en la información; (iv) facilitar la migración de organizaciones que han implementado modelos como MoProSoft o ISO 9000; y (v) permitir el posicionamiento de las empresas que lo implementan destacando sobre sus competidores y en beneficio de sus clientes.

2.2.2. PEQUEÑA Y MICRO EMPRESA (PYME)

Como se mencionó anteriormente, en el contexto de la ISO/IEC 29110 define a las VSEs (*Very Small Entities*) como pequeñas organizaciones a aquella entidad (empresa, organización, departamento o proyecto) de hasta 25 personas que se dedica al desarrollo y/o mantenimiento de software; entonces en el Perú de VSE equivaldría al concepto de Pequeña y Micro Empresa (PYME) la cual es considerada, según la Legislación Peruana (Ley N° 28015) como: “la

unidad económica constituida por una persona natural o jurídica, bajo cualquier forma de organización o gestión empresarial contemplada en la legislación vigente, que tiene como objeto desarrollar actividades de extracción, transformación, producción, comercialización de bienes o prestación de servicios debiendo contar con las siguientes características:

Microempresa:

- *Número total de trabajadores entre uno (1) y diez(10)*
- *Niveles de ventas anuales no mayores a 150 Unidades Impositivas Tributarias - UIT*

Pequeña empresa

- *Número total de trabajadores hasta un máximo de cincuenta (50)*
- *Niveles de ventas anuales desde 150 hasta 850 Unidades Impositivas Tributarias – UIT.”*

Tomando en cuenta los párrafos anteriores, se puede concluir que una VSE es equivalente a PYME en el Perú, y para efectos de la presente investigación y se tomara como caso de estudio a la empresa Lead Working Partner, el cual es una PYME dedicada al desarrollo y mantenimiento de Software.

2.2.3. PROCESOS DE SOFTWARE

Existe una amplia información bibliográfica en el mundo sobre la definición de ¿qué es un proceso?, pero hay una definición interesante que según en (Juan, 2011) se menciona:

“Proceso es un conjunto de actividades, interacciones y recursos con una finalidad común: transformar las entradas en salidas que agreguen valor a los clientes. El proceso es realizado por personas organizadas según una cierta estructura, tienen tecnología de apoyo y manejan información...”

...Las entradas y Salidas incluyen tránsito de información y de productos”

Por lo tanto, teniendo en cuenta la definición anterior, un Proceso de Software es una serie de actividades relacionadas que tiene por finalidad la fabricación o elaboración de un producto de software. Según (Sommerville, 2011) existen muchos diferentes procesos de software, pero todos deben incluir cuatro actividades que son fundamentales para la ingeniería de software:

- *Especificación del software*, tienen que definirse tanto la funcionalidad del software como las restricciones de su operación.
- *Diseño e implementación del software*, debe desarrollarse el software para cumplir las especificaciones.
- *Validación de software*, hay que validar el software para asegurarse de que cumple lo que el cliente quiere.
- *Evolución del software*, el software tiene que evolucionar para satisfacer las necesidades cambiantes del sistema.

Además, también existen actividades de soporte al proceso, como la *documentación y el manejo de la configuración del software*.

Asimismo, (Sommerville, 2011), indica que los procesos de software son complejos y, como todos los procesos intelectuales y creativos, se apoyan en personas con capacidad de juzgar y tomar decisiones. No hay un proceso ideal; además, la mayoría de las organizaciones han diseñado sus propios procesos de desarrollo de software.

2.2.4. FASES DE DESARROLLO DE SOFTWARE

Existen modelos para representar un proceso de Software, estos modelos son abstracciones del proceso que se utilizan para explicar los diferentes enfoques del desarrollo de software. Según (Sommerville, 2011) los modelos genéricos de proceso de Software son:

- *El modelo en cascada*, Éste toma las actividades fundamentales del proceso de especificación, desarrollo, validación y evolución y, luego, los representa como fases separadas del proceso, tal como especificación de requerimientos, diseño de software, implementación, pruebas, etcétera.
- *Desarrollo Incremental*, Este enfoque vincula las actividades de especificación, desarrollo y validación. El sistema se desarrolla como una serie de versiones (incrementos), y cada versión añade funcionalidad a la versión anterior.
- *Ingeniería de software orientada a la reutilización*, Este enfoque se basa en la existencia de un número significativo de componentes reutilizables. El proceso de desarrollo del sistema se enfoca en la integración de estos componentes en un sistema, en vez de desarrollarlo desde cero.

Basándonos en la descripción del modelo en cascada o también llamado ciclo de vida del software, que consiste en el paso de una fase en cascada a otra, y que en cada fase se produce documentación; mostraremos las fases del desarrollo de software basado en este modelo como se aprecia en los conceptos de la página web (Área de Sistemas de la Vicerrectoría de la Docencia de la Benemérita Universidad Autónoma de Puebla, 2008) donde textualmente se cita:

Análisis de requisitos. - extraer los requisitos de un producto de software es la primera etapa para crearlo. Mientras que los clientes piensan que ellos saben lo que el software tiene que hacer, se requiere de habilidad y experiencia en la ingeniería de software para reconocer requisitos incompletos, ambiguos o contradictorios. El resultado del análisis de requisitos con el cliente se plasma en el documento ERS, Especificación de Requerimientos del Sistema, cuya estructura puede venir definida por varios estándares, tales como CMM-I. Asimismo, se define un diagrama de Entidad/Relación, en el que se plasman las principales entidades que participarán en el desarrollo del software. La captura, análisis y especificación de requisitos, es una parte crucial; de esta etapa depende en gran medida el logro de los objetivos finales.

Diseño y arquitectura. - se refiere a determinar cómo funcionará de forma general sin entrar en detalles. Consiste en incorporar consideraciones de la implementación tecnológica, como el hardware, la red, etc. Se definen los

Casos de Uso para cubrir las funciones que realizará el sistema, y se transforman las entidades definidas en el análisis de requisitos en clases de diseño, obteniendo un modelo cercano a la programación orientada a objetos.

Programación. - reducir un diseño a código puede ser la parte más obvia del trabajo de ingeniería de software, pero no es necesariamente la porción más larga. La complejidad y la duración de esta etapa está íntimamente ligada al o a los lenguajes de programación utilizados.

Pruebas. - Consiste en comprobar que el software realice correctamente las tareas indicadas en la especificación. Una técnica de prueba es probar por separado cada módulo del software, y luego probarlo de forma integral, para así llegar al objetivo. Se considera una buena práctica el que las pruebas sean efectuadas por alguien distinto al desarrollador que la programó, idealmente un área de pruebas; sin perjuicio de lo anterior el programador debe hacer sus propias pruebas. En general hay dos grandes formas de organizar un área de pruebas, la primera es que esté compuesta por personal inexperto y que desconozca el tema de pruebas, de esta forma se evalúa que la documentación entregada sea de calidad, que los procesos descritos son tan claros que cualquiera puede entenderlos y el software hace las cosas tal y como están descritas. El segundo enfoque es tener un área de pruebas conformada por programadores con experiencia, personas que saben sin mayores indicaciones

en qué condiciones puede fallar una aplicación y que pueden poner atención en detalles que personal inexperto no consideraría.

Documentación. - Todo lo concerniente a la documentación del propio desarrollo del software y de la gestión del proyecto, pasando por modelaciones (UML), diagramas, pruebas, manuales de usuario, manuales técnicos, etc.; todo con el propósito de eventuales correcciones, usabilidad, mantenimiento futuro y ampliaciones al sistema.

Mantenimiento. - Mantener y mejorar el software para enfrentar errores descubiertos y nuevos requisitos. Esto puede llevar más tiempo incluso que el desarrollo inicial del software. Alrededor de 2/3 de toda la ingeniería de software tiene que ver con dar mantenimiento. Una pequeña parte de este trabajo consiste en arreglar errores, o bugs. La mayor parte consiste en extender el sistema para hacer nuevas cosas.

Todas estas fases, concatenadas adecuadamente, se muestran en la figura 1 fases del proceso de desarrollo de software.

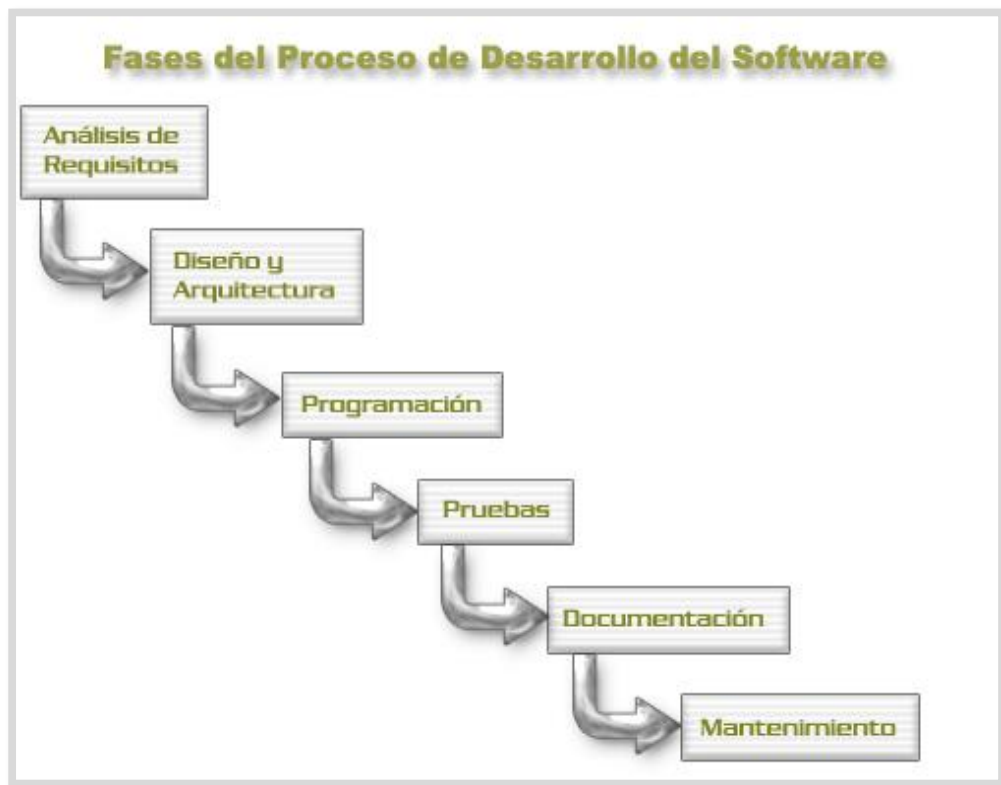


Figura 1: Fases del desarrollo de proceso de software

Fuente: (Área de Sistemas de la Vicerrectoría de la Docencia de la Benemérita Universidad Autónoma de Puebla, 2008)

2.2.5. HERRAMIENTAS DE INGENIERIA DE SOFTWARE

Según (Sommerville, 2011) menciona que, los procesos de software real son secuencias entrelazadas de actividades técnicas, colaborativas y administrativas con la meta general de especificar, diseñar, implementar y probar un sistema de software. Los desarrolladores de software usan en su trabajo diferentes herramientas de software. Las herramientas son útiles particularmente para dar apoyo a la edición de distintos tipos de documento y para manejar el inmenso volumen de información detallada que se reproduce en un gran proyecto de software.

Entonces las herramientas de desarrollo de software son programas usados para apoyar las actividades del proceso de la ingeniería de software; tal como lo menciona (Sommerville, 2011) que Las herramientas de software ofrecen apoyo de proceso al automatizar algunas actividades del proceso y brindar información sobre el software que se desarrolla. Los ejemplos de actividades susceptibles de automatizarse son:

- Desarrollo de modelos de sistemas gráficos, como parte de la especificación de requerimientos o del diseño del software.
- Generación de código a partir de dichos modelos de sistemas gráficos.
- Producción de interfaces de usuario a partir de una descripción de interfaz gráfica, creada por el usuario de manera interactiva.
- Depuración del programa mediante el suministro de información sobre un programa que se ejecuta.
- Traducción automatizada de programas escritos, usando una versión anterior de un lenguaje de programación para tener una versión más reciente.

2.2.6. CUADRANTE DE GARTNER

El Cuadrante de Mágico de Gartner es una herramienta de análisis, que según (Gonzalez, 2011), es una representación gráfica de un determinado mercado en un periodo de tiempo específico, en el cual se analizan y miden diversos proveedores, basándose en criterios definidos por la consultora Gartner, líder mundial en investigación y consultoría en Tecnologías de la

Información. Las posiciones reflejadas por los mismos en dicho cuadrante son el resultado de la evaluación en cuanto a su capacidad de ejecución y su visión integral en dicho periodo.

Los cuadrantes, según (Cardona, 2011), proporcionan una visión gráfica de cuatro tipos de posicionamiento competitivo para los proveedores de tecnología, donde el crecimiento del mercado es alto y la diferenciación de proveedores es clara, de esta forma aparecen los cuatro tipos de posicionamiento siguiente:

- **Líderes:** Ejecutan bien de acuerdo con la visión actual y están bien posicionados para el futuro.

Aquellos que presentan ofertas maduras que alcanzan las expectativas del mercado, de igual forma demuestran la visión necesaria para sostener su posición en el mercado, en la medida en que los requerimientos evolucionan. Estos actores tienen una gran base de clientes satisfechos (en relación con el tamaño del mercado) y disfrutan de una visibilidad alta dentro del mercado.

- **Visionarios:** Entienden hacia donde se dirige el mercado o tienen la visión de un mercado cambiante, pero no lo hacen bien todavía.
- **Especialistas o jugadores de nicho:** Se enfocan de manera exitosa en un pequeño segmento, o están desenfocados y no innovan ni superan a otros.
- **Retadores:** Lo hacen bien hoy o dominan un segmento amplio, pero no demuestran un entendimiento de la dirección del mercado”

Algo importante que mencionar es que el uso de los cuadrantes, según (Cardona, 2011), se orienta a brindar al cliente un entendimiento de los proveedores de tecnología que se están considerando para una oportunidad de inversión específica, teniendo en cuenta cada uno de los cuadrantes tiene su propósito y que no necesariamente se deba orientar a aquellos que se ubican en el cuadrante de líderes, toda vez que quienes están ubicados en el cuadrante de retadores, especialistas o visionarios, pueden cumplir con las expectativas de cada cliente en particular, dependiendo de qué tan alineado está el proveedor con los objetivos de negocio.

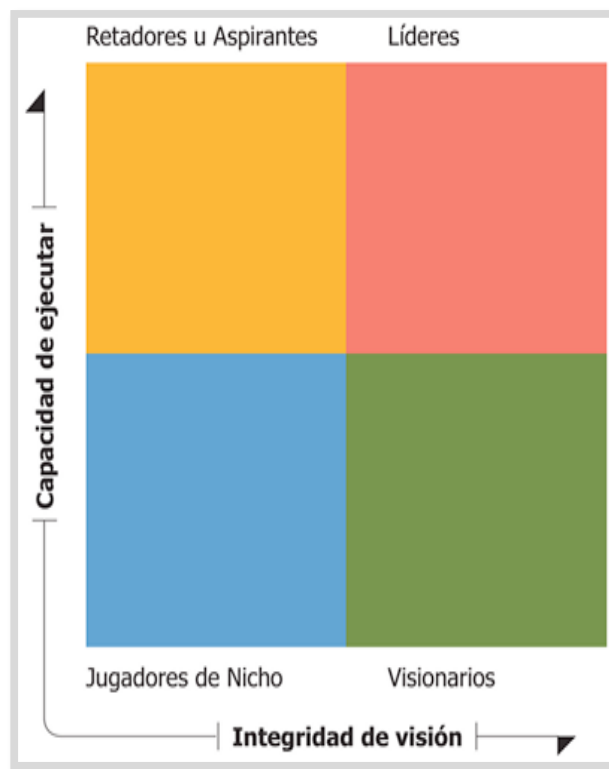


Figura 2: Cuadrante de Gartner
Fuente Revista Help Desk TIC (Costa, 2015)

2.3. DEFINICIONES OPERACIONALES

2.3.1. UTILIZACION

Son las tendencias de uso de las herramientas de ingeniería de software obtenidas de la aplicación web Google Trends, también la utilización hace referencia a la cantidad de documentación disponible en internet de estas herramientas, utilizando en el motor de búsqueda web Google.

2.3.2. COSTOS

Son los costos de adquisición (licencias) y mantenimiento de las herramientas de ingeniería de software, obtenidas de las páginas web oficiales de los propietarios de estas herramientas.

2.3.3. PROCESO DE GESTION

Son los procesos para apoyar a la gestión del desarrollo de software contemplados en la ISO/IEC 29110 que se mencionan a continuación: planeación del proyecto, ejecución del plan del proyecto, evaluación del plan del proyecto, evaluación y control del proyecto, cierre del proyecto.

2.3.4. PROCESO DE INGENIERIA

Son los procesos para apoyar a la Ingeniería de software contemplados en la ISO/IEC 29110 que se mencionan a continuación: inicio de implementación del software, análisis de requisitos del software, arquitectura y

diseño detallado de software, construcción de software, integración y pruebas de software, entrega de productos.

III. OBJETIVOS E HIPÓTESIS

3.1. OBJETIVOS

- **General**

Determinar una combinación de herramientas de ingeniería de software, producidas por las empresas líderes del mercado, que cubran la implementación de los procesos de la ISO/IEC 29110, atendiendo a criterios de costo y utilización para mejorar la eficiencia en los procesos de desarrollo de software de las PYMES. Caso de estudio en Lead Working Partner SAC.

3.2. HIPÓTESIS

- **Hipótesis General**

La implementación de las herramientas de ingeniería de Software, a seleccionar permitirá cumplir los procesos contemplados en la ISO/IEC 29110, atendiendo los criterios de costo y utilización para el desarrollo de software en una PYME.

- **Variables e Indicadores**

$$y = f(x)$$

Donde:

y = Variable dependiente.

x = Variable independiente.

y = Procesos de la ISO/IEC 29110.

x = Herramientas de Ingeniería de Software.

De allí las siguientes dimensiones para X son:

X1 = Costo

X2 = Utilización

Dimensiones para Y son:

Y1 = Procesos de Gestión

Y2= Procesos de Ingeniería

Tabla 1: Operación de variables

Variable	Definición Conceptual	Definición Operacional
Variable Independiente (x): Herramientas de Ingeniería de Software.	Son programas usados para apoyar las actividades del proceso de la ingeniería de software.	(X1) Utilización: - Tendencias de uso de la herramienta. - Cantidad de documentación disponible en internet. (X2) Costo: - Costo de Adquisición (licencia). - Costo de Mantenimiento (licencia).
Variable Dependiente(y): Procesos de la ISO/IEC 29110	Son los procesos (documentación) contemplados en la ISO/IEC 29110 para el desarrollo de software en las organizaciones pequeñas (VSE), específicamente el perfil de proceso Básico.	(Y1) Procesos de Gestión: - Número de procesos involucrados. - Nivel de satisfacción del responsable del proceso. (Y2) Procesos de Ingeniería: - Número de procesos involucrados. - Nivel de satisfacción del responsable del proceso.

