



**UNIVERSIDAD POLITÉCNICA SALESIANA**

**SEDE GUAYAQUIL**

**CARRERA: INGENIERÍA DE SISTEMAS**

**Tesis previa a la obtención del título de: INGENIERO DE SISTEMAS**

**TEMA:**

**DESARROLLO DEL MÓDULO PARA ADMINISTRACIÓN DE  
PROYECTOS UTILIZANDO LA PLATAFORMA LIBRE ODOO,  
APLICANDO EL MÉTODO CADENA CRÍTICA, BASADO EN LA TEORÍA  
DE RESTRICCIONES DEL DR ELIYAHU GOLDRATT**

**AUTORES:**

**MERCEDES MOGROVEJO BAQUE**

**CHRISTOPHER ORMAZA ACOSTA**

**DIRECTORA:**

**ING. VANESSA JURADO**

**GUAYAQUIL, MARZO DEL 2015**

**DECLARATORIA DE RESPONSABILIDAD Y AUTORIZACIÓN DE USO  
DEL TRABAJO DE GRADO**

Nosotros Mercedes Leonor Mogrovejo Baque y Christopher Emilio Ormaza Acosta autorizamos a la Universidad Politécnica Salesiana la publicación total o parcial de este trabajo de grado y su reproducción sin fines de lucro.

Además declaramos que los conceptos y análisis desarrollados y las conclusiones del presente trabajo son de exclusiva responsabilidad de los autores.

-----

-----

-

Mercedes Mogrovejo Baque

Christopher Ormaza Acosta

CC: 0916899917

CC: 0915068258

## **DEDICATORIA**

Esta tesis va dedicada primero a Dios, ya que gracias a su infinito amor y misericordia me ha permitido llegar a terminar este largo camino hasta llegar a la titulación profesional. A mi madre y abuela que con su apoyo y cuidados constantes de mi salud, bienestar, sus consejos y constancia me han ayudado a perseverar a pesar de las dificultades, y sobre todo a Mercedes mi compañera, amiga, futura esposa la cual me ha acompañado desde el inicio de esta carrera, gracias a su amor incondicional y empuje, no abandoné los estudios en los momentos más difíciles.

**Christopher Ormaza**

## **DEDICATORIA**

Dedico este esfuerzo a Dios, porque gracias a él he podido alcanzar una de las metas más importantes de mi vida.

También dedico este proyecto a mi madre, hermana y demás familiares que me han brindado su amor y apoyo en cada una de las etapas de mi vida profesional.

Además dedico esta tesis a mi mejor amigo, que me ha acompañado en gran de parte de mi carrera, quién me ha brindado su tiempo, paciencia y apoyo en este caminar de aprendizaje, además por la confianza que me ha brindado ayudándome en mi desenvolvimiento profesional.

Finalmente este proyecto ha sido el resultado del esfuerzo de mis docentes Ingenieros Nicolás Pulgar, Joe Llerena, Lily Santos, Miguel Chang, Erwin Chiluzza, Ricardo Naranjo, Miguel Quiroz, Nelson Mora, Joffre León, Marjorie Tingo, Raúl Tingo, Johnny Jiménez, Jorge Yaguno, David Vásquez, Roberto Rangel, quienes con su trabajo extraordinario, paciencia, dedicación, conocimientos y apoyo, han sentado bases importantes en mi carrera profesional y laboral.

**Mercedes Mogrovejo**

## **AGRADECIMIENTO**

Primero a Dios, por haberme encontrado en el camino y restituirme, darme la fuerza y valor para terminar este camino que empezó hace ya varios años, a mi familia por su apoyo, a Mercedes por su paciencia conmigo, y a mis Docentes que se encargaron de dejar su experiencia plasmada en cada clase, taller, proyecto, lección, por su esfuerzo por educarme y aconsejarme como un profesional calificado no solo técnicamente, sino en el ámbito personal y social, gracias a todos ellos por su participación en este camino, hay algo de cada uno de ellos en estas páginas que reflejan el esfuerzo, paciencia y cariño de cada una de estas personas que han participado en mi vida.

**Christopher Ormaza**

## **AGRADECIMIENTO**

Agradecida con Dios que me ha bendecido con la vida, con la familia y con todas personas que me han encaminado y me han permitido concluir con una etapa importante de mi vida.

Agradezco a mi madre por su amor incondicional, por sus desvelos y compañía, que me ha impulsado en la perseverancia para concluir con el esfuerzo de 5 años de estudios. Adicionalmente agradezco a mi hermana, quién desde el comienzo de mi carrera confió y brindó su apoyo para que me prepare para un mejor futuro.

Agradezco a mi compañero, amigo y futuro esposo, quién con su disposición, apoyo y conocimiento me ha acompañado en este caminar de aprendizaje, siempre brindándome su amor y compañía en las dificultades y complejidad en las diferentes áreas.

Además agradezco a mis compañeras y compañeros de estudio, quienes con su apoyo y compañía, han sido solidarios en los momentos más difíciles de mi carrera.

También hago extensivo mi agradecimiento muy especialmente a todos los docentes quiénes han dado y entregado su tiempo, esfuerzo y conocimiento, para que cada uno de sus alumnos seamos profesionales calificados y con valores, ya que gracias a su experiencia compartida, hemos podido enriquecer nuestros conocimientos, dando frutos en nuestra carrera profesional y laboral, brindando nuestras habilidades y aptitudes al servicio de la sociedad.