Fuente: Elaboración propia.

IV. METODOLOGÍA DE LA INVESTIGACIÓN

4.1. MÉTODO

El tipo de investigación en la que se enmarca la presente tesis, es la Investigación Tecnológica. Según (Espinoza Montes, 2014) la *“investigación Aplicada, también conocida como diseño o innovación tiene como propósito aplicar resultados de la investigación experimental para diseñar tecnologías de aplicación inmediata en la solución de los problemas de la sociedad, buscando eficiencia y productividad”* (p.91).

Por lo tanto, el nivel de investigación es de tipo Aplicada y el diseño es descriptivo simple, ya que el objeto de la investigación fue identificar o determinar las Herramientas de ingeniería de software apropiadas para la implementación la ISO/IEC 29110 en las PYMES de la Industria del Software, basado en un caso de estudio de una empresa, teniendo en cuenta aspectos de costo y utilización de las herramientas.

4.2. TÉCNICA

Esta investigación se establece por medio de diferentes técnicas para la investigación.

- **Análisis y revisión bibliográfica.** - Para recopilar información de libros, textos y documentos relacionados y más apropiados con la problemática de investigación.

- **Encuesta.** - Realizados a los involucrados en el desarrollo del proyecto RESTORi de la empresa Lead Working Partner, los cuales se encuentran en el anexo 13.3 al 13.8.

4.3. UNIVERSO Y MUESTRA

- **Población y Muestra**

Para la presente tesis, no se aplica muestreo porque es un caso de estudio de una sola empresa, considerando a la empresa Lead Working Partner SAC como población total.

V. ANÁLISIS SITUACIONAL DE LA INDUSTRIA DEL SOFTWARE

5.1. LA INDUSTRIA DEL SOFTWARE EN EL MUNDO

El desarrollo de software a nivel mundial es cada vez más grande, India es uno de los lugares más reconocidos actualmente en este tema, sin embargo, las adquisiciones de software y sus estadísticas son altamente favorables en el presente y en el futuro. Bangalore, la tercera ciudad con más habitantes en la India, es llamada “Silicón Valley” de ese país. Es reconocida como una de las ciudades símbolo en el desarrollo de la industria del software a nivel mundial. (Edery, 2008)

También (Edery, 2008) menciona que, los ingresos generados por esta industria del Software se estimaron en US\$ 52 mil millones para el 2008. De las cuales 77.5% constituyen exportaciones a USA y el Reino Unido. El outsourcing a empresas de diferentes sectores económicos como comunicaciones (Verizon), transporte aéreo (Delta Airlines) o el financiero (HSBC), constituye un componente principal de la industria del software. Actualmente, es el rubro de más rápido crecimiento, habiéndose triplicado desde el 2004. Los ingresos generados por outsourcing y el monto exportado equivalen

al 24% y al 27% del total de los ingresos y de las exportaciones de la industria del software.

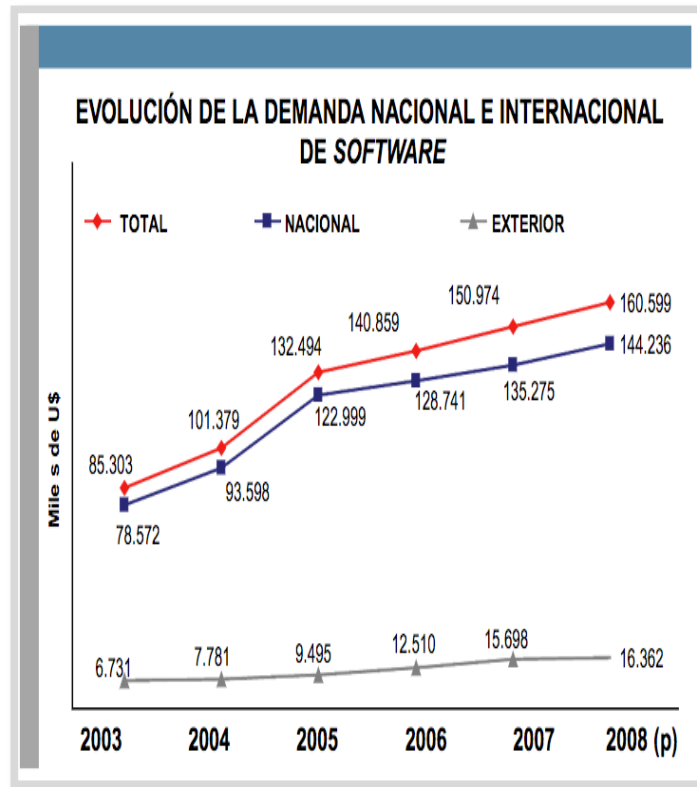


Figura 3: Demanda de software
Fuente: (Edery, 2008)

5.2. LA INDUSTRIA DEL SOFTWARE EN AMÉRICA LATINA

Ya hace muchos años atrás la industria viene creciendo, el mismo comportamiento se ha producido específicamente en Latino América. Según (Statista, 2013), los pronósticos para los próximos años de consumo de software en Latino América son prometedores, al 2019 las compras en software serán de US\$ 30 billones, y el Perú debería ser uno de los países que atrae esos ingresos.

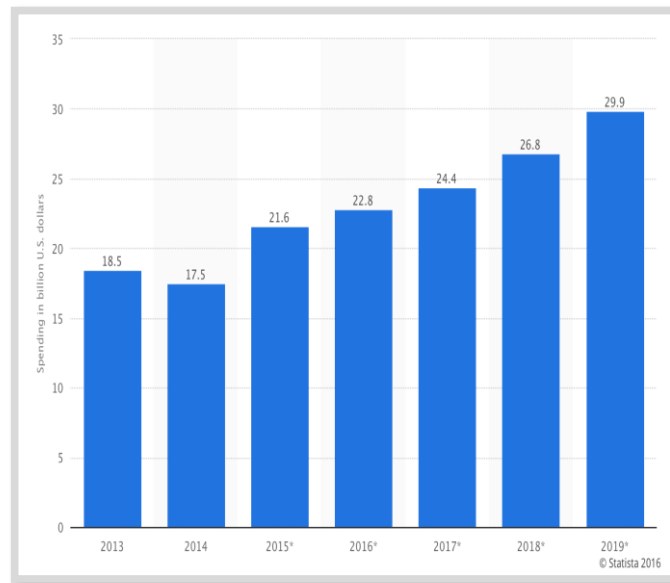


Figura 4: Pronósticos de adquisición de software en Latino América
 Fuente: (Statista, 2013)

Brasil en el 2010 reportó una buena distribución de empresas de software según su número de trabajadores, representando un gran crecimiento de las empresas dedicadas a Software y servicios de TI. (Ramos & Mendoza, 2014)

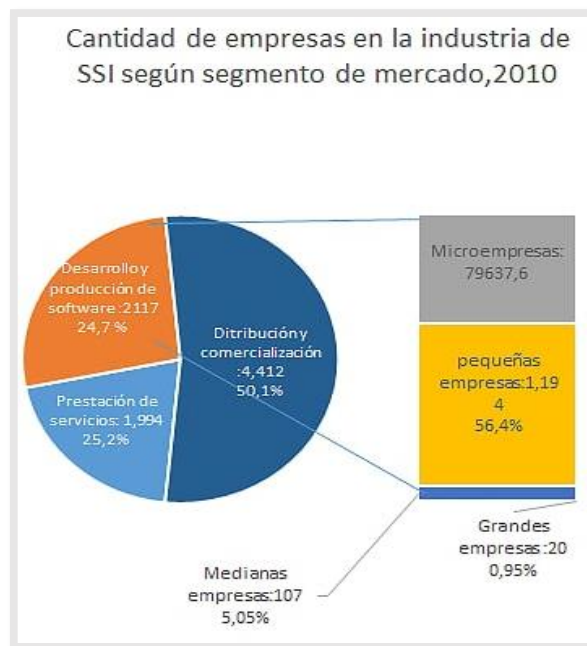


Figura 5: Empresas de TI en Brasil
 Fuente: (Ramos & Mendoza, 2014)

Así mismo Argentina en el 2013, realizó una encuesta para ver la situación de ese momento sobre el sector software con una muestra de 113 empresas. (Ramos & Mendoza, 2014)

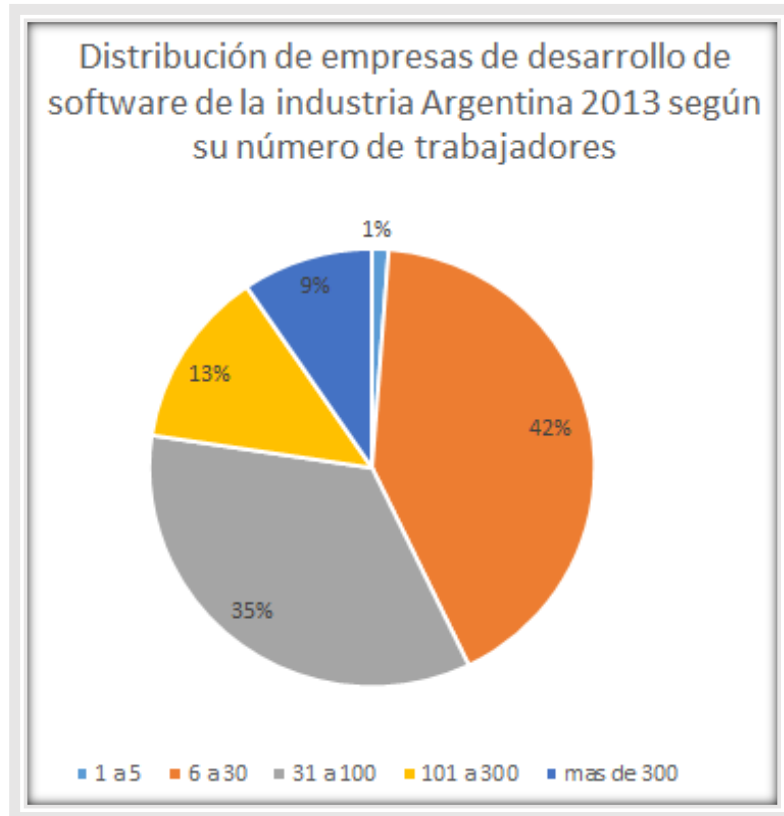


Figura 6: Empresas de Software en Argentina
Fuente: (Ramos & Mendoza, 2014)

5.3. LA INDUSTRIA DEL SOFTWARE EN EL PERÚ

Según (Edery, 2008), el Perú es un actor que tiene significativas ventajas para obtener parte del consumo internacional, ya que cuenta con un activo capital humano (30 000 programadores) y con la presencia de más de 300 empresas (90% pequeñas y microempresas), que en su mayoría no superan una década de funcionamiento.

El país ya ha venido exportando software al mundo, por tanto, se puede asumir que las puertas están abiertas para las demás empresas del sector, siendo que otras empresas han logrado dejar en alto el nombre del país en este sector.

En el Perú, la industria del software se ha desarrollado gracias a la apuesta que viene realizando el sector privado con visión de futuro y la capacidad de crear soluciones tecnológicas competitivas a nivel internacional. Lo que permite estar presente en 17 mercados de destino. (Edery, 2008)

PRINCIPALES DESTINOS DE VENTA DE PRODUCTOS-SERVICIOS AL EXTERIOR 2007			
PAÍSES DE DESTINO	TOTAL	PRODUCTO	SERVICIO
EE.UU	52,80%	41,00%	72,60%
Comunidad Andina	26,90%	32,80%	16,90%
Europa	13,60%	19,10%	4,30%
Mercosur	3,10%	3,60%	2,20%
Centroamérica	2,20%	2,35%	1,90%
Otros	1,50%	1,20%	2,10%

Figura 7: Destinos de Venta de Software 2007

Fuente: (Edery, 2008)

Las soluciones que se ofrecen son ERP y desarrollo de outsourcing, esta última aprovecha el huso horario del país respecto a los mercados de interés y los costos ofrecidos.

En (Edery, 2008) menciona, que desde el año 2000 PROMPERU (Comisión de Promoción del Perú para la Exportación y el Turismo) viene promoviendo este sector de la Industria del Software. Así se impulsó la creación de la APESOFT (Asociación Peruana de Software), como gremio que aglutina a la mayor parte de empresarios del rubro y a través del cual se ejecuta el programa CREA SOFTWARE PERU, que representa la marca de este sector (Industria del Software) y distingue a las empresas con capacidad de exportar sus soluciones (Software). Esta iniciativa se sustenta en la estrategia nacional de competitividad y es la base del Plan Operativo Exportador del Sector Servicios, que se viene ejecutando desde el MINCETUR (Ministerio de Comercio Exterior y Turismo), dentro del marco de Plan Estratégico Nacional Exportador - PENX. (Edery, 2008)

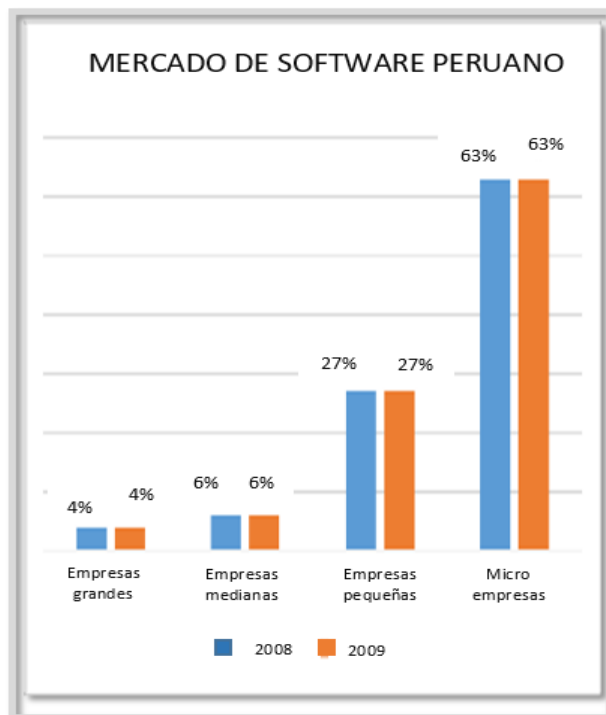


Figura 8: Mercado de Software Peruano al 2009

Fuente: (Ramos & Mendoza, 2014)

Según PROMPERU en el 2011 la tendencia del sector del software es un mercado relativamente joven, donde las microempresas y pequeñas empresas forman una parte significativa del mercado y que representa sus ventas en 85 millones de dólares creciendo a 167 millones.

Sin embargo, no solo se trata de tener la intención de querer hacerlo, para satisfacer esas necesidades las empresas peruanas deben asegurar que su trabajo sea de calidad, para el cual hay muchos caminos de lograrlo.

5.4. MODELOS DE MEJORA DE PROCESOS PARA PYMES

Según (Pesado, Bertone, Esponda, Pasini, & Martorelli, 2009) menciona que, la gestión formal de calidad es particularmente importante para equipos que desarrollan sistemas grandes y complejos. Por este motivo es que las principales normas reconocidas por la industria del software a nivel mundial, como CMM, CMMI, ISO (9000, 12207, 15504) tienen su centro en la gestión de la calidad de las grandes empresas. Pero, para el desarrollo orientado a pequeñas y medianas empresas (PyMEs) de software, la calidad no pasa a ser un tema menor. Si bien las normas más reconocidas internacionalmente son conceptualmente aplicables, resultan en prácticas muy costosas y difíciles, por la necesidad de contar con recursos humanos capacitados.

En algunos países iberoamericanos se está intentando abordar este problema con algunas iniciativas como el modelo MoProSoft de México, el proyecto MPS.BR de Brasil, el modelo SIMEP-SW de Colombia, ITMark, COMPETISOFT, entre otros, concluyendo en la norma ISO/IEC 29110 del año 2010. (Pesado, Bertone, Esponda, Pasini, & Martorelli, 2009)

MPS.BR.- El proyecto mps.BR está basado en ISO/IEC 12207:2002, CMMI e ISO/IEC 15504:2003, tiene como objetivo principal definir e implementar un modelo para la mejora de procesos de software. Pretende dar respuesta a la pregunta ¿Cómo mejorar radicalmente los procesos

de software en Brasil, de forma que estas obtengan un nivel de madurez 2 o 3 a un costo accesible? (Pinto, Garcia, Ruiz, & Piattini, 2016)

SIMEP-SW – Agile Spi.- El proyecto Sistema Integral para el Mejoramiento de los Procesos de Desarrollo de Software en Colombia, SIMEP-SW se formó con la idea de proporcionar a las empresas del sector informático de Colombia las herramientas necesarias para motivarlas a mejorar sus procesos de desarrollo de software con el objetivo de facilitar el posicionamiento y la competitividad en mercados nacionales e internacionales. El proyecto buscaba crear, aplicar y probar un sistema de mejoramiento que integre elementos de modelos de calidad, mejoramiento y evaluación reconocidos internacionalmente, adaptados a las características propias de la industria del software colombiana y que pueda ser replicado a industrias de características similares a nivel nacional e internacional.

IT - MARK.- modelo de calidad internacional diseñado para PYMES para la gestión general de la empresa, la seguridad de la información, los procesos de desarrollo de software y sistemas y núcleo del modelo, basado en CMMI.

Moprosoft.- El Modelo de Procesos de Software fue desarrollado a solicitud de la Secretaría de Economía para servir de base a la Norma Mexicana para la Industria de Desarrollo y Mantenimiento de Software bajo el

convenio con la Facultad de Ciencias, Universidad Nacional Autónoma de México. (Hanna, Claudia, & Angelica Su, 2003)

Competisoft.- “Mejora de procesos para fomentar la competitividad de la pequeña y mediana industria del software de Iberoamérica”, es un proyecto de investigación en el que participaron universidades, empresas, centros públicos y organismos de estandarización de trece países de la región, entre los que se encuentran: Argentina, Brasil, Chile, Colombia, Costa Rica, Cuba, Ecuador, España, México, Perú, Portugal, Uruguay y Venezuela.

ISO/IEC 29110.- Evidenciando la necesidad de establecer modelos de mejora para PyMEs, ISO constituyó el grupo de trabajo SC7-WG24, cuyo objetivo es lograr que sus estándares de procesos de software (o adaptaciones de estos) se puedan aplicar a pequeñas y medianas empresas desarrolladoras de software.

VI. SELECCIÓN DE HERRAMIENTAS

6.1. IDENTIFICACIÓN DE HERRAMIENTAS

Las herramientas que van a cubrir la mayoría de los procesos de la Ingeniería de software son las herramientas ALM (*Application Lifecycle Management*), para elegir las se debe seleccionar las empresas líderes que producen estas herramientas, y eso se conoce a partir del cuadrante de Gartner, descrita anteriormente.



Figura 9: Cuadrante de Gartner ALM

Fuente: (InfoQ, 2015)

Como se observa en la tabla 2 son cinco empresas líderes que producen herramientas ALM identificadas en el 2015 según el cuadrante de Gartner.

Tabla 2: Herramientas ALM

Empresa	Herramientas ALM
IBM	con la suite de herramientas Rational
Microsoft	con Team Foundation Service
Atlassian	con Jira
VersionOne	con VersionOne
Rally	Con Rally

Fuente: Elaboración propia

6.2. CARACTERÍSTICAS DE LAS HERRAMIENTAS

6.2.1. TEAM FOUNDATION SERVICE (MICROSOFT)

Team Foundation Service es un servicio de la empresa Microsoft, que proporciona herramientas de colaboración en el ciclo de Desarrollo de Software, donde se puede trabajar eficazmente entre los involucrados del proyecto.

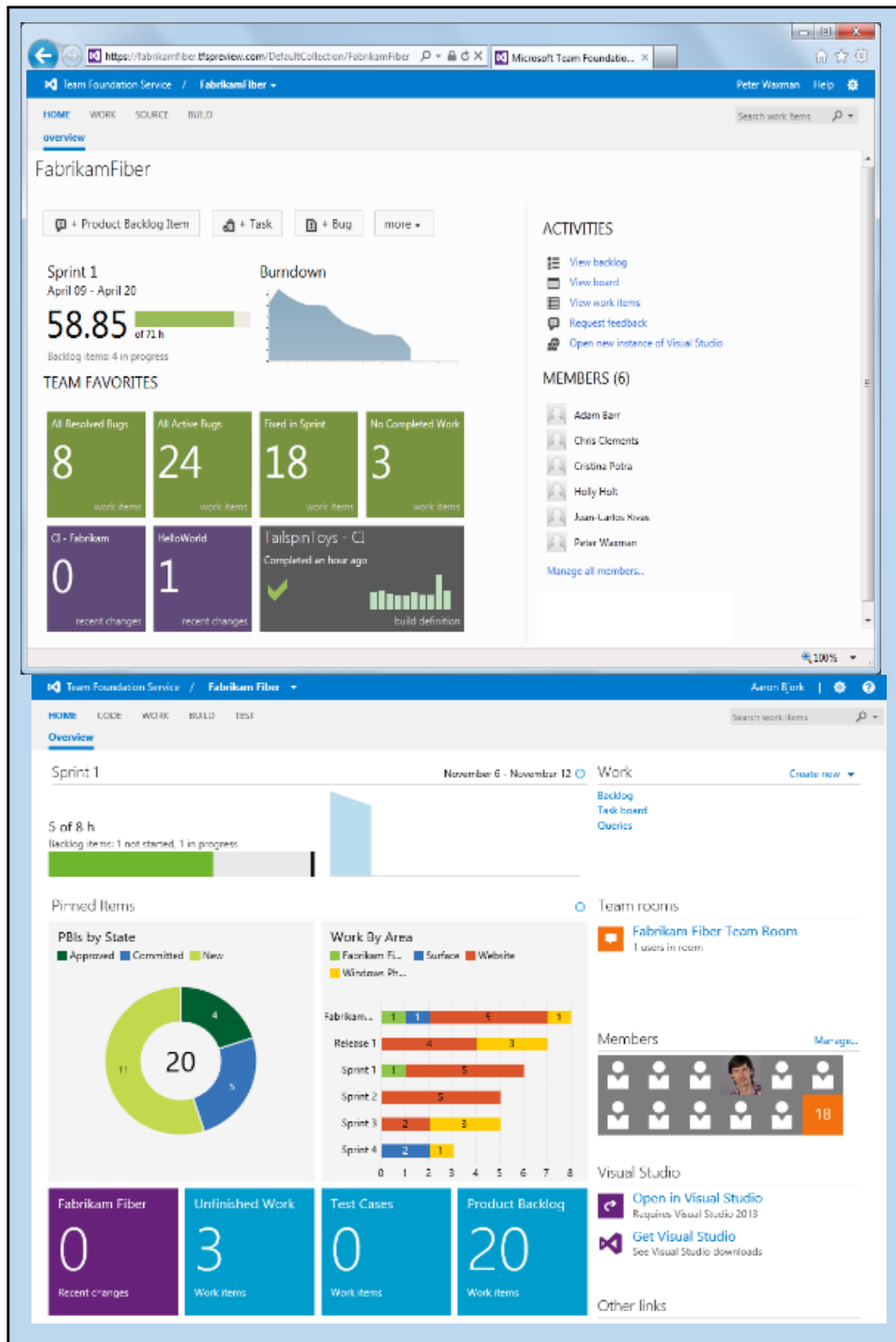


Figura 10: Screenshot de TFS
Fuente: (Microsoft. s.f.)

Tabla 3: Características *TEAM FOUNDATION SERVICE* (TFS)

CARACTERISTICA	DESCRIPCION
Control de versiones	Con repositorios de código privados ilimitados.
Herramientas de Agile	Priorización de trabajos pendientes de arrastrar y colocar. Paneles de tareas y Kanban Board. Planificación de sprint. Administración de errores. Gráficos y paneles.
Herramientas para equipos de Java	Codifique en Eclipse, IntelliJ o en su IDE de Java favorita. Repositorios de Git privados ilimitados gratuitos. Planeamiento ágil y seguimiento de los elementos de trabajo. Integración continua y compilaciones con Ant, Maven o Gradle. Integración con las herramientas de Java o servicios como Jenkins.
Integración continua	Cree compilaciones en Linux, Mac y Windows. Use los agentes de compilación multiplataforma de Android, iOS, Java, .NET y otras aplicaciones. Integre su trabajo, las pruebas que haga, el código, la compilación y la versión sin interrupciones. Realice un seguimiento de las compilaciones mediante el estado de compilación en tiempo real.
Pruebas.	Generación de plan de pruebas. Ejecución de pruebas manuales.

Fuente: Elaboración propia

- **Costos de licencia.-** Las licencias de TFS de acuerdo a si es libre o pagada cubren la gestión de testing. Sin embargo en la versión libre se pueden crear casos de prueba sin problema alguno.

Tabla 4: Licencias TFS

Nro. Usuarios	Costo total mensual
5	Libre
10	US\$ 30.00
20	US\$ 110.00
50	US\$ 350.00
100	US\$ 750.00
200	US\$ 1150.00
1000	US\$ 4350.00

Fuente: Elaboración propia

- **Tendencia de uso.-** Las tendencias de uso de TFS hasta el momento de la recopilación han sido bastante positivas.

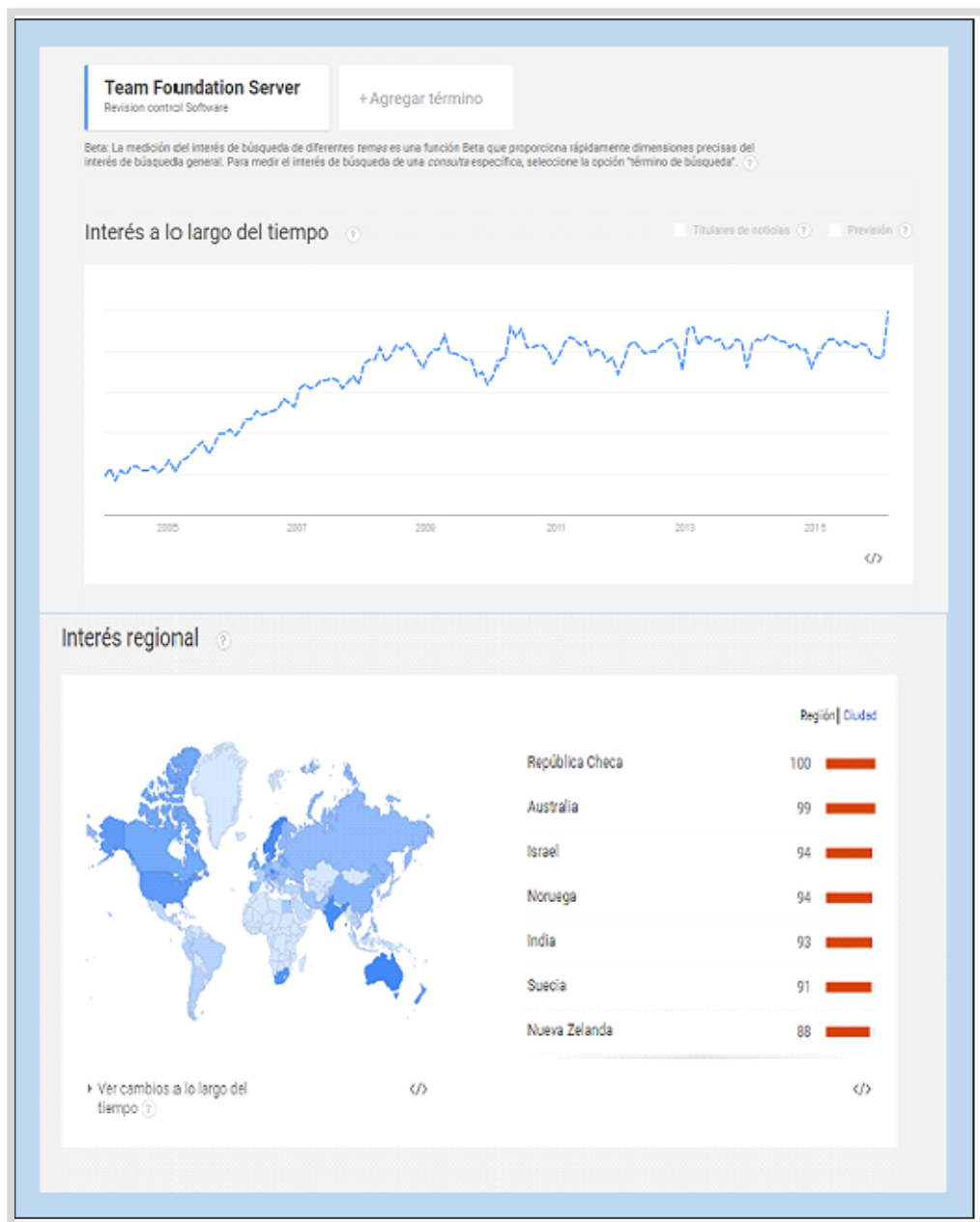


Figura 11: Utilización de TFS

Fuente: Obtenido de Google Trends

<https://trends.google.es/trends/explore?q=team%20foundation%20service>

• **Cantidad de documentación.-** Al buscador web más utilizado a nivel mundial es google.com por tanto al buscar documentación sobre el uso de TFS se encontraron 533,000 resultados.

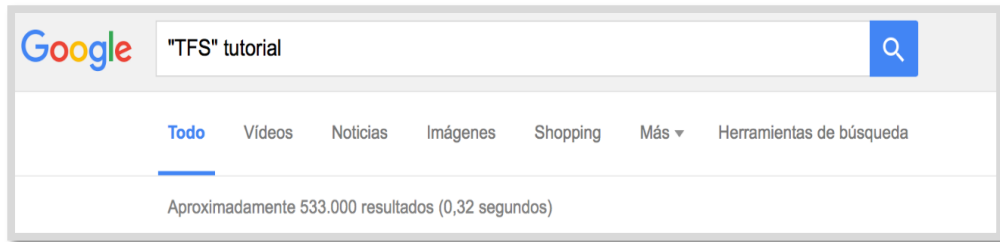


Figura 12: Resultados de búsquedas en Google
Fuente: Obtenido del Buscador Google

6.2.2. VERSIONONE

En esta herramienta las características dependen de la versión de licencias, en la tabla se observan las características según el tipo de licencia.

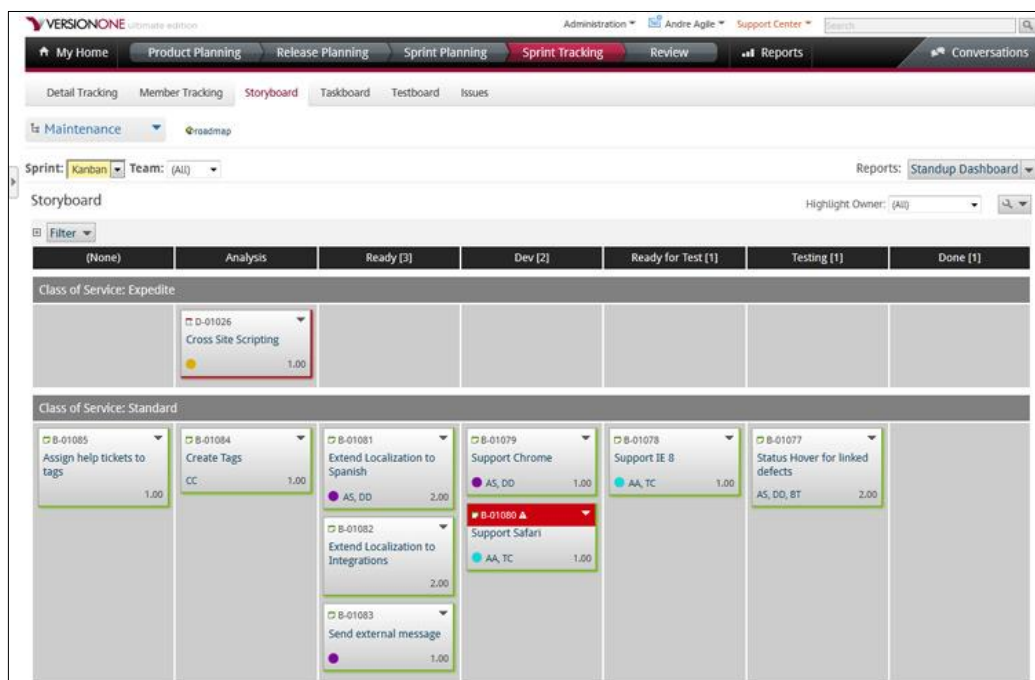


Figura 13: Vistas de Versión ONE
Fuente: (VersionOne, 2016)

Tabla 5: Características Versión ONE

Team	Catalyst	Enterprise	Ultimate
<ul style="list-style-type: none"> •Proyecto individual •Planificación de la liberación y la iteración •Seguimiento de historia •Gestión de Defectos •Storyboard, TaskBoard y Testboard •Seguimiento de Pruebas de Aceptación •Burndown y la velocidad de informes. 	<ul style="list-style-type: none"> •Múltiples Proyectos •Juntas Kanban Board •Entre proyectos Rollups •Seguimiento de incidencia. •Vistas y flujo de trabajo personalizados •Informes mejorados 	<ul style="list-style-type: none"> •Múltiples Equipos y Proyectos •Gestión de programas •Gestión de epic •Los espacios de trabajo personalizados •AdvancedReporting •Proyecto y seguridad basada en roles •TeamRooms personalizados 	<ul style="list-style-type: none"> •Gestión ágil de la cartera •Informes personalizados •Tableros de instrumentos ejecutivos •Las visualizaciones ágiles •Pruebas de regresión •Roadmapping producto •Ideación cliente

Fuente: Elaboración propia.

• **Costos de licencias.-** De acuerdo a las características del proyecto y de la necesidad de la organización se ofrece el costo de las licencias.

Tabla 6: Licencias Versión ONE

Team	Libre
Catalyst	US\$ 175.00 / 20 usuarios / mes
Enterprise	US\$ 29.00 / usuario / mes
Ultimate	US\$ 39.00 / usuario / mes

Fuente: Elaboración propia

• **Tendencia de uso.-** Esta herramienta está siendo muy utilizado en Filipinas, su crecimiento está bastante creciente en los últimos tiempos.

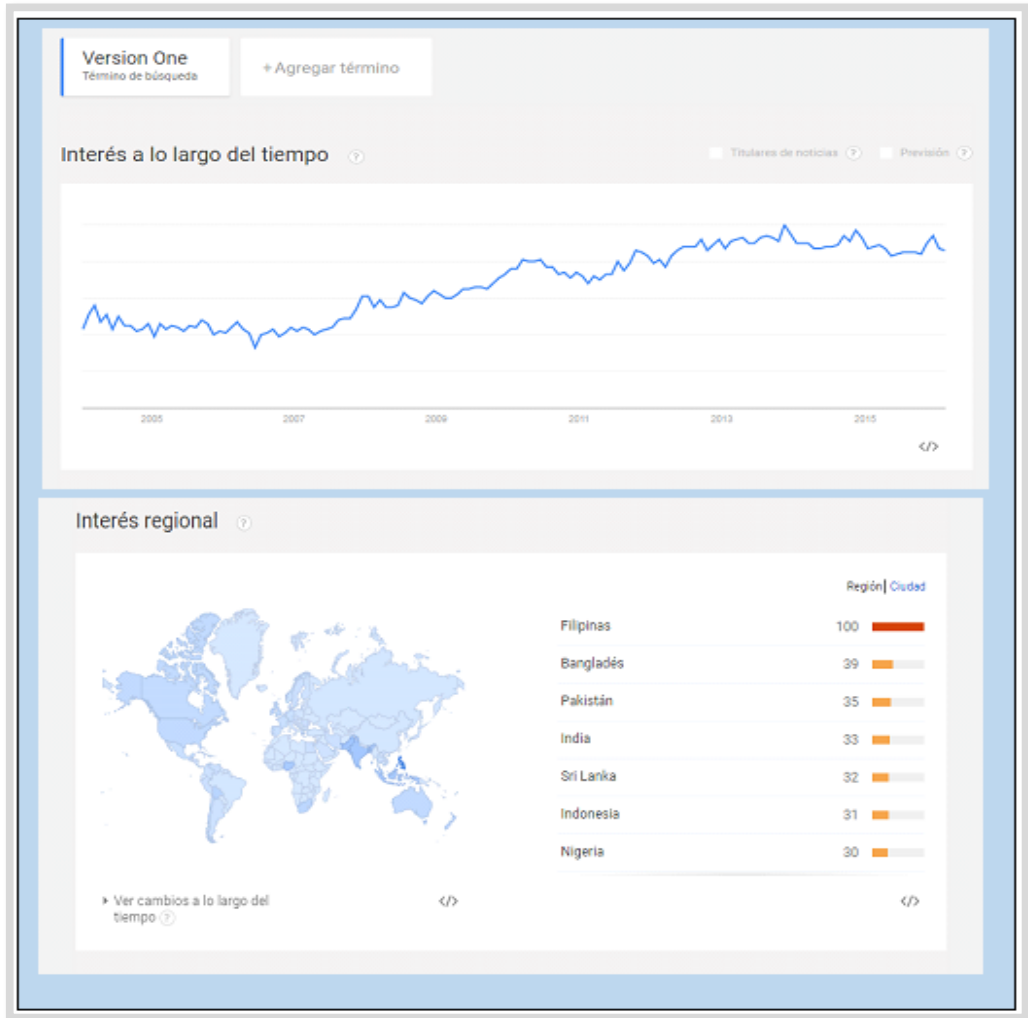


Figura 14: Utilización de VERSIONONE

Fuente: Obtenido de Google Trends
<https://trends.google.com/trends/explore?q=versionone>

• **Cantidad de documentación.-** Al buscar documentación sobre VersionOne ALM se encontraron 337 resultados.