**Mercedes Mogrovejo**

## TABLA DE CONTENIDO

Introducción .....	1
CAPÍTULO I.....	3
1 Planteamiento del problema.....	3
1.1 Enunciado del problema.....	3
1.1.1 Factores Intermedios: .....	3
1.1.2 Factores Inmediatos:.....	5
1.2 Formulación del Problema:.....	5
1.3 Sistematización del problema de investigación .....	5
1.4 Objetivos: .....	6
1.4.1 Objetivo General .....	6
1.4.2 Objetivos Específicos .....	6
1.5 Justificación.....	6
1.6 Importancia .....	7
1.7 Necesidad .....	7
1.8 Beneficios que aporta.....	8
1.9 Beneficiarios .....	8
CAPÍTULO II .....	9
2 Marco Teórico.....	9
2.1 Antecedentes .....	9
2.1.1 Proyecto.....	9
2.1.2 Administración de Proyectos.....	9
2.1.3 Teoría de Restricciones .....	12
2.1.4 Proceso de mejoramiento continuo: .....	13
2.1.5 Cadena Crítica .....	14
2.1.6 Odoos .....	16
2.1.7 Oerplib.....	20

2.1.8	Postgresql .....	20
2.1.9	Python.....	23
2.1.10	Xml.....	25
2.1.11	Xpath .....	30
2.1.12	Java.....	30
2.1.13	JavaScript .....	30
2.2	Marco legal.....	32
2.2.1	Licencia Pública General Affero de GNU (AGPL) versión 3 (#AGPL) (#AGPLv3.0).....	33
2.2.2	Licencia BSD Modificada (#ModifiedBSD).....	33
2.2.3	Licencia Creative Commons Reconocimiento 4.0 (también conocida como CC BY) (#ccby) .....	34
2.2.4	Licencia de Expat (#Expat) .....	34
2.2.5	Licencia Apache, versión 2.0 (#apache2).....	34
2.2.6	Python Software Foundation License.....	34
CAPÍTULO III.....		36
3	Análisis del Sistema .....	36
3.1	Requerimientos Funcionales:.....	36
3.1.1	Proceso de importación de archivo CSV:.....	36
3.1.2	Creación de un proyecto.....	36
3.1.3	Descripción: Creación de un proyecto .....	36
3.1.4	Creación de tareas de un proyecto.....	37
3.1.5	Cálculo de Ruta Crítica .....	37
3.1.6	Cálculo de Tareas Alimentadoras.....	38
3.1.7	Restricción de Multitareas (Sobre asignaciones) .....	38
3.1.8	Reducción de Tiempo de Tareas del Proyecto .....	38
3.1.9	Cálculo de Amortiguador de Tareas Alimentadoras .....	39
3.1.10	Cálculo de Amortiguador del Proyecto .....	39

3.1.11	Reporte de Penetración del Proyecto.....	40
3.1.12	Reporte de Tareas Urgentes.....	40
3.2	Requerimientos No Funcionales:.....	40
3.2.1	Sistema Operativo del Servidor.....	41
3.2.2	Base de datos Postgresql .....	41
3.2.3	Navegador para Usuarios .....	41
3.3	Casos de Uso .....	42
3.3.1	Importación de Proyectos .....	43
3.3.2	Creación de un recurso .....	44
3.3.3	Creación de Proyecto.....	45
3.3.4	Creación de Tareas .....	45
3.3.5	Cálculo de Ruta Crítica .....	46
3.3.6	Cálculo y Asignación de Cadena Crítica.....	47
3.3.7	Aprobación del Proyecto .....	48
3.3.8	Registro de Avance en Tareas .....	48
3.3.9	Reporte de Penetración de Amortiguadores .....	49
3.3.10	Cierre de Proyecto .....	49
3.3.11	Creación de calendario .....	50
3.4	Definición de Roles en los Módulos:.....	50
3.4.1	Administrador del Proyecto:.....	50
3.4.2	Recurso del Proyecto:.....	51
3.4.3	Definición de operaciones de base de datos permitidas .....	52
CAPÍTULO IV.....		54
4	Diseño del Sistema.....	54
4.1	Diseño de la arquitectura del sistema.....	54
4.1.1	Diseño Arquitectónico.....	54
4.1.2	Módulos del Sistema .....	57

4.2	Diseño de las Interfaces .....	59
4.3	Diagrama de Clases del Sistema .....	65
4.4	Modelo Lógico de la Base de Datos .....	76
CAPÍTULO V .....		77
5	Implementación y Pruebas .....	77
5.1	Capas del Sistema y Comunicación entre Capas .....	77
5.2	Plan de Pruebas .....	79
5.3	Resultado de pruebas y métricas tomadas.....	80
CAPÍTULO VI.....		84
6	Conclusiones y recomendaciones .....	84
6.1	Conclusión.....	84
6.2	Recomendación.....	85
Bibliografía .....		87