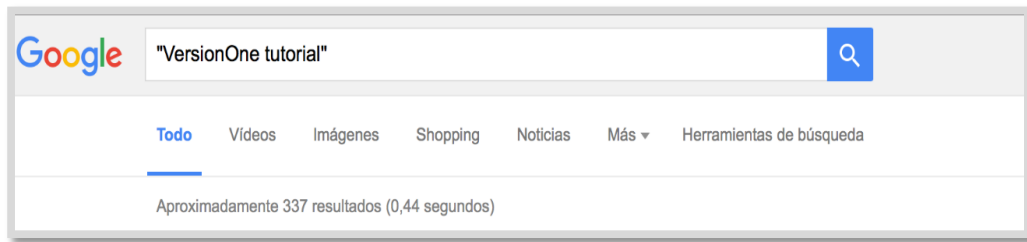


Figura 15: Búsquedas de Versión ONE
Fuente: Obtenido del Buscador Google

6.2.3. ATLISSIAN (JIRA)

JIRA es una de las herramientas más conocidas en los entornos de desarrollo de software libre, ofrece varias herramientas conocidas, pero el ALM si tiene costos en comparación de otras empresas.

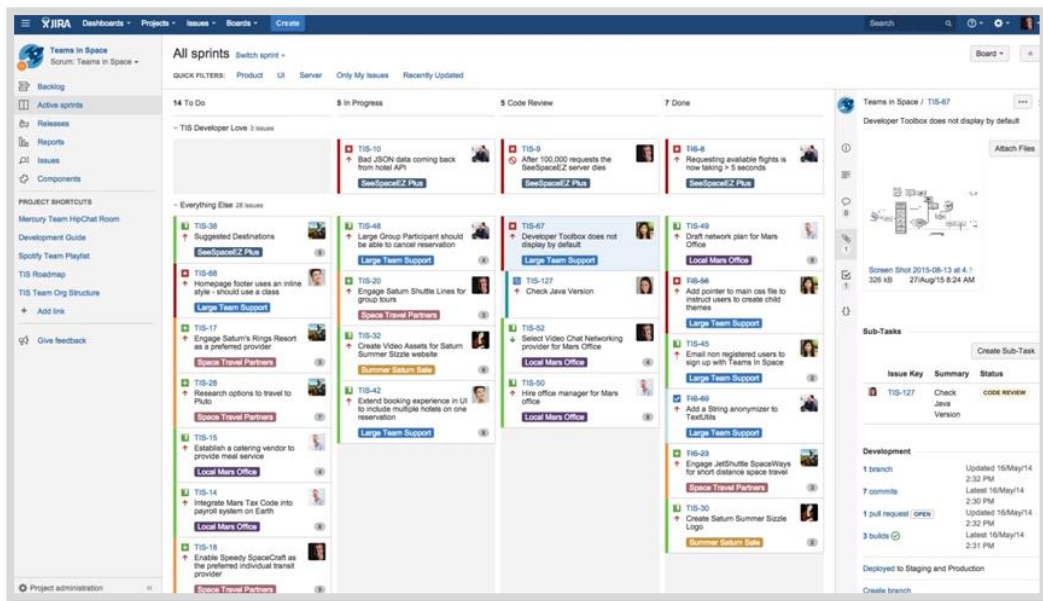


Figura 16: Vistas de JIRA
Fuente: (Atlassian. s.f.)

Tabla 7: Características JIRA

Característica	Descripción
Planificación	Crea historias de usuario e incidencias, planifica sprints y distribuye tareas entre tu equipo de software.
Supervisión	Prioriza y analiza el trabajo de tu equipo en su contexto y con una completa visibilidad.
Creación de Informes	Mejora el rendimiento del equipo con datos visuales en tiempo real de gran utilidad.
Integración con otras herramientas	<ul style="list-style-type: none"> ➤ Bitbucket: Versionamiento de código fuente. ➤ Bamboo: Integración Continua. ➤ HipChat: Colaboración en tiempo real.

Fuente: Elaboración propia

- **Costos de licencias.-** El costo de licencias de uso varía según el numero personas que integran los proyectos, mientras la empresa va creciendo va aumentando el costo.

Tabla 8: Licencias JIRA

Equipos pequeños	US\$ 10.00 / 10 usuarios / mes
Equipo en crecimiento	US\$ 75.00 / 15 usuarios / mes
Equipo en Crecimiento	US\$ 150.00 / 25 usuarios / mes
Equipo en Crecimiento	US\$ 300.00 / 50 usuarios / mes

Fuente: Elaboración propia

- **Tendencias de uso.-** El uso de Jira es bastante estable a lo largo de tiempo y usado en muchas partes del mundo, ya que viene ofreciendo sus herramientas desde hace mucho años atrás y ha sabido madurar sus productos.

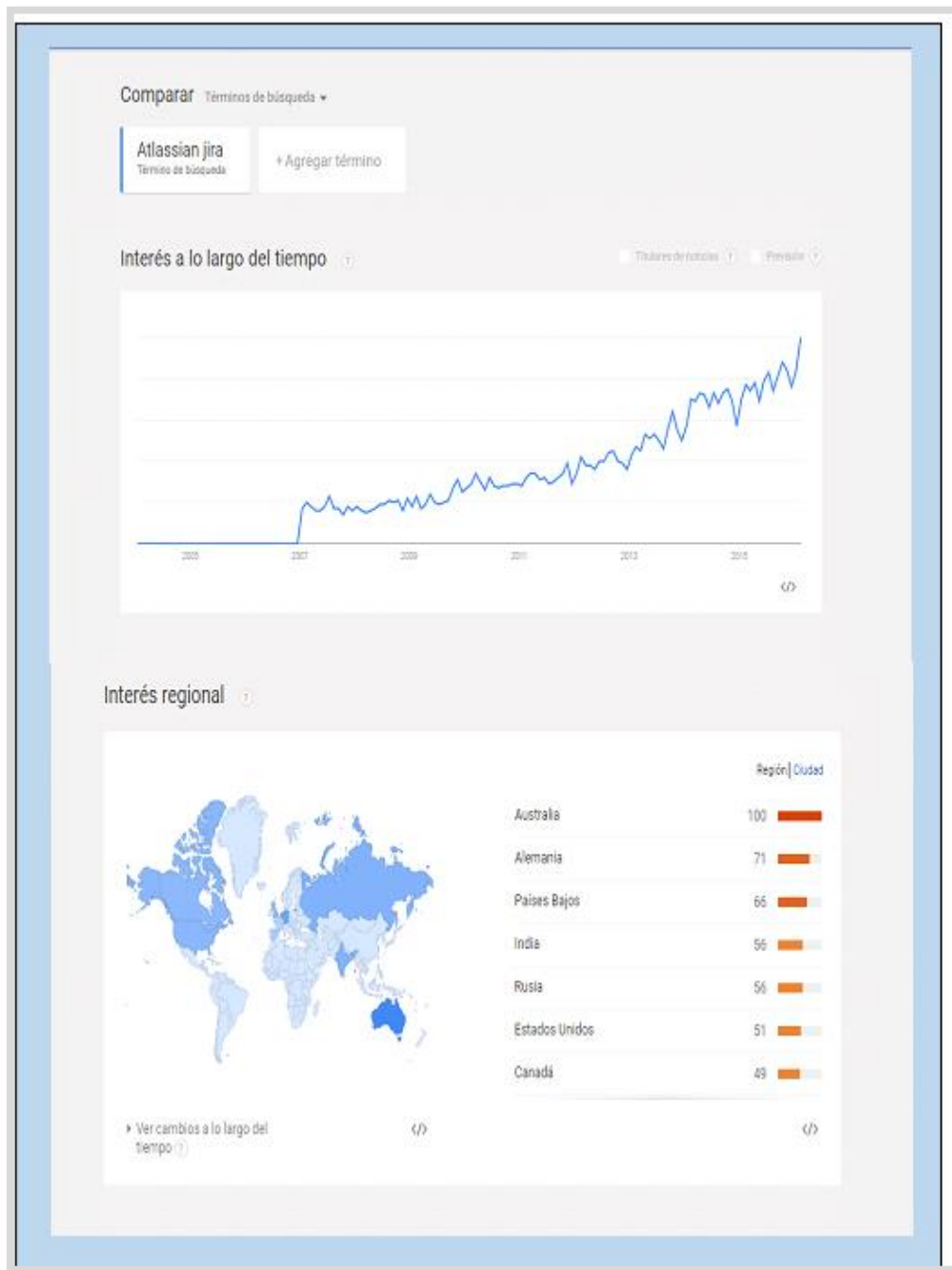


Figura 17: Tendencia de JIRA

Fuente: Obtenido de Google Trends

<https://trends.google.com/trends/explore?q=JIRA>

• **Cantidad de documentación.-** Al buscar documentación sobre el uso de Jira, se encontraron 15,100 resultados.

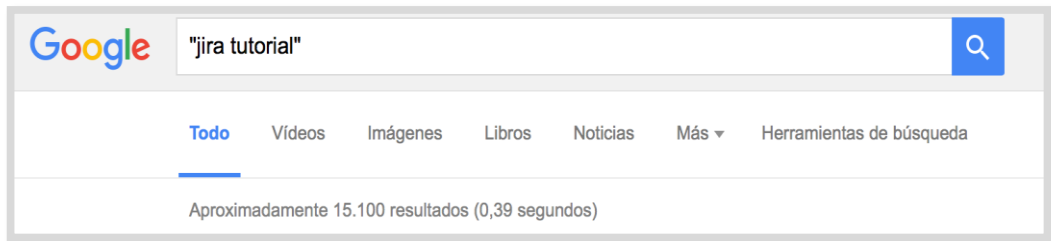


Figura 18: Búsquedas de JIRA
Fuente: Obtenido del Buscador Google

6.2.4. IBM (RATIONAL)

IBM ha propuesto toda una suite de herramientas para gestión el ciclo de desarrollo de software la que denomina *Rational Team Concert*.



Figura 19: Vistas de Team Concert
Fuente: (IBM, IBM Systems Magazine, 2016)

Tabla 9: Características Team Concert

Característica	Descripción
Planificación	La planificación en tiempo real diseñado para vincular los planes de trabajo real y mejorar el tiempo de entrega.
Trazabilidad	Trazabilidad del ciclo de vida para obtener información, tomar mejores decisiones y mejorar la gestión de entrega de software y previsibilidad.
Colaboración	En contexto de colaboración sobre los diseños, requisitos, planes, código y tareas a tomar mejores decisiones a partir de una única fuente de la verdad.
Indicadores	Desarrollo de inteligencia para establecer objetivos medibles y rastrear fácilmente.
Mejora continua.	La mejora continua para proporcionar una mejora gradual y la mejora avance a través de la automatización y la reutilización de las mejores prácticas.

Fuente: Elaboración propia

• **Costos de Licencias.-** Las licencias de IBM son proporcionadas por cada módulo que el cliente quiere usar, por tanto dependerá del proyecto a usar, los precios de la tabla son anuales.

Tabla 10: Pagos de Licencias Team Concert

IBM Team Concert Developer Authorized user single Install License + SW	US\$ 3726.00
IBM Team Concert Developer Authorized user single Install from Rational	US\$ 2613.00
IBM Team Concert Developer for WorkGroups Starter Pack 10 Authorized User	US\$ 5129.00

Fuente: IBM Company (IBM, IBM, s.f.)

• **Tendencias de uso.-** El Team Concert de IBM tiene poca tendencia de uso a nivel mundial no porque sea una mala herramienta, al contrario es una grandiosa herramienta, pero las pequeñas empresas en su mayoría no pueden acceder a los costos de la licencia aun.

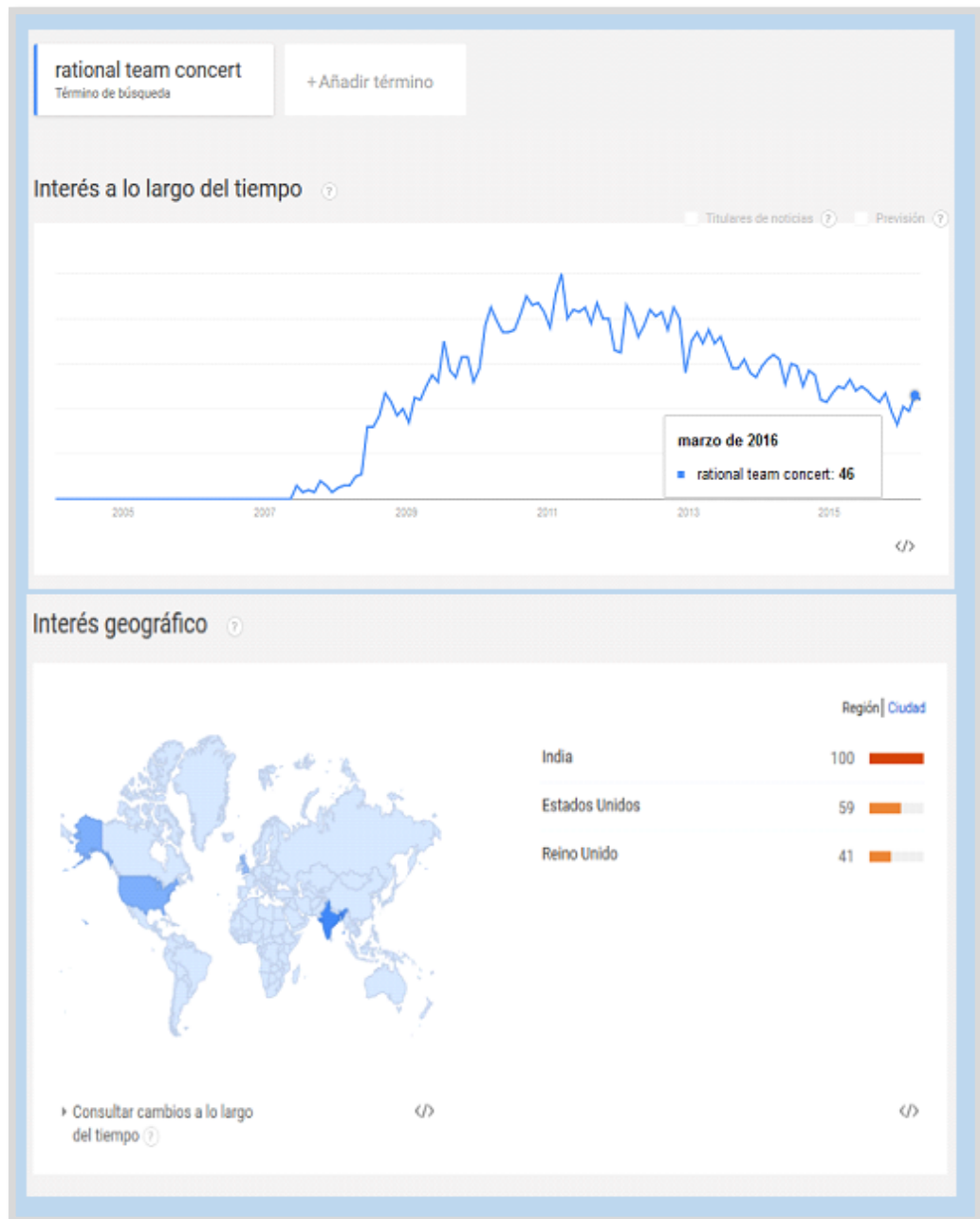


Figura 20: Tendencia de Team Concert

Fuente: Obtenido de Google Trends

<https://trends.google.com/trends/explore?q=Rational%20Team%20Concert>

• **Cantidad de documentación.-** Al buscar documentación sobre el uso de TeamConcert, se encontraron 1230 resultados.

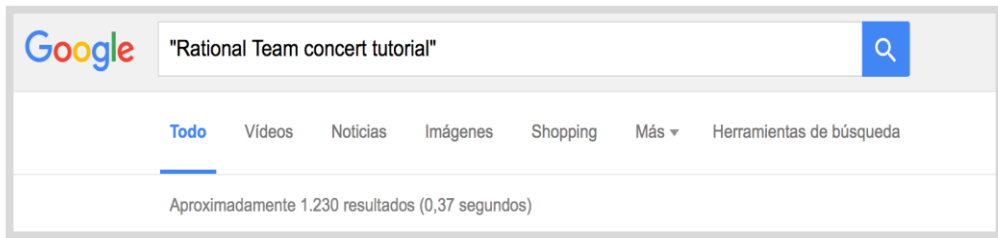


Figura 21: Búsquedas de Team Concert
Fuente: Obtenido del Buscador Google

6.2.5. RALLY

La Plataforma de Rally ágil para la Gestión del ciclo de vida de aplicaciones (ALM) recientemente agregada al Cuadrante de Gartner pero con mucho tiempo en el mercado del software.



Figura 22: Vista 1 de RALLY

Fuente: (Rally. s.f.)



Figura 24: Vista 2 de RALLY
Fuente: (Rally, s.f.)

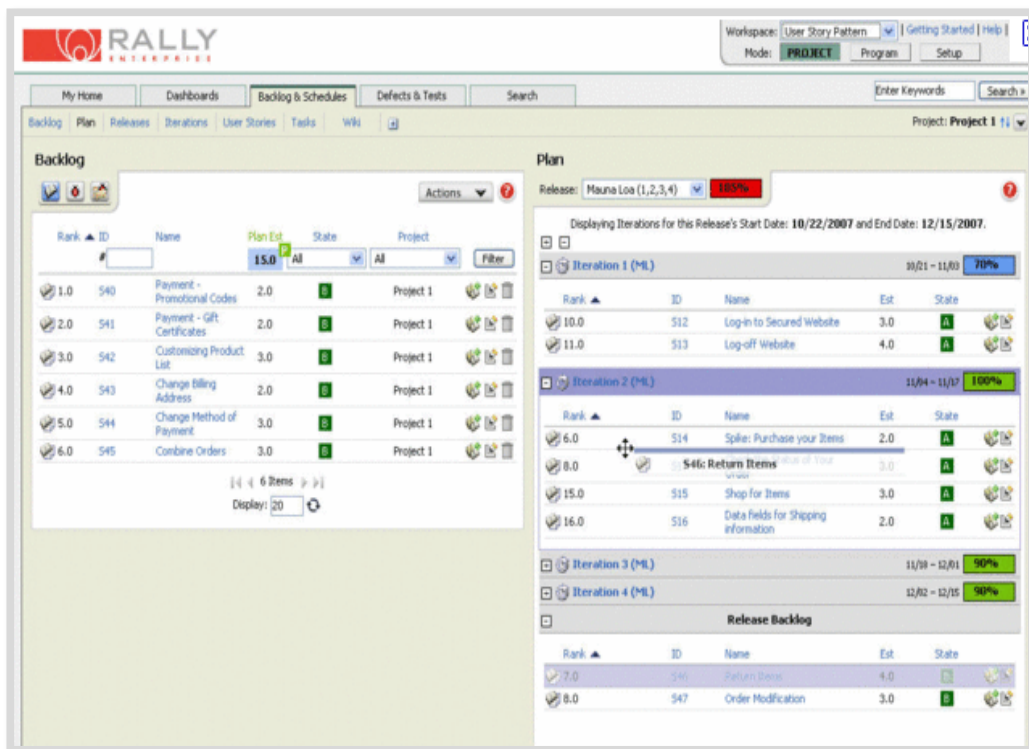


Figura 23: Vista 3 de RALLY
Fuente: Rally (Rally, s.f.)

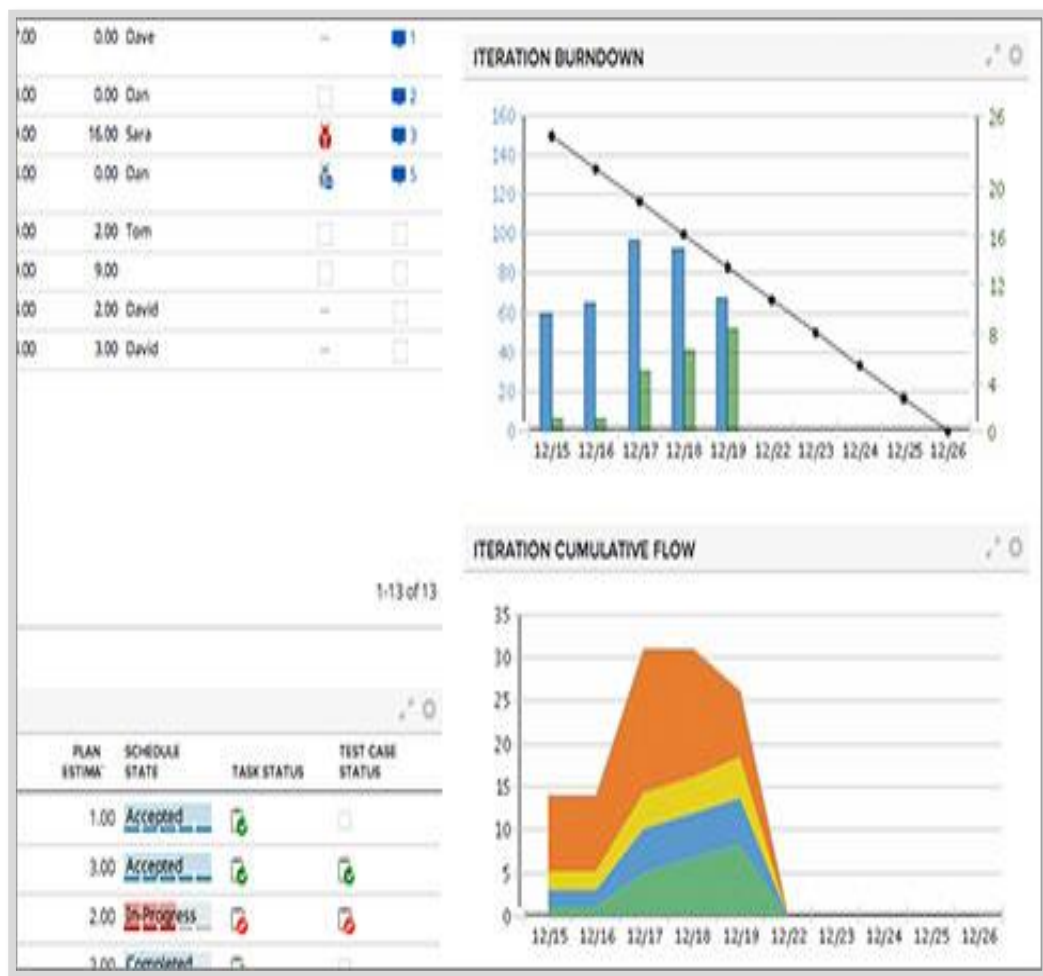


Figura 25: Vista 4 de RALLY
Fuente: Rally (Rally, s.f.)

Tabla 11: Características RALLY

Características	Descripción
Planeamiento	Seguimiento y gestión de su trabajo. Conectar con la estrategia de ejecución. Mejor toma de decisiones en un mundo de ritmo rápido. integrar y personalizar con una plataforma segura

Fuente: Elaboración propia

• **Costos de licencias.-** De acuerdo a las características del proyecto y de la necesidad de la organización se ofrece el costo de las licencias.

Tabla 12: Licencias RALLY

Community	Libre / 10 usuarios
Enterprise	US\$ 35 / usuario / mes
Unlimeted	US\$ 49.00 / usuario / mes

Fuente: Rally

• **Tendencias de uso.-** Rally es una herramienta que hace poco ha aparecido entre los líderes según el cuadrante de Gartner, está siendo muy utilizado en el habla inglesa sobre todo en USA.

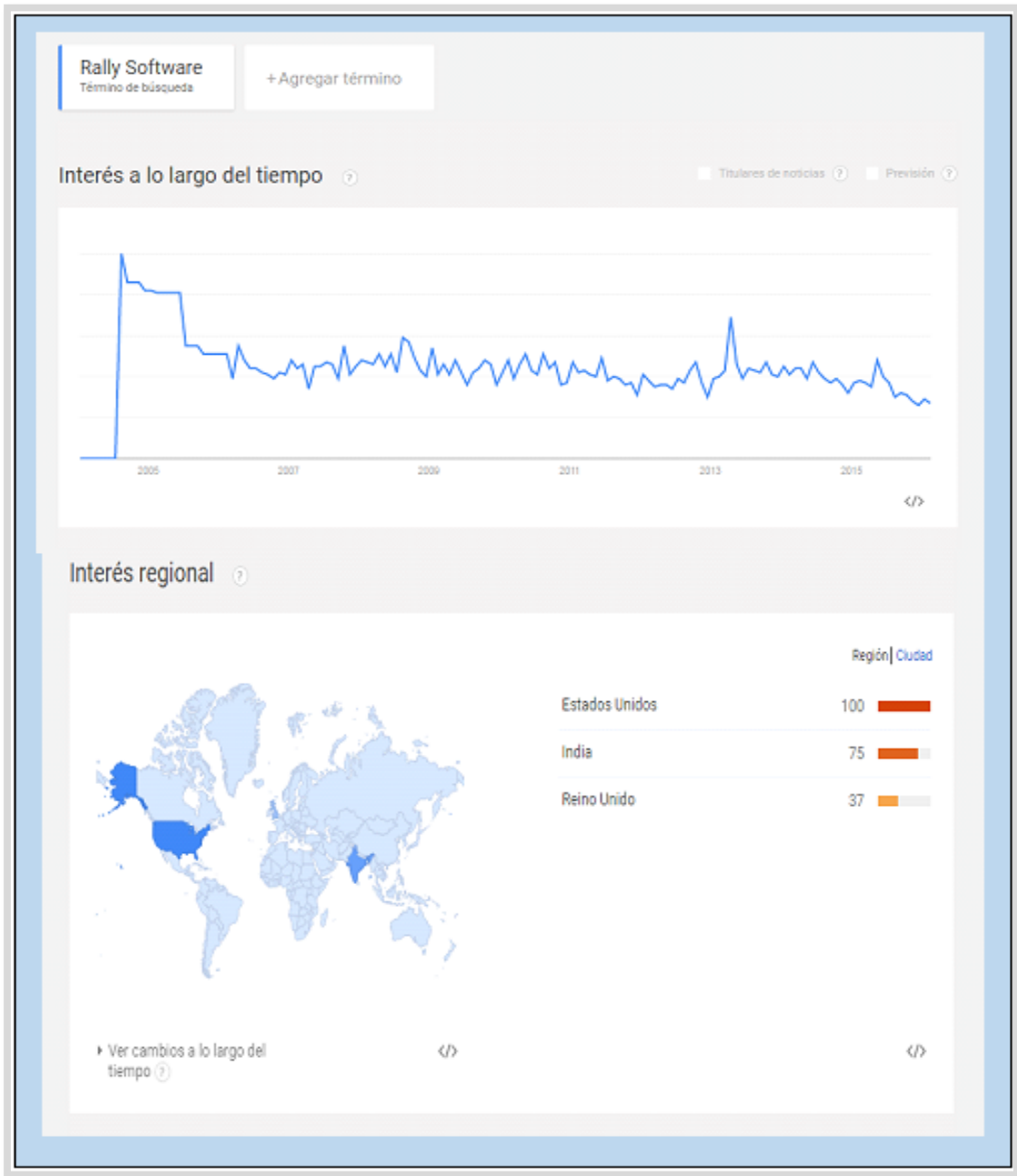


Figura 26: Tendencias de RALLY

Fuente: Obtenido de Google Trends

<https://trends.google.com/trends/explore?q=Rally%20Software>

• **Cantidad de documentación.-** Al buscar documentación sobre el uso de Rally Software, se encontraron 692 resultados.

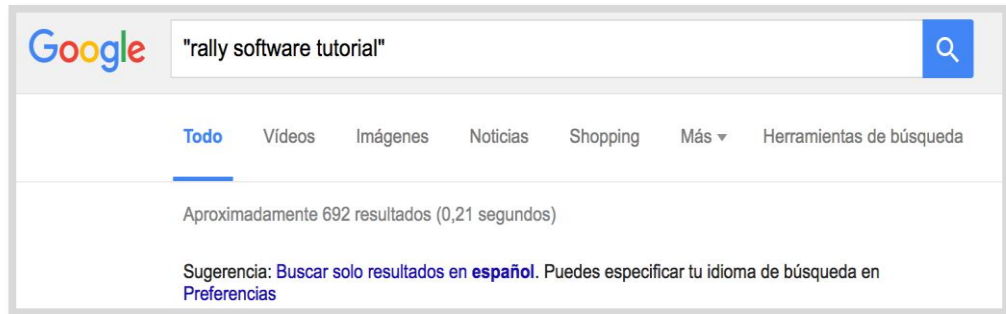


Figura 27: Búsquedas de RALLY
Fuente: Obtenido del Buscador Google

6.3. COMPARACIÓN DE HERRAMIENTAS

Para determinar la herramientas idónea se debe equilibrar de acuerdo a los criterios seleccionados en la investigación además sobre todo debe evaluarse en el corto plazo y a largo plazo.

Tabla 13: Comparación de herramientas

	Corto plazo	Usuarios a corto plazo	Largo plazo (50 Usuarios)	Proyectos	Tendencia	Docs.
IBM (Rational)	\$208 Mensual	Hasta 1 u.	\$10,400 Mensual	No definido	Media	1230
Microsoft (TFS)	\$150 Mensual	Hasta 10 u.	\$350 Mensual	Sin limite	Alta	533,000
Atlassian (JIRA)	\$100 mensual	Hasta 10 u.	\$300 Mensual	Sin limite	Alta	15,100
VersionOne	\$175 mensual	Hasta 20 u.	\$ 1,045 Mensual	Múltiples	Alta	337
Rally Software	Libre	Hasta 10 u.	\$1,750 Mensual	5	Media	692

Fuente: Elaboración propia.

Después de comparar las cuatro herramientas ALM teniendo en cuenta varios aspectos como el costo, opciones, cantidad de usuarios, cantidad de proyectos soportados y tendencias de uso y la documentación ofrecida en la web, las herramientas a continuar en comparación son TFS y JIRA.

La diferencia entre estas dos herramientas son muy pocas según los criterios de comparación utilizados. Sin embargo TFS se integra con las dos IDEs más utilizados en el desarrollo de software, entonces se decide usar TFS en la presente investigación para ampliar el alcance del uso de la investigación.

Tabla 14: Definición de herramientas

ALM	INTEGRACION CON IDEs	Observaciones
Team Foundation Service	<ul style="list-style-type: none"> • Visual Studio. • Eclipse. 	Integrados a través de plugins
Jira	<ul style="list-style-type: none"> • Eclipse. 	Integrado a través de plugins.

Fuente: Elaboración Propia

TFS en teoría permite ampliar la cobertura de las herramientas para cubrir los procesos de la ISO/IEC 29110. Sin embargo para corroborar ello, probamos estas herramientas en un caso de estudio.

VII. COBERTURA DE LAS HERRAMIENTAS SELECCIONADA EN LOS PROCESOS DE LA ISO/IEC 29110

7.1. MAPEO DE ARTEFACTOS

La ISO/IEC 29110 describe una serie de artefactos utilizados en sus procesos, algunos son entradas de un proceso y otros son resultados de la aplicación de un proceso, a su vez este artefacto resultado puede ser entrada de otro proceso y estos artefactos pueden irse modificándose durante la ejecución del proyecto.

Tabla 15: Mapeo de Artefactos

Artefacto	TFS	Herramientas de TFS
GESTIÓN DE PROYECTOS		
Plan de proyecto	NO	
Enunciado del trabajo	NO	
Resultado de verificación	SI	Product Backlog
Acta de reunión	NO	
Repositorio del proyecto	NO	
Respaldo del repositorio del proyecto	NO	
Acciones correctivas	SI	Bugs en Product Backlogs
Reporte de avance	SI	Charts, Kanban board

Solicitud de cambio	NO	
Configuración de software	SI	TFVC, GIT
Acta de aceptación	NO	
IMPLEMENTACION DE SOFTWARE		
Plan de proyecto (asignación de tareas)	SI	Product Backlogs, Tasks, Bugs, Test Cases, KANBAN BOARD.
Repositorio del proyecto	SI	TFVC, GIT.
Resultado de validación	SI	Product Backlogs.
Resultados de verificación	SI	Product Backlogs, Bugs.
Especificación de requisitos (Interfaces del proyecto)	SI	Product Backlogs.
Casos de prueba y procedimientos de prueba	SI	Test Case, test Manager.
Solicitud de cambio	NO	
Registro de trazabilidad	SI	Definición de Product Backlog, Feature.
Diseño de software	NO	
Componentes de software	SI	TFVC, GIT.
Reporte de pruebas	SI	Progress, ExploratoryTesting.
Configuración de software	SI	TFVC, GIT.
Manual de operaciones	SI	TFVC, GIT.
Manual de usuario	SI	TFVC, GIT.
Software	SI	Build.
Manual de mantenimiento	SI	TFVC, GIT.

Fuente: Elaboración propia

En resumen, de los 27 artefactos propuestos por la norma, 18 son cubiertos por *Team Foundation Service*, pero las 09 restantes deben ser cubiertas por otras herramientas complementarias.

08 artefactos son documentos del proyecto, se puede usar las herramientas de almacenamiento de archivos como Google Drive/OneDrive y correo corporativo Gmail/Hotmail ya que ambas tienen las mismas características. Para seleccionar la herramienta de almacenamiento de archivos se tomó como referencia la tabla 16.

Tabla 16: Herramienta de almacenamiento de archivos

Características	OneDrive	Dropbox	Google Drive
Espacio Libre	5 GB	2 GB	15 GB
Espacio Extra	No	Si	No
Plan de pago	\$ 2/mes por 50 GB	\$10/mes por 1 TB	\$2/mes por 100GB \$10/mes por 1 TB
Sistema Operativo soportado	Windows, Mac, Android, IOS, Windows Phone	Windows, Mac, Linux, Android, IOS, Windows Phone, BlackBerry, Kindle Fire	Windows, Mac, Android, IOS

Fuente: (Mitroff, 2016)

Para la investigación se seleccionó el Google Drive y por consecuente a Gmail, en la tabla 17 aplicamos estas herramientas con los 08 artefactos de la ISO/IEC 29100.

Tabla 17: Herramientas complementarias

Artefacto	Herramienta	Aplicación
Plan de Proyecto	Google Drive	Documentos
Enunciando del Trabajo	Google Drive	Documentos
Acta de Reunión	Google Drive	Fotos
Repositorio del Proyecto	Google Drive	
Respaldo de repositorio del proyecto	Google Drive	

Artefacto	Herramienta	Aplicación
Solicitud de Cambio de Procesos de Gestión	Gmail	Correo
Solicitud de Cambio de Procesos de Ingeniería	Gmail	Correo
Acta de Aceptación	Google Drive	Fotos

Fuente: Elaboración propia

Sin embargo para el diseño de software se deben usar herramientas con otras prestaciones. Eclipse es un IDE de desarrollo libre para proyectos de desarrollo de software, por lo tanto en la tabla 18 se visualiza que el IDE Eclipse mediante un plugin cubre el artefacto de diseño de software.

Tabla 18: Herramientas de Diseño e Integración

Artefacto	Herramienta
Diseño de software.	Eclipse IDE UML2 Plugin.

Fuente: Elaboración propia

7.2. MAPEO DE PROCESOS

Los procesos propuestos por la norma son cubiertos en su totalidad con las herramientas de la tabla inferior.

Tabla 19: Mapeo de procesos

Proceso	TFS	Herramienta
Gestión de proyecto		
Planeación.	SI	ALM TFS, G. Drive.

Ejecución del plan.	SI	ALM TFS, G. Drive, Gmail.
Evaluación y control.	SI	ALM TFS, G. Drive, Gmail.
Cierre.	SI	ALM TFS, G. Drive.
Implementación de software		
Inicio.	SI	ALM TFS, G Drive.
Análisis de requerimientos.	SI	ALM TFS, G Drive.
Arquitectura y diseño.	NO	Eclipse IDE plugin UML2, G Drive.
Construcción.	SI	ALM TFS, Eclipse IDE.
Integración y pruebas.	SI	ALM TFS, Jenkins, Eclipse IDE.
Entrega de productos.	SI	ALM TFS, Eclipse IDE, G. Drive.

Fuente: Elaboración propia

Como se puede ver los procesos y la evidencia del ejercicio de estos deben trabajar de manera conjunta con las demás herramientas, el único proceso donde no participa el ALM seleccionado es en la Arquitectura y Diseño.

Para cubrir el proceso de Integración y pruebas se ha seleccionado la herramienta de software libre Jenkins.

7.3. DEFINICIÓN DE LA COMBINACIÓN DE HERRAMIENTAS

La mayoría de procesos están cubiertas por la herramienta ALM *Team Foundation Service*, con el apoyo de herramientas complementarias.

Por tanto esta sería la combinación de herramientas recomendadas:

- Team Foundation Service.
- Eclipse IDE, Plugin UML2.

- Jenkins.
- Google Drive.
- Gmail

VIII. CASO DE ESTUDIO

8.1. CARACTERÍSTICAS DEL PROYECTO DE APLICACIÓN

Es un proyecto de desarrollo de software para el sector hotelero denominado “RESTORI” diseñado para apoyar las ventas del sector hostelero, concretamente para bares y restaurantes.