## ÍNDICE DE TABLAS

Tabla 1: Características Técnicas Postgresql .....	23
Tabla 2: Requerimiento Funcional Proceso de Importación archivos CSV .....	36
Tabla 3: Requerimiento Funcional Creación de un proyecto.....	36
Tabla 4: Requerimiento Funcional Creación de tareas .....	37
Tabla 5: Requerimiento Funcional Cálculo de ruta crítica .....	37
Tabla 6: Requerimiento Funcional Cálculo de tareas alimentadoras.....	38
Tabla 7: Requerimiento Funcional Restricción de multitareas.....	38
Tabla 8: Requerimiento Funcional Reducción de tiempo de tareas.....	38
Tabla 9: Requerimiento Funcional Cálculo de amortiguador de tareas alimentadoras .....	39
Tabla 10: Requerimiento Funcional Cálculo de amortiguador del proyecto .....	39
Tabla 11: Requerimiento Funcional Reporte de penetración del proyecto.....	40
Tabla 12: Requerimiento Funcional Reporte de tareas urgentes .....	40
Tabla 13: Requerimiento no funcional Sistema operativo del servidor.....	41
Tabla 14: Requerimiento no funcional base de datos postgresql.....	41
Tabla 15: Requerimiento no funcional Navegador para usuarios.....	41
Tabla 16: Caso de Uso Importación de Proyecto.....	43
Tabla 17 Caso de Uso Creación de Recurso .....	44
Tabla 18: Caso de Uso Creación de Proyecto.....	45
Tabla 19: Caso de uso creación de tarea .....	45
Tabla 20: Caso de Uso Cálculo de .....	46
Tabla 21: Caso de Uso Cálculo y Asignación de Cadena Crítica.....	47
Tabla 22: Caso de Uso Aprobación del Proyecto .....	48
Tabla 23: Caso de uso Registro de Avance en Tareas .....	48
Tabla 24: Caso de uso Reporte de Penetración de Amortiguadores .....	49
Tabla 25: Caso de Uso Cierre de Proyecto .....	49
Tabla 26: Caso de Uso: Creación de Calendario .....	50
Tabla 27: Lista de Campos creados automáticamente por el framework de Odoo....	54
Tabla 28: Campos solicitados para importación de archivo csv .....	58

## ÍNDICE DE ILUSTRACIONES

Ilustración 1: Tareas en Ejecución Simple comparación con Tareas en Ejecución Multitareas.....	4
Ilustración 2: Síndrome del Estudiante .....	4
Ilustración 3 : Desperdicio de tiempo con ley de parkinson .....	5
Ilustración 4: Ejemplo de red Pert.....	11
Ilustración 5: Ejemplo de Gráfica Gantt .....	12
Ilustración 6: Registro de Proyecto en Plataforma Odoo (OpenERP).....	19
Ilustración 7: Registro de Tarea de Proyecto Plataforma Odoo (OpenERP).....	19
Ilustración 8: Componentes Postgresql.....	21
Ilustración 9: Diagrama de Casos de Uso .....	42
Ilustración 10: Modelo conceptual de la aplicación.....	54
Ilustración 11: Pantalla de Acceso Odoo .....	59
Ilustración 12: Pantalla general de búsqueda del sistema .....	60
Ilustración 13: Asistente de Importación de Archivo CSV .....	61
Ilustración 14: Asistente de Importación de Archivos de Project MPP, y XML.....	61
Ilustración 15: Asistente de importación de archivo, resumen de importación .....	62
Ilustración 16: Pantalla de creación de proyectos .....	62
Ilustración 17: Vista de Gantt de Tareas .....	63
Ilustración 18: Diagrama de Clase project.category .....	65
Ilustración 19: Diagrama de Clase project.config.settings.....	66
Ilustración 20: Diagrama de Clase project.project.....	67
Ilustración 21: Diagrama de Clase project.resource .....	68
Ilustración 22: Diagrama de Clase project.resource.assignment .....	69
Ilustración 23: Diagrama de Clase project.resource.calendar.holiday .....	69
Ilustración 24: Diagrama de Clase project.task.....	70
Ilustración 25: Diagrama de Clase project.task.delegate .....	71
Ilustración 26: Diagrama de Clase project.task.history .....	71
Ilustración 27: Diagrama de Clase project.task.history.cumulative.....	72
Ilustración 28: Diagrama de Clase project.task.type.....	73
Ilustración 29: Diagrama de Clase project.task.work .....	73
Ilustración 30: Diagrama de Clase report.project.task.user .....	74
Ilustración 31: Diagrama de Clase wizard.import.project.line .....	75

Ilustración 32: Diagrama de Clase wizard.import.project.user.....	75
Ilustración 33: Diagrama Entidad Relación .....	76
Ilustración 34: Arquitectura General del Sistema .....	78
Ilustración 35: Modelo Conceptual de Comunicación entre Odoos y módulo Java....	79

## ÍNDICE DE ANEXOS

Anexo 1: Matriz de Causa y Efecto .....	90
Anexo 2: Diccionario de Datos .....	92
Anexo 3: Formato de Pruebas Unitarias .....	100
Anexo 4: Formato de Pruebas de Integración .....	101

## RESUMEN

La teoría de restricciones desarrollada en los años 80 por el Dr. Eliyahu M. Goldratt, en su evolución como filosofía para la administración de diferentes áreas de las compañías se encontró con la aplicación de sus preceptos en la administración de proyectos, lo cual permitió crear la Cadena Crítica, que pretende solucionar puntualmente 3 problemas, el síndrome del estudiante, la ley de parkinson, y el cambio constante entre multitareas, que con las filosofías tradicionales de administración de proyectos, en un impulso por permitir a los administradores de proyecto tener acceso a una herramienta de administración de proyectos que posea los aspecto de Cadena Crítica necesarios para su correcta implementación, para este proyecto se escogió una herramienta de código abierto de amplia acogida en el ámbito empresarial, como lo es en la actualidad Odoos, con más de 2.000.000 de usuarios activos, 2000 desarrolladores a nivel mundial.

El módulo de administración de proyectos que posee Odoos en su versión 8.0 sirve de base para realizar la implementación de la cadena crítica, con el uso de varias herramientas de código abierto, fue posible realizar una implementación simple, siguiendo los lineamiento de la propia teoría de restricciones, que promueve la simplicidad inherente en los problemas existente en los entornos empresariales, y para este caso en la rama de administración de proyectos.

Las herramientas comerciales existentes al momento de esta investigación poseen la limitante de ser aisladas de los software de gestión, Odoos integra su proceso de administración de proyectos a los demás procesos empresariales, como la facturación, recursos humanos, haciendo que la implementación de la teoría en este caso no quede limitada a su uso por parte de los administradores de proyectos, sino que pueda usarse de manera integral dentro de una compañía por los departamentos relacionados a esta actividad.

## **ABSTRACT**

The theory of constraints developed in the 80s by Dr. Eliyahu M. Goldratt, in its evolution as a philosophy for managing different areas of the companies met the provisions thereof in project management, which helped create the Critical Chain, which aims to promptly solve three problems, student's syndrome, parkinson's law, and constant change between multitasking, than traditional project management philosophies, on an impulse to allow project managers access a tool for project management possessing the features of Critical Chain necessary for a proper implementation, to project we choose an open source tool with worldwide acceptance in business scope, as it is currently Odoo, with more than 2,000. 000 active users, 2000 developers worldwide.

The project management module that owns Odoo in version 8.0 is the base for the implementation of the critical chain, using several open source tools, it was possible to make a simple implementation, following the outline of the theory of constraints, which promotes the inherent simplicity of the existing problems in enterprise environments, and in this case in the field of project management.

Existing at the time of this research commercial tools have the limitation of being isolated from software management, Odoo integrates its project management process to other business processes such as billing, human resources, making implementation of theory in this case is limited to use by project managers, but can be used holistically within a company by the departments related to this activity.

## INTRODUCCIÓN

Para reconocer el problema con respecto a la administración de proyectos es necesario comprender cuál son los objetivos generales de todo proyecto los cuales se puede listar:

- ❖ Cumplir el presupuesto económico
- ❖ Cumplir en el tiempo acordado o cronograma
- ❖ Cumplir con las especificaciones o calidad esperada

Durante varios años se han realizado estudios y encuestas sobre el éxito que tienen los proyectos ejecutados en todos los ámbitos de la ingeniería, dentro de estos estudios se destaca los que refieren a la ingeniería de sistema, y la implementación de tecnologías de información en el ámbito empresarial, se puede destacar los siguientes estudios:

McKinsey & Company, junto con la Universidad de Oxford, realizó un estudio publicado en Octubre de 2012, enfocado a Grandes Proyectos de TI, de acuerdo con esta investigación, de los 5400 proyecto de TI consultados, 45% han superado su presupuesto, 7% no se han cumplido dentro del cronograma, 56% no cumplen con la calidad esperada. (Bloch, Blumberg, & Laartz, 2012).

International Data Corporation realizó un estudio en el 2009 donde se determina que el 25% de los proyectos de TI fracasan completamente, del 20% al 25% de estos no tienen retorno sobre la inversión realizada, y el 50 % de estos proyectos requiere reelaboración. (Gulla, 2012).

La compañía Gartner realizó un estudio en el año 2011 a 845 proyectos de TI realizados con la ayuda de un proveedor externo para las compañías, resalta la estadística de 42.5% de los proyectos no entregan los beneficios esperados, 42% no fueron entregados a tiempo, y 44% se entregaron por encima del presupuesto inicial. (Tan, 2011).

Gracias a estos estudios relacionados con la estadística de fracaso de los proyectos se concluye que en la actualidad existe una tendencia marcada hacia los proyectos fracasados.

De aquí se desprende la preocupación por reducir esta tendencia con la implementación de nuevas técnicas y métodos, para asegurar el término exitoso de un proyecto (Barros, 2010).

En el año 2001 el Dr. Eliyahu Goldratt publicó el libro llamado Cadena Crítica sobre la cual se plantean varios problemas que no son evaluados correctamente en la administración, presenta a través de su teoría de restricciones un análisis sistemático diferente a la administración tradicional, sin embargo con resultados notorios de mejora. (Pérez de Eulate & Oyarbide Zubillaga, 2005).

A continuación se presenta un resumen de lo contenido en cada uno de los capítulos de este documento:

**Capítulo I:** Se realiza una explicación de la problemática que poseen actualmente los proyectos, la forma en que se plantea la solución a este problema cadena crítica y como con el apoyo tecnológico puede ayudar a los implementadores a utilizar la teoría.

**Capítulo II:** Se muestra el antecedente científico detrás de la propuesta planteada, se explica al respecto de las herramientas utilizadas en el proyecto, y los aspectos legales necesarios a tener en cuenta para los posteriores desarrollos derivados del mismo.

**Capítulo III:** En general se habla de las especificaciones que se consideran necesarias para cumplir con la entrega de información que la teoría cadena crítica solicita para la toma de decisiones.

**Capítulo IV:** Se establece el diseño técnico sobre el cual se estructura el proyecto, los módulos desarrollados, la interacción entre las diferentes tecnologías aplicadas y las interfaces usadas para el proyecto.

**Capítulo V:** Se contiene de manera general el resultado de las pruebas en la implementación del desarrollo, y las métricas obtenidas del mismo.