8.1.1. DEL PROYECTO

El proyecto se desarrolló en 5 meses de acuerdo a la planificación del mismo, y conto con los siguientes roles:

Tabla 20: Roles según la ISO/IEC 29110

Rol	Acrónimo
Gestor de Proyecto	GP
Analista	AN
Diseñador	DI
Líder Técnico	LT
Programador	PG
Cliente	CL
Equipo de Trabajo	ET

Fuente: Elaboración propia

8.1.2. DEL SOFTWARE

El software se describe basado en sus características funcionalidad.

Tabla 21: Requisitos de software

Requisitos funcionales de alto nivel
Módulo de Almacén
Módulo de Producción
Módulo de Pedido.
Módulo de Venta.
Requisitos no funcionales
Codificación: AngularJS, Java Spring MVC, Hibernate.
Base de datos: PostgreSQL 9+.
Servicio IAAS: AWS, Ubuntu 14+.
Arquitectura Orientada a Servicios bajo REST.

Fuente: Elaboración propia

8.2. DISEÑO Y PLANIFICACION DEL CASO

8.2.1. Exposición de la ISO/IEC 29110 y sus respectivas Herramientas

Se explica a los involucrados del proyecto los procesos y artefactos que la ISO comprende, además de explicar el uso de las herramientas seleccionadas de las que ya el equipo conoce pero es necesario explicar la coordinación e integración entre ellas, con el fin de que sepan que ellos trabajaran de manera natural, pero este proyecto aporta a una investigación.

8.2.2. Diseño de la ejecución encuesta

Se elabora una encuesta en formato de cuestionario basado en Google Forms el cual se puede apreciar en el Anexo 13.3 a 13.8, para que los involucrados de manera digital puedan al final de los 05 meses del proyecto responder mediante escala de Likert el nivel de satisfacción ante los procesos y artefactos cubiertas por las herramientas elegidas para el caso de estudio.

8.2.3. Análisis estadístico

Al contar con los resultados de los involucrados, se procede mediante análisis estadístico de medias verificar si la satisfacción supera el puntaje de 4 (valoración satisfactoria), tanto los procesos de gestión e ingeniería, así como de los artefactos también de gestión y de ingeniería.

8.3. RESULTADOS DEL CASO

El proyecto es ejecutado creando un proyecto en *Team Foundation Service* con los parámetros de Scrum, enfocado en el desarrollo por funcionalidades, del que se documenta el cumplimiento con los artefactos y procesos de la ISO/IEC 29110.

Como también se obtuvo la calificación de satisfacción de los responsables del proyecto con los artefactos propuestos por la ISO/IEC 29110 con el uso de las herramientas.

Tabla 22: Resultados de caso

Artefacto	Inicio (# Mes)	Herramienta	Resp.	(*) Calif. (1-5)
GESTIÓN DE PROYECTOS				
Plan de proyecto	1	G. Drive	GP	5
Enunciado del trabajo	1	G. Drive	GP	5
Resultado de verificación	5	Product Backlog	CL	3
Acta de reunión	1	G. Drive	GP	5
Repositorio del proyecto	1	G. Drive	GP	5
Respaldo del repositorio del proyecto	1	G. Drive	GP	5
Acciones correctivas	4	Bugs	AN	5
Reporte de avance	2	Charts, Progress, KANBAN BOARD	GP	4
Solicitud de cambio	4	Gmail	CL	2
Configuración de software	1	G. Drive	GP	5
Acta de aceptación	5	G. Drive	GP	5
IMPLEMENTACION DE SOFTWARE				
Plan de proyecto (asignación de tareas)	1	Product Backlogs, Tasks, Bugs, Test Cases, KANBAN BOARD.	AN	3
Repositorio del proyecto	1	TFVC, GIT.	LT	5
Resultado de validación	5	Product Backlogs.	LT	4
Resultados de verificación	5	Product Backlogs, Bugs.	LT	3

Especificación de requisitos (Interfaces del proyecto)	1	Product Backlogs.	AN	5
Casos de prueba y procedimientos de prueba	4	Test Case, test Manager.	DI	4
Solicitud de cambio	4	Product Backlog	AN	5
Registro de trazabilidad	2	Definición de Product Backlog, Feature.	AN	3
Diseño de software	1	Eclipse UML2	DI	3
Componentes de software	1	TFVC, GIT.	PG	5
Reporte de pruebas	5	Progress, ExploratoryTesting.	PG	4
Configuración de software	3	Jenkins	LT	5
Manual de operaciones	6	TFVC, GIT.	PG	5
Manual de usuario	6	TFVC, GIT.	PG	5
Software	1	Build.	PG	5
Manual de mantenimiento	6	TFVC, GIT.	PG	5
(*) Leyenda: Totalmente Satisfecho (5), Satisfecho(4), Poco Satisfecho (3), Insatisfecho(2), Totalmente Insatisfecho (1)				

Fuente: Elaboración propia

Los procesos también fueron calificados por los responsables:

Tabla 23: Calificación de procesos

Proceso	Herramienta	Resp.	(*) Calif. (1-5)
Gestión de proyecto			
Planeación.	ALM TFS, G. Drive.	GP	4
Ejecución del plan.	ALM TFS, G. Drive.	GP	4

Evaluación y control.	ALM TFS, G. Drive.	GP	4
Cierre.	ALM TFS, G. Drive.	GP	5
Implementación de software			
Inicio.	ALM TFS, G Drive.	LT	4
Análisis de requerimientos.	ALM TFS, G Drive.	AN	3
Arquitectura y diseño.	Eclipse IDE, G Drive.	DI	4
Construcción.	ALM TFS, Eclipse IDE.	PG	5
Integración y pruebas.	ALM TFS, Jenkins, Eclipse IDE.	PG	3
Entrega de productos.	ALM TFS, Eclipse IDE, G. Drive.	LT	5
(*) Leyenda: Totalmente Satisfecho (5), Satisfecho(4), Poco Satisfecho (3), Insatisfecho(2), Totalmente Insatisfecho (1)			

Fuente: Elaboración propia

El repositorio en Google Drive quedó conformado por múltiples archivos que se fueron modificando durante el desarrollo normal del proyecto, y este podía ser compartido con el cliente directamente con permisos de lectura y escritura según era la necesidad.

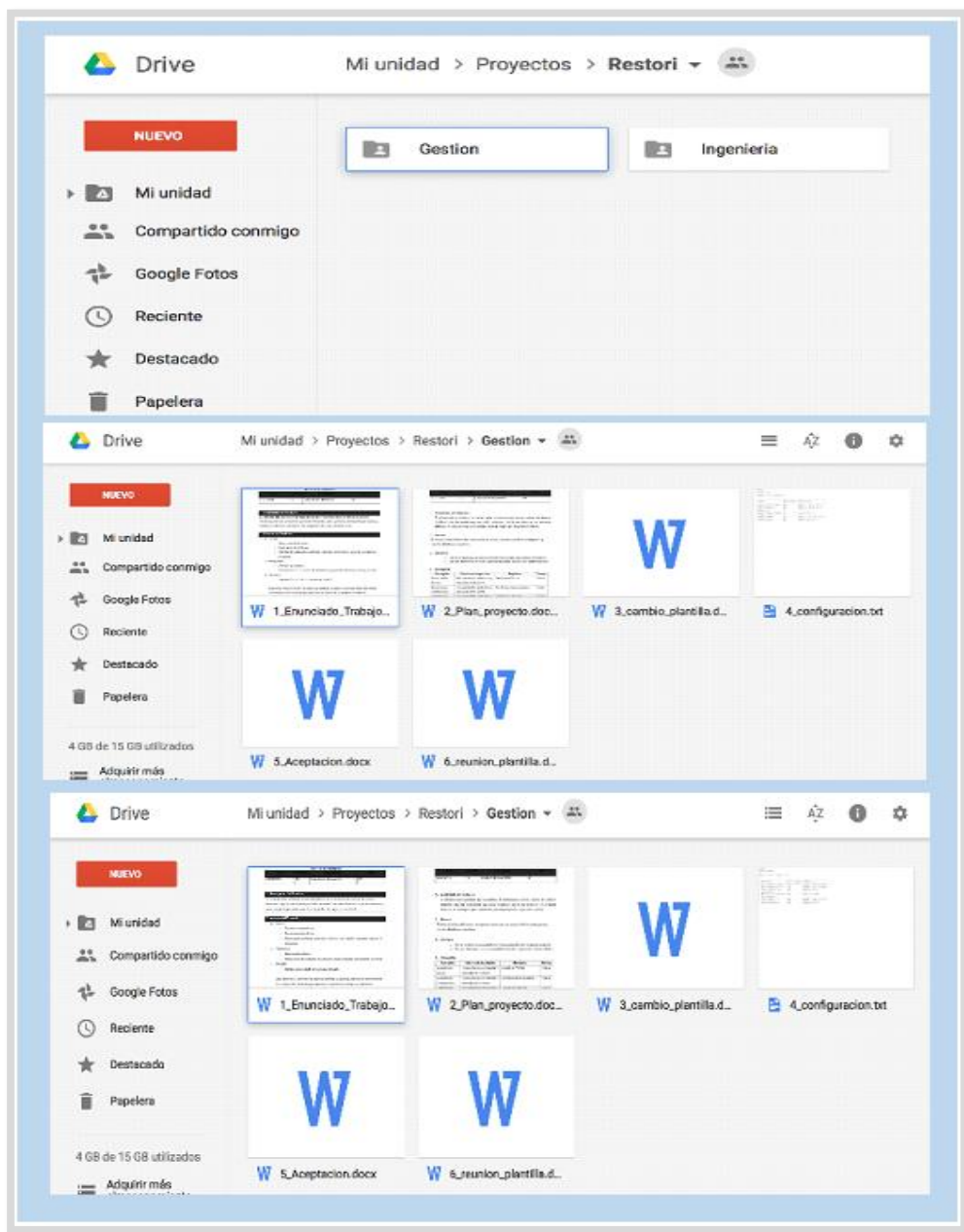


Figura 28: Vistas del Drive del proyecto

Fuente: Elaboración propia

El proyecto en TFS quedo configurado y ejecutado con facilidad por parte de Gestor de Proyecto (GP), como muestra la calificación en la tabla 22.

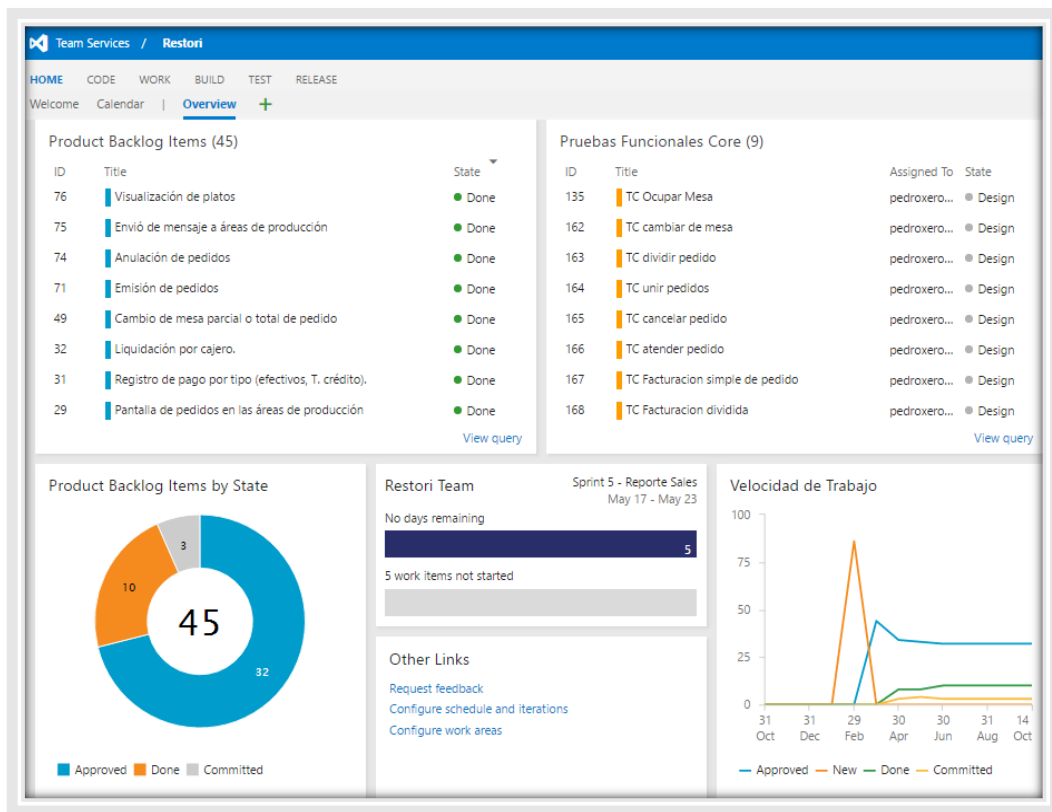


Figura 30: Vista 1 de TFS en el proyecto
Fuente: Elaboración propia

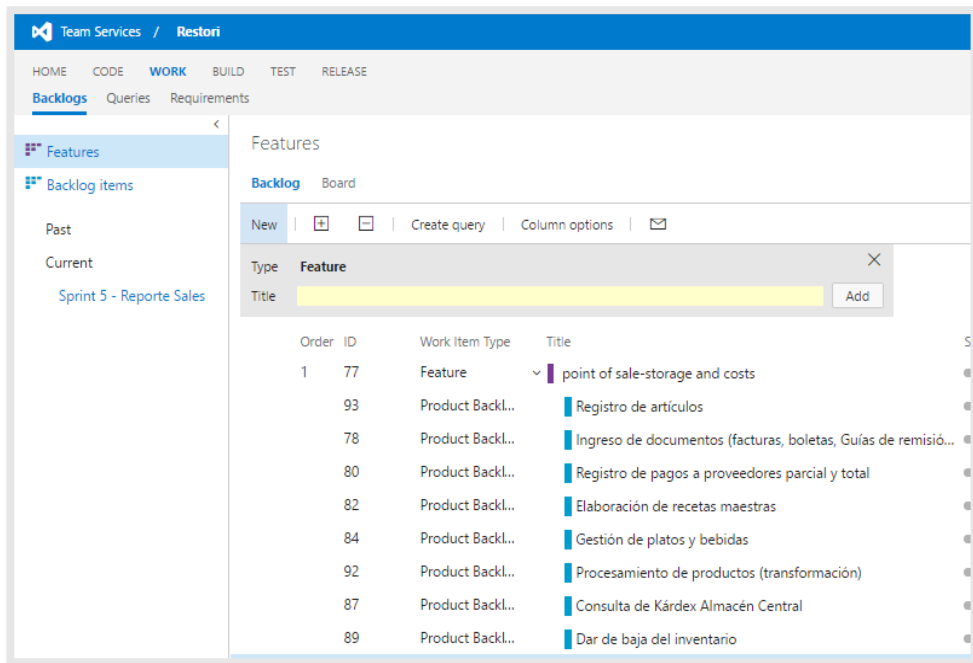


Figura 29: Vista 2 de TFS en el proyecto
Fuente: Elaboración propia

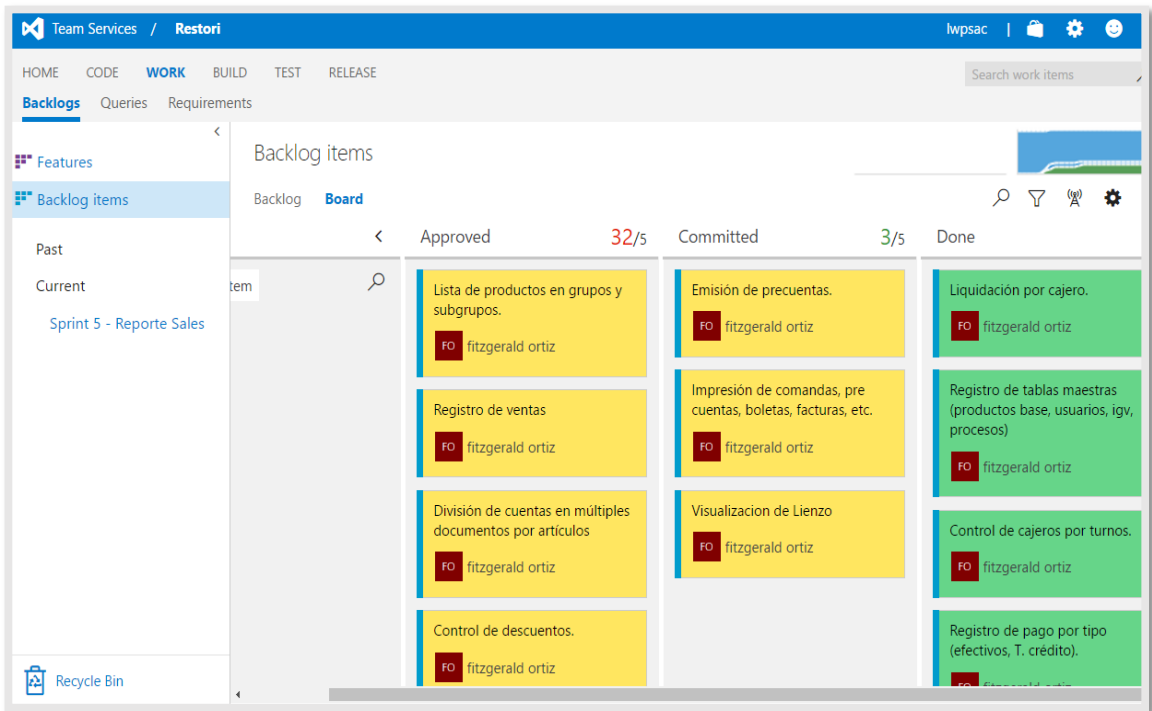


Figura 32: Vista 3 de TFS en el proyecto
Fuente: Elaboración propia

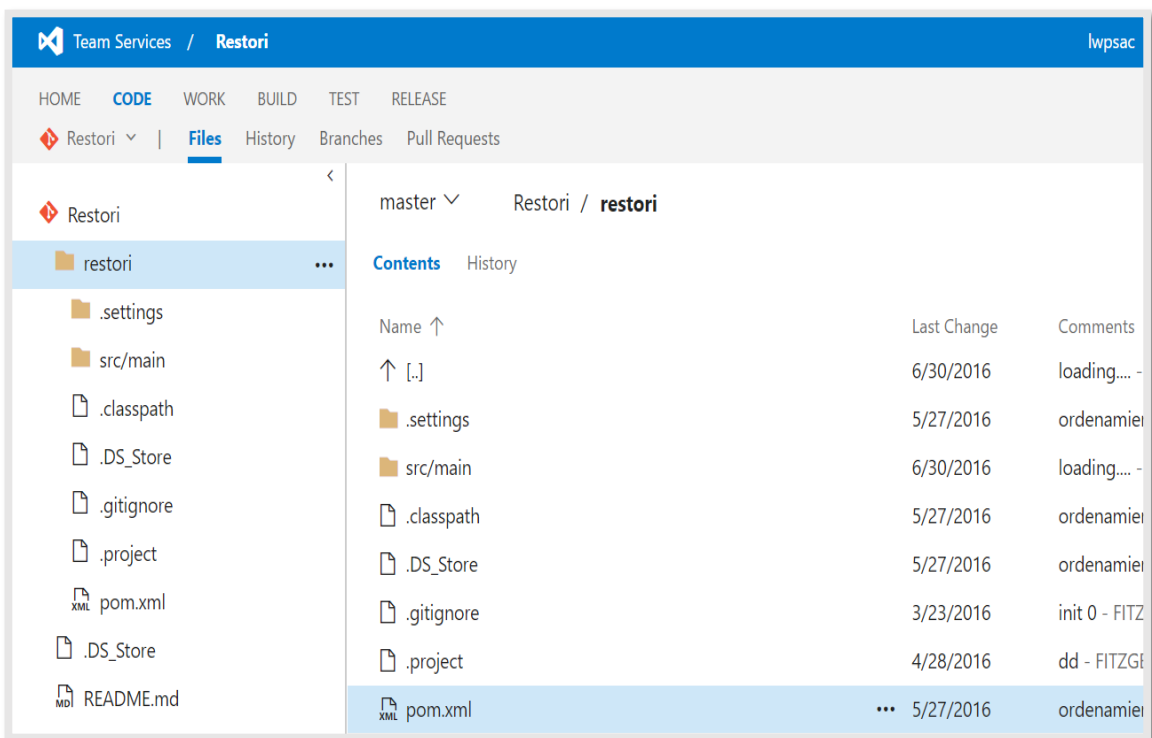


Figura 31: Vista 4 de TFS en el proyecto
Fuente: Elaboración propia

La construcción en eclipse estaba conectado con Git con el proyecto creado en TFS, y la entrega con Jenkins también conectado al proyecto en TFS a través de Git.