**Capítulo VI:** Las conclusiones derivadas del desarrollo de este proyecto, así como las recomendaciones necesarias para su uso o extensión posterior.

# CAPÍTULO I

## PLANTEAMIENTO DEL PROBLEMA

### 1.1 Enunciado del problema

Al método cadena crítica para la administración de proyecto, le faltan herramientas informáticas accesibles para su implementación, la adopción de herramientas de código abierto para administración de proyectos, posee poca aceptación debido a la difusión y uso de herramientas comerciales como Microsoft Project.

#### 1.1.1 Factores Intermedios:

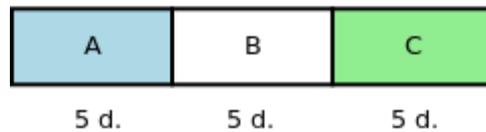
En la actualidad en la gestión de proyectos se administra de manera tradicional con varios métodos(CPM, PERT, GANTT, etc), su enfoque consiste en prestar atención con las tareas que pertenecen a la ruta crítica para determinar el tiempo total del proyecto y subordinar las decisiones a esta ruta, en la época en la que este concepto fue creado revolucionó la forma en la que se administraban proyectos, sin embargo en la actualidad con la aplicación de la misma, se han denotado puntos que no son tratados por el esquema tradicional que pueden repercutir negativamente en el cumplimiento de los 3 objetivos básicos de todos los proyectos:

- ❖ Cumplir en el tiempo estipulado
- ❖ Cumplir con el presupuesto económico acordado
- ❖ Cumplir con la calidad esperada del producto del proyecto

El método de Cadena Crítica plantea la solución a varios fenómenos que se presentan en la ejecución de los proyectos, estos fenómenos dan paso a retrasos en los proyectos al no tenerlos presentes en el enfoque del término del proyecto:

**Multitareas.-** El personal asignado a un proyecto debido a que está cambiando constantemente de enfoques en las tareas que tiene asignada, generalmente superpone tiempos de ejecución, lo cual produce que la personal para 'llegar al punto' en el que está 100% ejecutando la tarea se diluye entre tanto cambio, alargando todas las tareas que entran en este esquema de ejecución, en la ilustración # 1 se puede ver el efecto que tiene en el tiempo este constante cambio de contexto.

### Ejecución Simple



### Ejecución en Multitareas

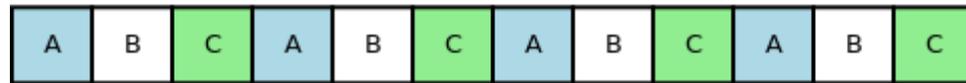


Ilustración 1: Tareas en Ejecución Simple comparación con Tareas en Ejecución Multitareas

Elaborado por: Los Autores

**Síndrome del Estudiante.-** La persona con una tarea asignada evalúa el tiempo efectivo o de toque que tiene que disponer la tarea para realizarse, por lo cual supone que la holgura que posee el tiempo para la ejecución de la misma permite comenzar lo más cerca del final de su tiempo límite, quiere decir por ejemplo si se tiene 7 días para realizar una tarea, pero se conoce o se estima que la tarea demora efectivamente 3 días, el ejecutor lo comenzara desde el día 3 o 4, permitiendo que si la tarea presenta alguna demora de cualquier tipo, se incumpla con cualquiera o todos los objetivos principales del proyecto, en la ilustración # 2 es posible visualizar el impacto que tiene este tiempo desperdiciado por este síndrome.

### Estimación Original

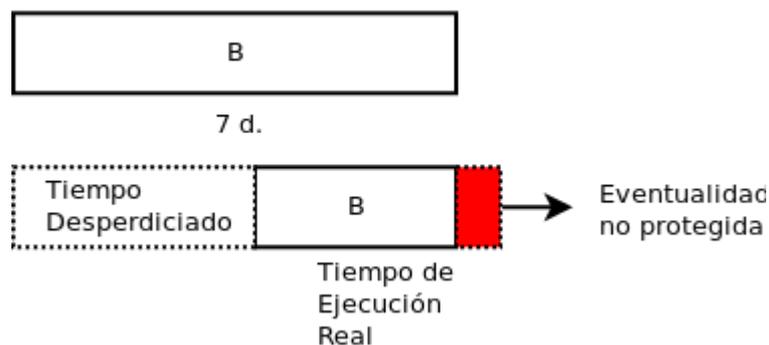


Ilustración 2: Síndrome del Estudiante

Elaborado por: Los Autores

**Ley de Parkinson.-** Este se da por un tema más subjetivo en la estimación del tiempo de ejecución de la tarea que no siempre es evidenciable, pero que sucede; volviendo al ejemplo anterior, si la tarea estaba estimada con un tiempo de 7 días, sin embargo en la ejecución se logra hacerlo en 4 días, en teoría debería ganarse 3 días

más para avanzar con otra tarea, pero el personal no reporta su término por varios motivos, que el más común puede ser: que la persona que estimó la tarea pierda credibilidad, dado que la próxima vez que estime tanta holgura probablemente le asignen menos tiempo del que estimo la primera vez, en la ilustración # 3 se evidencia el tiempo que no es notificado como ganancia del proyecto debido al efecto antes mencionado.

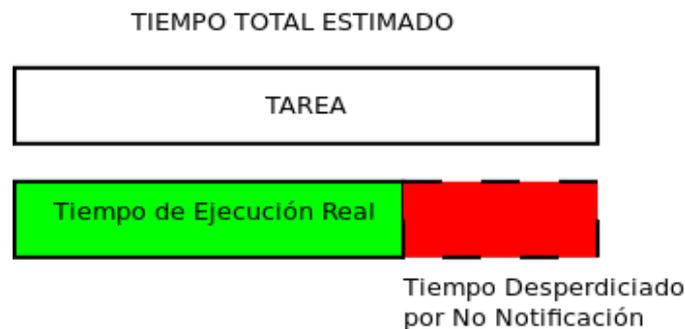


Ilustración 3 : Desperdicio de tiempo con ley de parkinson

Elaborado por: Los Autores

### **1.1.2 Factores Inmediatos:**

Actualmente sin las herramientas no es posible la aplicación del método cadena crítica desde la fase de planificación del proyecto, por lo tanto no se puede tener información del estado del proyecto real y de su avance, debido a la falta de información de los datos del tiempo de amortiguación del proyecto o también llamado la penetración de amortiguadores que permita identificar cuáles son las tareas que necesitan atención inmediata, y que ponen en riesgo el cumplimiento de los objetivos del proyecto.

### **1.2 Formulación del Problema:**

¿Por qué en los proyectos no es viable aplicar cadena crítica para cumplir con los objetivos de terminar a tiempo, dentro del presupuesto esperado y con la calidad esperada?

### **1.3 Sistematización del problema de investigación**

¿Por qué no es posible determinar la cadena crítica en la fase de planificación del proyecto?

¿Por qué no es posible tener los reportes necesarios para el seguimiento del avance del proyecto que propone la cadena crítica para la medición?

¿Por qué no es posible identificar cuáles son las tareas que tienen un riesgo alto de retrasar todo el proyecto en general?

¿Por qué poseen poca aceptación las herramientas de código abierto para la administración de proyectos?

## **1.4 Objetivos:**

### **1.4.1 Objetivo General**

- ❖ Diseñar módulo para gestión de proyectos a través del método de cadena crítica en la plataforma de software libre Odoo, para facilitar la administración de los proyectos a través de este método de manera eficiente.

### **1.4.2 Objetivos Específicos**

- ❖ Desarrollar mecanismos para cálculo y sugerencia de amortiguadores para las distintas categorías que identifica la teoría, amortiguador de proyecto, amortiguadores de alimentación, y amortiguadores de recursos.
- ❖ Desarrollar los reportes necesarios para dar seguimiento a la penetración del amortiguador general del proyecto, y de las tareas específicas que posean amortiguadores de alimentación o de recursos.
- ❖ Adicionar los esquemas de semáforos propuestos por las teorías en los reportes de penetración de amortiguadores, y gráficas Gantt del proyecto.
- ❖ Construir un mecanismo para importar proyectos desde Microsoft Project (Archivos XML) y archivos separados por comas (.csv) en el gestor de proyectos de Odoo, que permitirá migrar proyectos actualmente gestionados a través de estas herramientas.

## **1.5 Justificación**

**Eliyahu M. Goldratt** publicó en 1997 su obra más conocida, “**critical chain**“, donde se expone en qué consiste este método, basado en la teoría de restricciones. Su

conocimiento en la materia le permitió desarrollar los algoritmos necesarios para garantizar proyectos finalizados en la mitad de tiempo y con una economía de, en ocasiones, un 50% menos de recursos utilizados.

El método de cadena crítica para administración de proyecto posee poca difusión y aceptación ya que sus esquemas son poco tradicionales y en ciertas ocasiones contradictorios con los esquemas generalmente aceptados de gestión de proyectos.

No existen plataformas que permitan el manejo de este método de manera adecuada, por la falta del apoyo tecnológico y su difusión es lenta, por esta razón el desarrollo de este módulo pretende acelerar su aceptación y difusión gracias al desarrollo implementado en una plataforma libre del cual la comunidad podrá nutrir con todas las experiencias acumuladas en el uso del mismo.

Está dirigido principalmente para administradores de proyectos que ya conocen los conceptos de cadena crítica y buscan la automatización del proceso de gestión de proyectos a través de un sistema informático.

## **1.6 Importancia**

Debido a la estadística desfavorable en proyectos exitosos cumpliendo los 3 requisitos que son: el tiempo, el presupuesto y la calidad, se hace necesario que los administradores de proyectos a más de aplicar metodologías efectivas que les sirvan de herramientas para tomar decisiones, cadena crítica se presenta como una alternativa en la administración de proyecto para lograr este cometido; de ahí que tenga el apoyo tecnológico necesario para su correcta y eficiente aplicación, nace este proyecto que apoyado con herramientas de código abierto, sea la base para la aplicación correcta de la teoría.

## **1.7 Necesidad**

En la experiencia recopilada buscando herramientas para la administración de proyectos usando la metodología de cadena crítica, se encontró un limitado número de alternativas y de opciones accesibles por el público en general, sobre todo enfocada a entornos windows, o módulos agregados a microsoft project lo cual su adquisición está sujeta a la inversión en licencias. Por esto es necesario agregar una alternativa de código abierto de tal manera que como otras herramientas de este tipo,

permita la aplicación del método y por consiguiente da apertura para la mejora por parte de las personas que estén dispuestas a desarrollar sobre este proyecto de manera posterior.