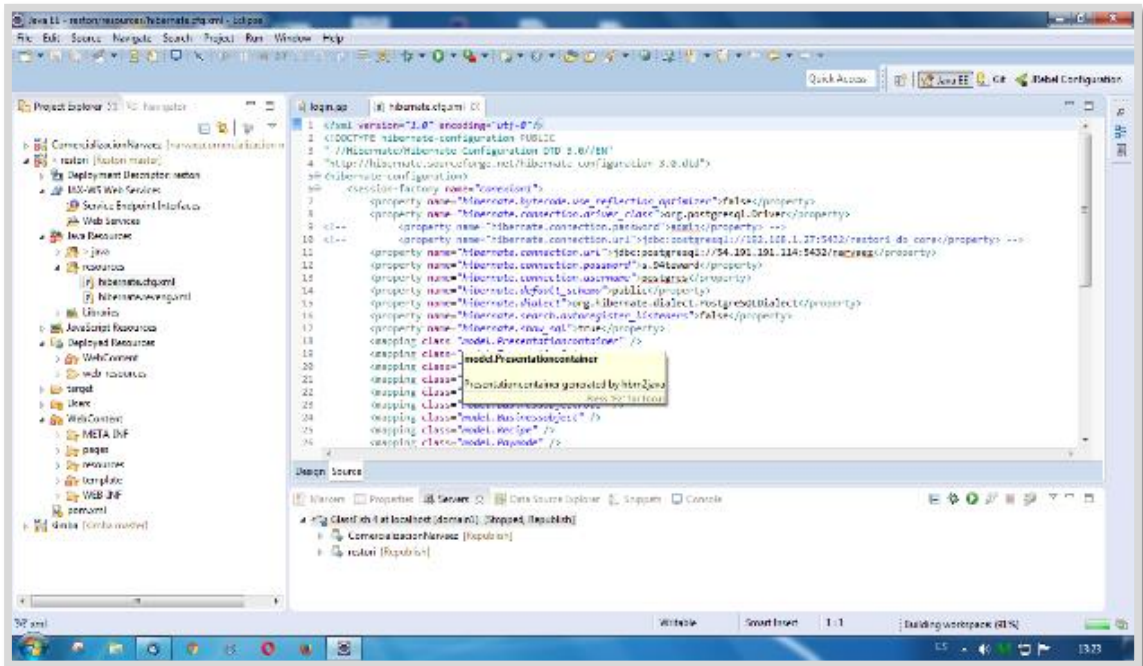


Figura 33: Vistas del código e interacción del proyecto
Fuente: Elaboración propia

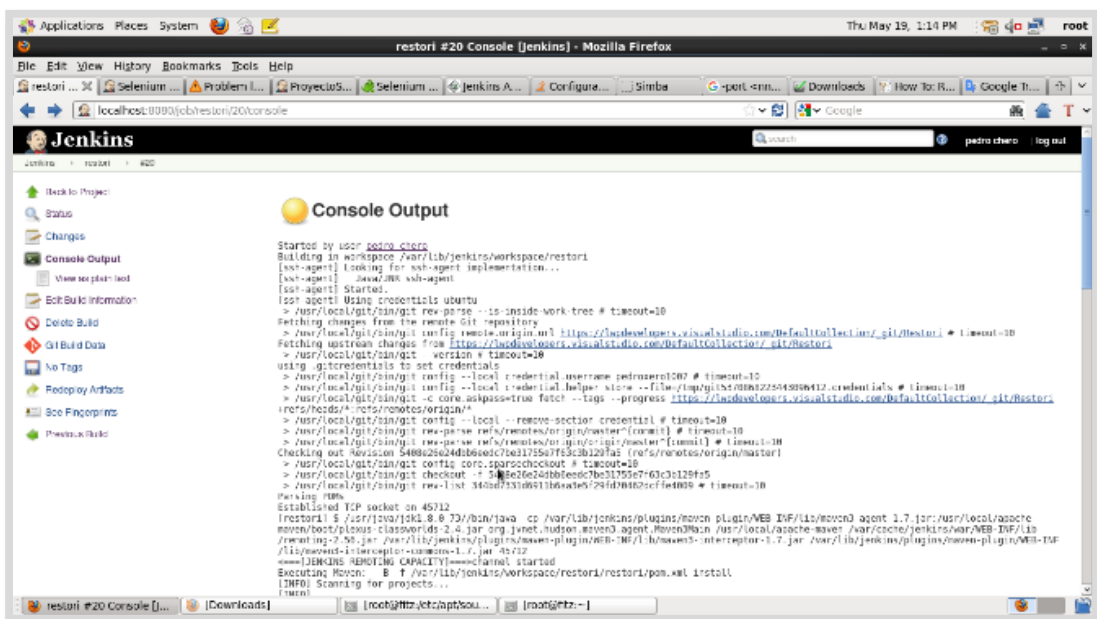


Figura 34: Vistas del código e integración del proyecto
Fuente: Elaboración propia

IX. DISCUSIÓN

Después de haber determinado y evaluado las herramientas de ingeniería de Software para la implementación de la ISO/IEC 29110 en una PYME, tomando el caso de la empresa Lead Working Partner, se han obtenido los siguientes resultados:

9.1. ARTEFACTOS CUBIERTOS

Las herramientas elegidas han logrado cubrir gran parte del proceso, ellas se detallan en la figura 35 y tabla 24, siguiente:

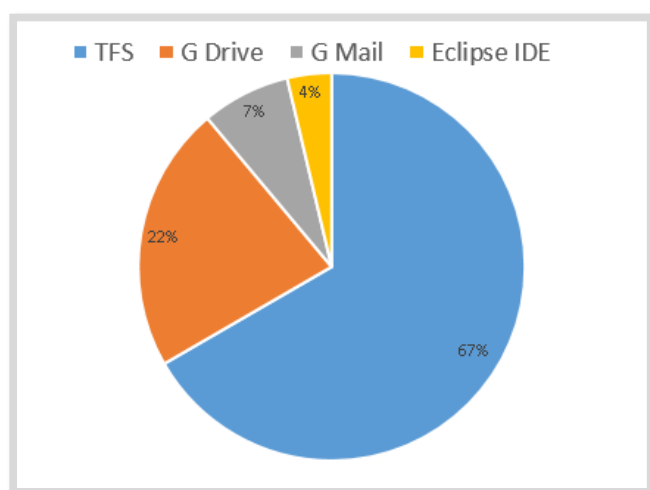


Figura 35: Artefactos cubiertos en el proyecto
Fuente: Elaboración propia

Tabla 24: Artefactos de la ISO/IEC 29110 Cubiertos en el proyecto

Herramienta	Artefactos	% Cumplimiento
TFS	18	67%
G Drive	6	22%
G Mail	2	7%
Eclipse IDE	1	4%
Total	27	100%
Total ISO	27	
Pendientes	0	

Fuente: Elaboración propia

De las herramientas seleccionadas el ALM TFS cubre el 67% de artefactos.

Muchos de los documentos sugeridos por la ISO están cubiertos por Google Drive, por lo que sirve como una gran herramienta de apoyo al ALM.

En la figura 26 y tabla 25 podremos apreciar la aplicación de las herramientas combinadas para cubrir los procesos de la ISO/IEC 29110

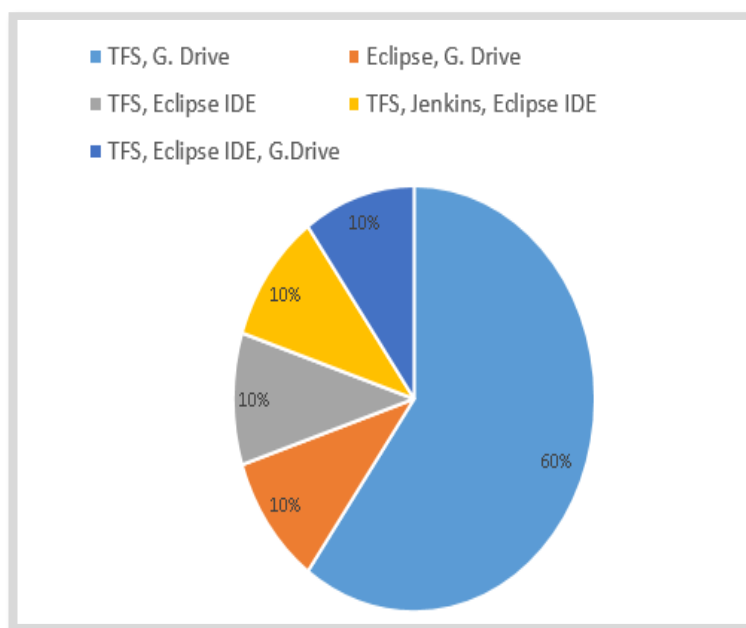


Figura 36: Procesos cubiertos en el proyecto

Fuente: Elaboración propia

Tabla 25: Procesos de la ISO/IEC 29100 cubiertos en el proyecto

Combinación Herramienta	Procesos	% Cumplimiento
TFS, G. Drive	6	60%
Eclipse, G. Drive	1	10%
TFS, Eclipse IDE	1	10%
TFS, Jenkins, Eclipse IDE	1	10%
TFS, Eclipse IDE, G. Drive.	1	10%
Total	10	100%
Total ISO	10	
Pendientes	0	

Fuente: Elaboración propia

Eclipse a pesar de cubrir 02 procesos, es muy importante porque es el núcleo de todo el producto.

9.2. VALORACIÓN DE HERRAMIENTAS EN LOS ARTEFACTOS

Los involucrados han calificado el uso de las herramientas para trabajar con los artefactos con valores favorables de acuerdo a los resultados obtenidos de la tabla 22, donde la mayoría de los involucrados del proyecto expresan estar totalmente satisfecho en el uso de las herramientas propuestas. Del gráfico 37 se puede ver que los artefactos de resultados de verificación, reportes de avance y sobre todo solicitudes de cambio pueden mejorarse de alguna manera, pero en la mayoría ha sido calificado como idóneo.



Figura 37: Valoración de artefactos de gestión ISO/IEC 29110

Fuente: Elaboración propia

Según la figura 38, de los 16 artefactos de Ingeniería de la ISO/IEC 29110, 12 tienen una valoración bastante satisfactoria, y 04 pueden mejorarse. Pero los resultados no son desalentadores.



Figura 38: Valoración de artefactos de implementación ISO/IEC 29110

Fuente: Elaboración propia

9.3. VALORACIÓN DE LAS HERRAMIENTAS EN LOS PROCESOS

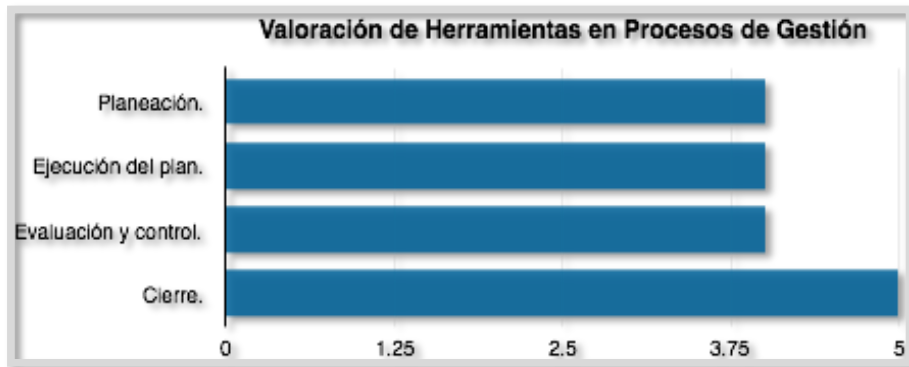


Figura 39: Valoración de procesos de gestión ISO/IEC 29100

Fuente: Elaboración propia

Según la figura 39, las herramientas que se asignaron para los procesos de gestión también han tenido buenas calificaciones, con opciones a mejorar.

En la figura 40 se puede apreciar a los procesos de implementación con resultados bastante favorables y con posibilidades de mejora sobre todo en los procesos de análisis de requerimientos y de Integración y pruebas.



Figura 40: Valoración de procesos de implementación ISO/IEC 29110

Fuente: Elaboración propia

9.4. ANALISIS DE MEDIAS DE ARTEFACTOS Y PROCESOS DE LA ISO/IEC

29100

Mediante el análisis de medias de la tabla 26 se puede verificar que tanto los artefactos y los procesos superan la media estimada de 4 (satisfecho), determinando que las herramientas elegidas son bastante favorable para el caso estudiado, por lo que se puede recomendar el uso de esta combinación de herramientas cuando una PYME quiere implementar esta ISO, ya que según el caso estudio facilitaría con gran satisfacción esta implementación

Tabla 26: Análisis de Medias de procesos y artefactos de la ISO/IEC 29110 cubiertos en el proyecto

	Artefactos de Proceso de Gestión	Artefactos de Proceso de Ingeniería	Procesos de Gestión	Procesos de Ingeniería
Media	4.45	4.31	4.25	4.00
N	11	16	4	6
Desv. Tip.	1.036	0.873	0.500	0.894

Fuente: Elaboración propia

X. CONCLUSIONES

Se logró determinar las herramientas de software que cubren los procesos de la implementación de la ISO/IEC 29110, a través del Proyecto “RESTOR”, Software para el sector Hotelero desarrollado por la PYME Lead Working Partner SAC.

- Como principal herramienta se seleccionó a TFS (*Team Foundation Service*), propiedad de la empresa Microsoft, empresa líder en el mercado de herramientas ALM (*Application Lifecycle Management*) según Gartner del 2015, teniendo en cuenta los criterios de costo y tendencia de uso. A la vez permitió seleccionar las herramientas complementarias como Eclipse IDE, quien se integra con el TFS. También se seleccionó en los procesos de Gestión del Proyecto a Google Drive y Gmail, ambos para la gestión de documentos, por último se seleccionó las herramientas Eclipse UML2 Plugin y Jenkins para el Diseño e Integración continua en los procesos de Implementación de Software.

- Mediante el análisis de media de la Tabla 26: Análisis de Medias de procesos y artefactos de la ISO/IEC 29110 cubiertos en el proyecto se pudo verificar que la combinación de las herramientas seleccionadas para cubrir los

procesos de la ISO/IEC 29110 superan la media estimada de 4(Satisfecho), por el cual se concluye que el uso de estas herramientas facilitarían la implementación de la ISO.

XI. RECOMENDACIONES

- Se recomienda que se utilicen las herramientas de software identificadas para la implementación de la ISO/IEC 29110 en otras PYMES de la industria del software, con el propósito de producir software alineado a estándares internacionales.

- Se sugiere la instalación de las herramientas identificadas en la investigación, *TEAM FOUNDATION SERVICE*, para una óptima gestión de proyectos e implementación de software. Esta herramienta ahorra costos y permite trabajar con IDE de desarrollo, Drive para almacenamiento de archivos y herramientas de integración continua, permitiendo tener un mejor seguimiento en el desarrollo de software.

- Se recomienda a las PYMES de la industria del software contratar a un personal con experiencia o conocimiento en la ISO/IEC 29110, para facilitar el proceso de implementación de la misma.