### **1.8 Beneficios que aporta**

- ❖ Basado en una herramienta de software libre, líder en el mercado de aplicaciones de negocios del segmento ERP, hace que sea parte de un paquete consolidado de aplicaciones, entre ellas la gestión de proyectos.
- ❖ Plataforma aplicable independiente del sistema operativo, ya que se desarrolla sobre un ambiente web, lo que lo hace portable y de fácil acceso.
- ❖ Con la ventaja que proporciona el desarrollo de código abierto, permite que el módulo sea escalable, dando opción a la continua mejora de su funcionalidad.

### **1.9 Beneficiarios**

Los administradores de proyectos que actualmente ya usan la metodología de cadena crítica encontrarán en este proyecto una alternativa accesible para implementar, que permitirá su fácil adopción.

Los estudiantes de administración de proyectos de pregrado o postgrado que quieran conocer más acerca de lo planteado por la metodología cadena crítica, encontrarán en esta herramienta una forma de experimentar con el conocimiento implementado en este proyecto.

## **CAPÍTULO II**

### **MARCO TEÓRICO**

#### **2.1 Antecedentes**

##### **2.1.1 Proyecto**

Según (Kerzner, 2013) un proyecto puede ser considerado como un conjunto de tareas y actividades que:

- ❖ Tiene un objetivo específico que debe ser cumplido dentro de ciertas especificaciones
- ❖ Tiene fechas definidas de inicio y fin
- ❖ Posee recursos económicos limitados(Si el mismo aplica)
- ❖ Consume recursos humanos y no humanos
- ❖ Son multifuncionales

##### **2.1.2 Administración de Proyectos**

La administración de proyectos según (Project Management Institute, 2000) es la aplicación de conocimientos, habilidades, herramientas y técnicas para lograr que las actividades del proyecto alcancen los requerimientos del proyecto, este proceso se desarrolla a través de 5 grupos de actividades definidas:

- ❖ Inicio
- ❖ Planeación
- ❖ Ejecución
- ❖ Monitoreo y Control
  - Alcance,
  - Calidad,
  - Cronograma,
  - Recursos,
  - Presupuesto,
  - Riesgos

Las metodologías en las que se apoya fuertemente la administración de proyectos son:

### 2.1.2.1 El Método de la Ruta Crítica (CPM)

Desarrollado en los años de 1950 por las fuerzas armadas estadounidenses.

Originalmente, el método consideraba la dependencia lógica entre los elementos terminales. Desde entonces, ha sido expandido para permitir la inclusión de recursos en cada actividad, a través de un proceso llamado asignación de recursos basados en actividades, o nivelación de recursos.

Es un algoritmo matemático usado para planificar un conjunto de actividades de un proyecto. Cualquier proyecto con tareas interdependientes puede aplicar este método.

La técnica esencial para usar CPM, es la construcción de un modelo del proyecto que incluye lo siguiente:

- ❖ Una lista de las actividades requeridas para completar el proyecto (Conocida como Estructura de Desglose de Trabajo (Work Breakdown Structure o WBS)).
- ❖ El tiempo que tomara para terminar cada actividad.
- ❖ La dependencia entre actividades.
- ❖ CPM calcula:
  - La ruta más larga de las actividades planificadas para el término del proyecto
  - La fecha más próxima y la más tardía que puede empezar o terminar una tarea sin que esto alargue el proyecto

Según (Santiago & Desirae, 2009) la ruta crítica determina las actividades “críticas” (que pertenecen a esta ruta más larga). Prioriza las actividades para una administración efectiva y acorta de la ruta crítica planeada de un proyecto por medio de:

- ❖ Reducción del tiempo de las actividades críticas
- ❖ “Seguimiento Rápido” (Realizando más actividades en paralelo)
- ❖ “Rompiendo la Cadena Crítica” (Acortando la duración de la ruta crítica, con el agregado de más recursos)

### 2.1.2.2 La Técnica de Evaluación y Revisión de Programas (PERT).

Fue creado en Estados Unidos en 1954 para la construcción del submarino POLARIS. Es un método que consiste en ordenar, bajo la forma de red, varias tareas que, gracias a su dependencia y cronología, concurren todas ellas a la obtención de un producto acabado.

Para aplicar el método se debe seguir los siguientes 4 pasos:

- ❖ Hacer una lista de actividades o tareas
- ❖ Hacer una “tabla de precedencias”
- ❖ Dibujar el grafo
- ❖ Calcular las duraciones

La red PERT(A veces también denominado gráfico PERT) consta de los siguientes elementos, que en la ilustración # 4 se puede identificar:

- ❖ Tareas, estas son representadas por flechas, a cada una se le asigna un código y una duración
- ❖ Etapas, es decir el inicio y el fin de la tarea, con excepción de las tareas inicial y final todas las tareas tienen una etapa de inicio y de fin, las etapas generalmente están representadas por un círculo, sin embargo también suelen ser representados por rectángulos, cuadrados, óvalos, etc. (Poggioli, 1976)