XII. BIBLIOGRAFÍA

1. Abarca Romero, M., Arisaca Mamani, R., & Dávila Ramón, A. (2015). *Implementación del Perfil Básico de la ISO/IEC 29110 de una Pequeña Empresa Desarrolladora de Software: Lecciones Aprendidas*. Obtenido de http://eventos.spc.org.pe/cibse2015/pdfs/06_IT15.pdf
2. Alarcón Aldana, A. C., González Sanabria, J. S., & Rodríguez García, S. L. (2011). *Guía para pymes desarrolladoras de software, basada en la norma ISO/IEC 15504*. Recuperado el 30 de 06 de 2016, de <http://revistavirtual.ucn.edu.co/index.php/RevistaUCN/article/viewFile/339/651>
3. Aplicadas, U. d. (s.f.). *Repositorio Académico UPC*. Recuperado el 2016, de <http://repositorioacademico.upc.edu.pe/upc/>
4. Área de Sistemas de la Vicerrectoría de la Docencia de la Benemérita Universidad Autónoma de Puebla. (5 de Julio de 2008). *Área de Sistemas de la Vicerrectoría de la Docencia de la Benemérita Universidad Autónoma de Puebla*. Obtenido de <https://sistemasvd.wordpress.com/2008/07/05/fases-del-proceso-de-desarrollo-del-software/>
5. Atlassian. (s.f.). *Jira Software*. Recuperado el 2016, de <https://es.atlassian.com/software/jira>
6. Cardona, A. E. (2011). *Metodología para la elaboración del mapa estratégico de tecnologías de información y comunicaciones para instituciones de educación superior en Colombia usando el Balanced Scorecard para TI*. Universidad Nacional de Colombia, Facultad de Minas, Escuela de Sistemas, Medellín. Obtenido de http://www.bdigital.unal.edu.co/5475/1/98568918.2011_1.pdf
7. Costa, E. A. (16 de Noviembre de 2015). *Agencia de la Costa*. Obtenido de <http://agenciadelacosta.com.ar/cuadrante-magico-de-gartner/>
8. Cuentas Marant, R., & Ramos Castro, J. (Setiembre de 2015). *Repositorio académico UPC*. Obtenido de http://repositorioacademico.upc.edu.pe/upc/bitstream/10757/582125/1/RAMOS_CJ.pdf
9. Edery, D. (Junio de 2008). *Industria del Software: - Industria del Software en la India. Conocimiento Peruano para el mercado mundial. Negocios Internacionales*. Obtenido de <http://www.comexperu.org.pe/media/files/revista/junio08%5Cespecial130.pdf>

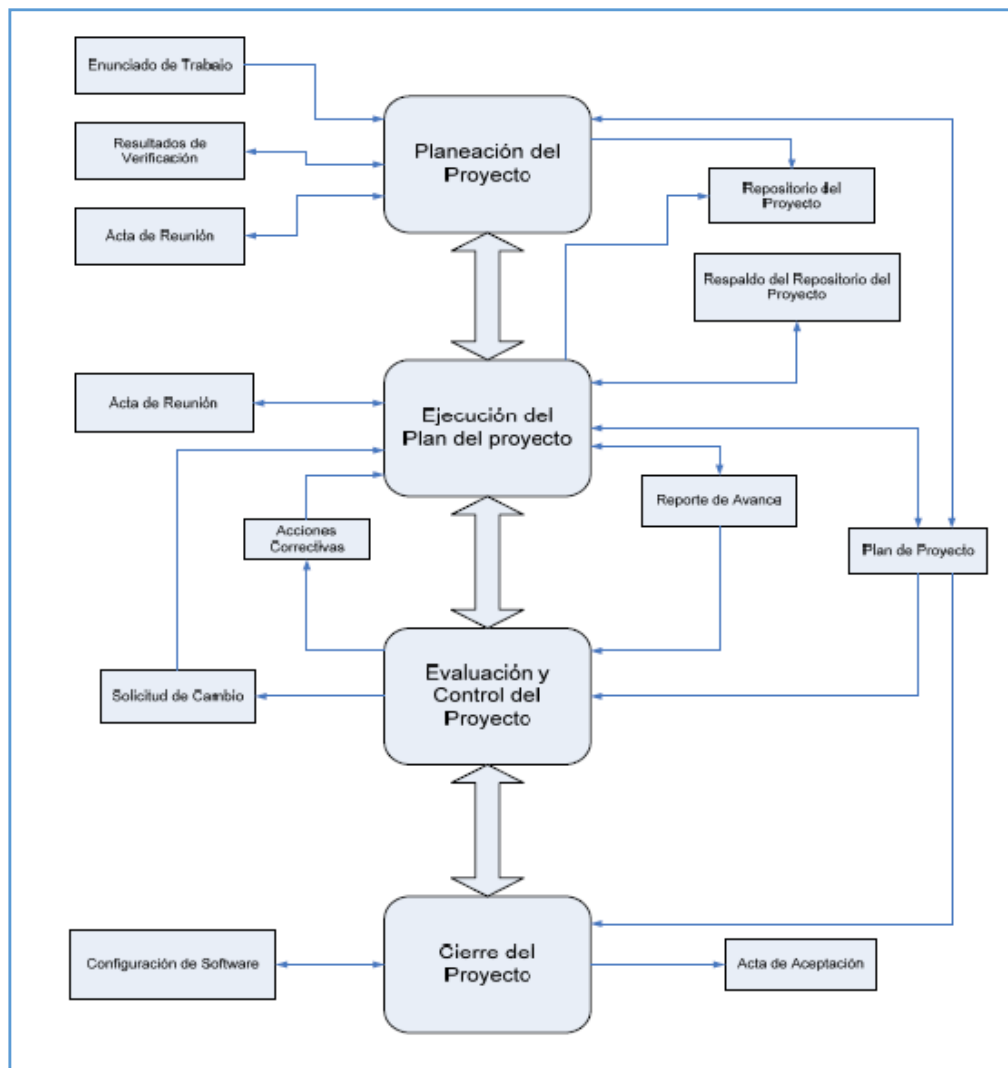
10. Espinoza Montes, C. (2014). *Metodología de Investigación Tecnológica Pensando en Sistemas* (Segunda Edición ed.). (C. Espinoza Montes, Ed.) Huancayo, Huancayo, Perú: Soluciones Gráficas S.A.C. Recuperado el 2017
11. Gartner. (s.f.). *Gartner symposium itxpo*. Recuperado el 2016, de <http://www.gartner.com/technology/home.jsp>
12. Getapp. (s.f.). *Desarrollo de aplicaciones móviles*. Recuperado el 2016, de <https://www.getapp.com/>
13. Gonzalez, M. P. (2011). *Propuesta de un Modelo de Implantación para un Sistema de Monitoreo de Infraestructura de TI*. Universidad Simon Bolivar, Decanato de Estudios de Postgrado, Caracas. Obtenido de <http://159.90.80.55/tesis/000151303.pdf>
14. Google. (s.f.). *Buscador Google*. Recuperado el 2016, de <https://www.google.com.pe/>
15. Hanna, O., Claudia, A., & Angelica Su, R. (Mayo de 2003). *Universidad Veracruzana*. Obtenido de <https://www.uv.mx/rrojano/MIS/desarrollo1/material/moprosoft-v1.1.pdf>
16. IBM. (2016). *IBM Systems Magazine*. Recuperado el 2016, de http://www.ibmssystemsmag.com/mainframe/trends/Modernization/CICS_RTC/?page=2
17. IBM. (s.f.). *IBM*. Recuperado el 2016, de <https://www.ibm.com/>
18. InfoQ. (Febrero de 23 de 2015). *InfoQ*. Obtenido de <https://www.infoq.com/news/2015/02/agile-management-tools>
19. Juan, B. (2011). *Gestión de Procesos (Alineados con la Estrategia)* (4ta ed.). Santiago de Chile, Chile: Evolución S.A. Obtenido de http://www.evolucion.cl/resumenes/Resumen_libro_Gesti%F3n_de_procesos_JBC_2011.pdf
20. Microsoft. (s.f.). *Visual Studio Team Services*. Recuperado el 2016, de <https://www.visualstudio.com/es/team-services/>
21. Mitroff, S. (01 de 02 de 2016). *CNET*. Obtenido de <https://www.cnet.com/how-to/onedrive-dropbox-google-drive-and-box-which-cloud-storage-service-is-right-for-you/>
22. Pesado, P., Bertone, R., Esponda, S., Pasini, A., & Martorelli, S. (2009). *Calidad en el desarrollo de Sistemas de Software*. Buenos Aires: Instituto de Investigación en Informática LIDI, Facultad de Informática – UNLP. Obtenido de http://digital.cic.gba.gov.ar/bitstream/handle/11746/3835/11746_3835.pdf-PDFA.pdf?sequence=1&isAllowed=y
23. Pinto, Garcia, Ruiz, & Piattini. (2016). *Adaptación de las Normas ISO/IEC 12207:2002 e ISO/IEC 15504:2003 para la Evaluación de la Madurez de Procesos Software en Países*

en Desarrollo. IEEE LATIN AMERICA TRANSACTIONS. Obtenido de <http://ieeexplore.ieee.org/stamp/stamp.jsp?reload=true&arnumber=1642455>

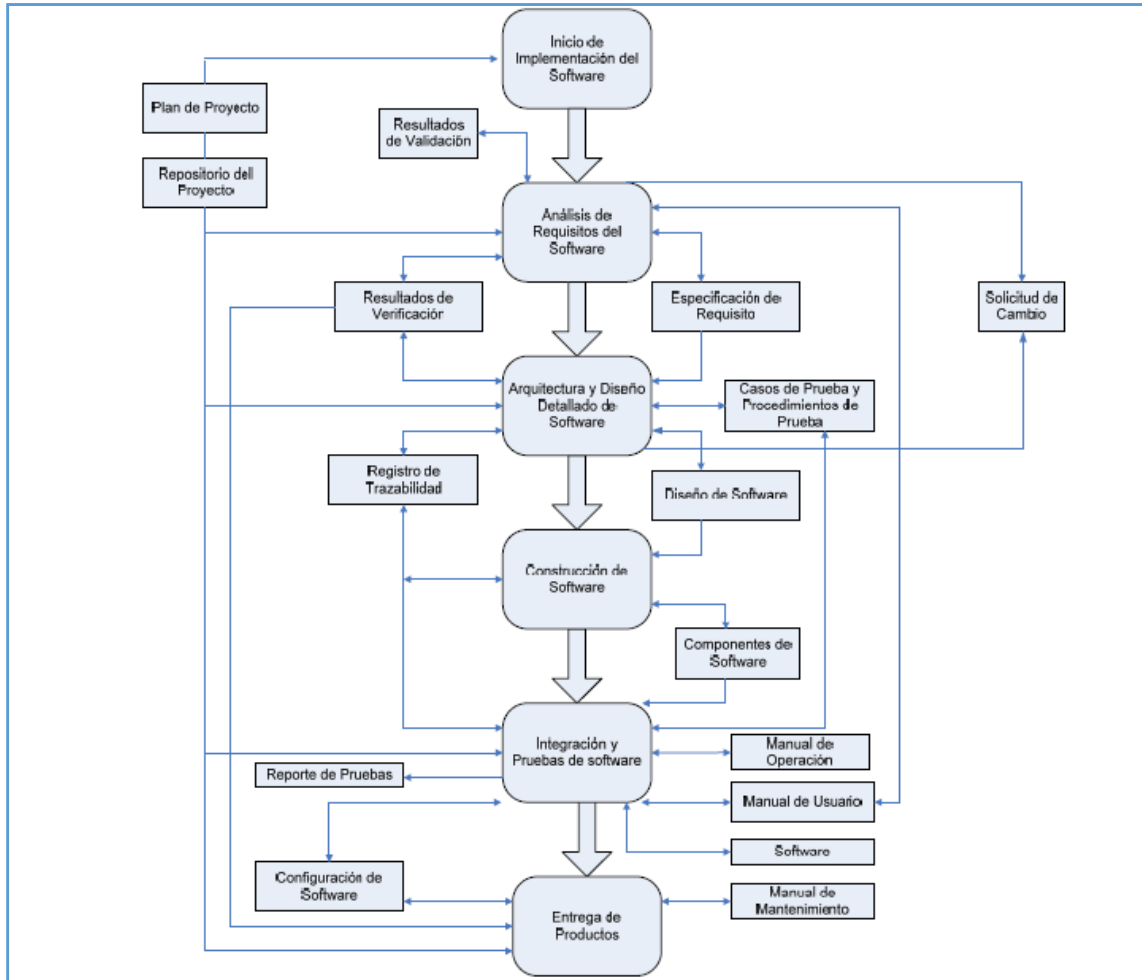
24. Rally. (s.f.). *rallydev*. Recuperado el 2016, de <https://www.rallydev.com/>
25. Ramos, C., & Mendoza, L. (2014). *Implementación del estándar ISO/IEC 29110-4-1 para pequeñas organizaciones de desarrollo de software*. Lima: Universidad Peruana de Ciencia Aplicada.
26. Rodriguez Forero, C., Andres Mauricio, C., & Chacon Torres, J. (2016). *Wordpress*. Obtenido de <https://sistemasvd.wordpress.com/2008/07/05/fases-del-proceso-de-desarrollo-del-software>.
27. Sommerville, I. (2011). *Ingeniería de Software* (Novena ed.). (V. Campos, Trad.) Mexico, Mexico: PEARSON EDUCACIÓN.
28. Statista. (2013). *Statisa*. Obtenido de <https://www.statista.com/statistics/376669/worldwide-packaged-software-spending-forecast/>
29. Trends, G. (s.f.). *Google Trends*. Recuperado el 2016, de <https://www.google.es/trends/>
30. VersionOne. (2016). *VersionOne*. Recuperado el 2016, de <http://info.versionone.com/v1-trial-free.html>

XIII. ANEXO

13.1. GRUPO DE PROCESOS DE GESTION DE LA ISO/IEC 29110



13.2. GRUPO DE PROCESOS DE INGENIERIA DE LA ISO/IEC 29110



13.3. RESULTADO DE EVALUACIÓN DEL GESTOR DE PROYECTO

Artefacto	Herramienta	Resp.	Calif. (1-5)	Observación
GESTIÓN DE PROYECTOS				
Plan de proyecto	Drive	GP	5	
Enunciado del trabajo	Drive	GP	5	
Acta de reunión	Drive	GP	5	
Repositorio del proyecto	Drive	GP	5	
Respaldo del repositorio del proyecto	Drive	GP	5	
Reporte de avance	Charts, Progress, KANBAN	GP	4	
Configuración de software	Drive	GP	5	
Acta de aceptación	Drive	GP	5	

Proceso	Herramienta	Resp.	Calif. (1-5)	Observaciones
Gestión de proyecto				
Planeación.	ALM TFS, G. Drive.	GP	4	La planificación según Scrum es muy buena, pero necesito un poco de control sobre las fechas y debo esta revisando para lograrlo.
Ejecución del plan.	ALM TFS, G. Drive.	GP	4	
Evaluación y control.	ALM TFS, G. Drive.	GP	4	
Cierre.	ALM TFS, G. Drive.	GP	5	

13.4. RESULTADO DE EVALUACIÓN DEL ANALISTA

Artefacto	Herramienta	Resp.	Calif. (1-5)	Observación
GESTIÓN DE PROYECTOS				
Acciones correctivas	Bugs	AN	5	
IMPLEMENTACION DE SOFTWARE				
Plan de proyecto (asignación de tareas)	Product Backlogs, Tasks, Bugs, Test Cases, KANBAN.	AN	3	No se puede planificar como un gantt para aproximar fechas.
Especificación de requisitos (Interfaces del proyecto)	Product Backlogs.	AN	5	
Solicitud de cambio	Product Backlog	AN	5	
Proceso	Herramienta	Resp.	Calif. (1-5)	Observaciones
Implementación de software				
Análisis de requerimientos.	ALM TFS, G Drive.	AN	3	La herramienta permite registrar los requerimientos, pero necesito gestionar con una matriz.

13.5. RESULTADO DE EVALUACIÓN DEL DISEÑADOR

Proceso	Herramienta	Resp.	Calif. (1-5)	Observaciones
Implementación de software				
Arquitectura y diseño.	Eclipse IDE, G Drive.	DI	4	

Artefacto	Herramienta	Resp.	Calif. (1-5)	Observación
IMPLEMENTACION DE SOFTWARE				
Casos de prueba y procedimientos de prueba	Test Case, test Manager.	DI	4	
Diseño de software	Eclipse UML2	DI	3	La herramienta es poco intuitiva.

13.6. RESULTADO DE EVALUACIÓN DEL PROGRAMADOR

Artefacto	Herramienta	Resp.	Calif. (1-5)	Observación
IMPLEMENTACION DE SOFTWARE				
Componentes de software	TFVC, GIT.	PG	5	
Reporte de pruebas	Progress, Exploratory Testing.	PG	4	
Manual de operaciones	TFVC, GIT.	PG	5	
Manual de usuario	TFVC, GIT.	PG	5	
Software	Build.	PG	5	
Manual de mantenimiento	TFVC, GIT.	PG	5	

Proceso	Herramienta	Resp.	Calif. (1-5)	Observaciones
Implementación de software				
Construcción.	ALM TFS, Eclipse IDE.	PG	5	
Integración y pruebas.	ALM TFS, Jenkins, Eclipse IDE.	PG	3	Se necesita revisar la cobertura de código.

13.7. RESULTADO DE EVALUACIÓN DEL LIDER TÉCNICO

Artefacto	Herramienta	Resp.	Calif. (1-5)	Observación
IMPLEMENTACION DE SOFTWARE				
Repositorio del proyecto	TFVC, GIT.	LT	5	
Resultado de validación	Product Backlogs.	LT	4	
Resultados de verificación	Product Backlogs, Bugs.	LT	3	Me parece Microsoft tiene una herramienta mas elaborada para esto.
Configuración de software	Jenkins	LT	5	

Proceso	Herramienta	Resp.	Calif. (1-5)	Observaciones
Implementación de software				
Inicio.	ALM TFS, G Drive.	LT	4	
Entrega de productos.	ALM TFS, Eclipse IDE, G. Drive.	LT	5	

13.8. RESULTADO DE EVALUACIÓN DEL CLIENTE

Artefacto	Herramienta	Resp.	Calif. (1-5)	Observación
GESTIÓN DE PROYECTOS				
Resultado de verificación	Product Backlog	CL	3	Podría haber una herramienta para mejorar el proceso.
Solicitud de cambio	e-mail	CL	2	Para usuarios que no conocen de TI se hace complejo.

13.9. MATRIZ DE CONSISTENCIA

Planteamiento del Problema	Objetivos	Hipótesis	Variables e indicadores	Muestra	Diseño	Instrumento
<p>Pregunta General</p> <ul style="list-style-type: none"> ¿Cuáles son las herramientas de ingeniería de software producidas por las empresas líderes del mercado que cubran los procesos de la ISO/IEC 29110 en su implementación, atendiendo a criterios de costo y utilización? 	<p>Objetivo General</p> <ul style="list-style-type: none"> Determinar una combinación de herramientas de ingeniería de software, producidas por las empresas líderes del mercado, que cubran la implementación de los procesos de la ISO/IEC 29110, atendiendo a criterios de costo y utilización para mejorar la eficiencia en los procesos de desarrollo de software de las PYMES. Caso de estudio en Lead Working Partner SAC. 	<p>Hipótesis General</p> <ul style="list-style-type: none"> La implementación de las herramientas de ingeniería de Software, a seleccionar permitirá cumplir los procesos contemplados en la ISO/IEC 29110, atendiendo los criterios de costo y utilización para el desarrollo de software en una PYME. 	<p>Variable Independiente (x): Herramientas de Ingeniería de Software.</p> <p>(X1) Utilización:</p> <ul style="list-style-type: none"> Cantidad de documentación disponible en internet. Tendencias de uso de la herramienta. <p>(X2) Costo:</p> <ul style="list-style-type: none"> Costo de Adquisición (licencia). Costo de Mantenimiento (licencia). <p>Variable Dependiente (y): Procesos de la ISO/IEC 29110.</p> <p>(Y1) Procesos de Gestión:</p> <ul style="list-style-type: none"> Número de procesos involucrados. Nivel de satisfacción del responsable del proceso. <p>(Y2) Procesos de Ingeniería:</p> <ul style="list-style-type: none"> Número de procesos involucrados. Nivel de satisfacción del responsable del proceso. 	<p>Para la presente tesis, no se aplica muestreo porque es un caso de estudio de una sola empresa, considerando a la empresa Lead Working Partner SAC como población total.</p>	<p>Tipo de Investigación: Investigación Tecnológica.</p> <p>Nivel de Investigación: Investigación Aplicada.</p> <p>Diseño: Descriptivo Simple.</p> <p>Diagrama: M → O Donde:</p> <p>M: Empresa O: Herramientas de ingeniería Software seleccionadas para cubrir los procesos de implementación de la ISO/IEC 29110.</p>	<ul style="list-style-type: none"> Lectura crítica. Cuestionario.