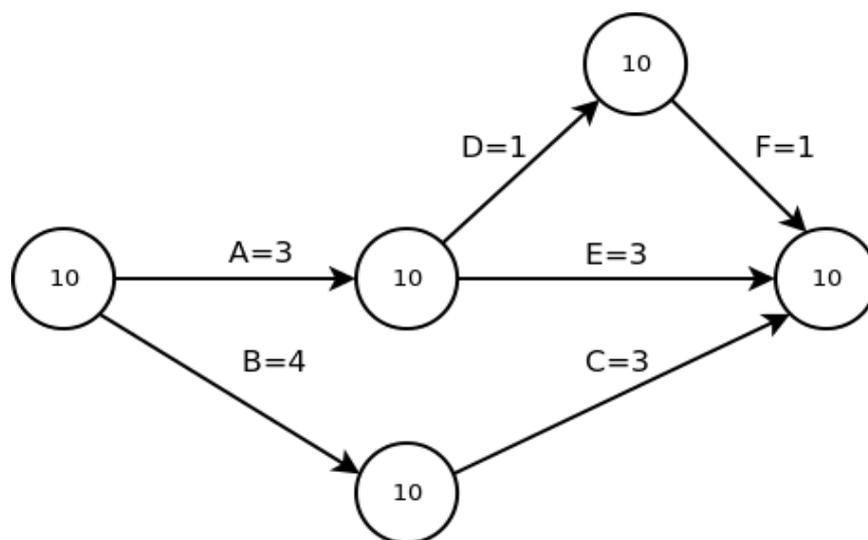


Ilustración 4: Ejemplo de red Pert

Elaborado por: Los Autores

### 2.1.2.3 La Gráfica de Barras o de Gantt.

Este diagrama fue creado por Henry Laurence Gantt entre 1910 y 1915.

Actualmente esta es una herramienta básica para la administración de proyectos, que permite visualizar de manera rápida, las actividades, tareas y fases de un proyecto, tal como se muestra en la ilustración # 5 se puede ver las actividades con su respectiva duración.

Su estructura es de 1 eje horizontal que muestra las actividades dentro de un calendario, y cuanto se van a demorar. (Santoyo González, Tenorio González, & Tenorio González, 2011).

	Tiempo en Meses											
Tareas	1	2	3	4	5	6	7	8	9	10	11	12
A	■											
B		■	■									
C			■	■	■							
D					■	■						
E					■							
F					■	■						
G						■	■					
H							■	■				
I								■				
J								■	■			
K										■	■	

Ilustración 5: Ejemplo de Gráfica Gantt

Elaborado por: Los Autores

### 2.1.3 Teoría de Restricciones

La teoría de las restricciones tiene su origen en programas fundamentados en la programación lineal siendo utilizada inicialmente en el ambiente de fábrica. Fue desarrollada por el físico israelí Eliyahu Goldratt, quien comenzó a analizar problemas de negocios casi que de manera casual, esta teoría fue introducida en 1984 en el libro “La Meta”.

Es una metodología al servicio de la gerencia que permite direccionar la empresa hacia la consecución de resultados de manera lógica y sistemática, contribuyendo a garantizar el principio de continuidad empresarial.

La teoría de las restricciones invita a los administradores de empresas a concentrar sus esfuerzos en las actividades que tienen incidencia directa sobre la eficacia de la empresa como un todo, es decir, sobre los resultados globales. Para que el sistema empresarial funcione adecuadamente las operaciones deben ser estabilizadas, para ello es necesario identificar y alterar las políticas contraproducentes. Entonces, se hace conveniente crear un "patrón o modelo que incluya no apenas conceptos, sino principios orientadores y prescripciones, con sus respectivas herramientas y aplicaciones.

Teoría de las restricciones tiene como fundamento la teoría de sistemas, cuyo punto de partida es la consideración de que los sistemas son teleológicos; es decir, que tienen un objetivo o propósito. La teoría considera la empresa como un sistema constituido con la intencionalidad de conseguir una meta.

La visión sistémica posibilita el análisis de la empresa a partir del estudio de partes menores (subsistemas que se interrelacionan entre sí en el cumplimiento de sus objetivos. De esta manera se puede considerar que "la empresa es un agrupamiento humano jerarquizado que pone en acción medios intelectuales, físicos y financieros, para extraer, transformar, transportar y distribuir riquezas o producir servicios, conforme objetivos definidos por una dirección individual o colegiada, haciendo intervenir, en diversos grados, motivación de beneficio y de utilidad social. De acuerdo con Goldratt, la meta de cualquier sistema industrial, comercial o de servicios es ganar dinero en el presente, como también garantizar su continuidad en el futuro.

La teoría de las restricciones propone el siguiente proceso para auxiliar a los gerentes.

#### **2.1.4 Proceso de mejoramiento continuo:**

En (Aguilera C., 2007) se indica los siguientes pasos para el proceso de mejoramiento continuo:

- ❖ Paso 1: Identificar la(s) restricción(es) del sistema.
- ❖ Paso 2: Explotar la(s) restricción(es) del sistema.
- ❖ Paso 3: Subordinar cualquier otra cosa a la decisión anterior.

- ❖ Paso 4: Elevar la(s) restricción(es) del sistema.
- ❖ Paso 5: Si una restricción es superada, vuelva al paso 1. No deje que la inercia sea la mayor restricción del sistema

### **2.1.5 Cadena Crítica**

El CCPM se basa en la teoría de las limitaciones (o de las restricciones) (TOC - Theory Of Constraints), lo que le distingue de los métodos CPM/PERT u otros. La aplicación de la TOC supone que diversos conceptos que pertenecen al ámbito de la gestión de la producción se han adaptado al ámbito de la gestión de proyectos, como por ejemplo, los conceptos cuello de botella, limitación, y dependencias de los recursos. En el método del camino crítico se consideran los problemas de capacidad de los recursos, pero la aplicación de la teoría de restricciones implica considerar las dependencias entre los recursos. Otro concepto que le aporta diferencias y está relacionado con las variaciones es el buffer. El buffer protege la limitación frente a las variaciones. Por otra parte, basarse en la teoría de restricciones, tiene por consecuencia utilizar un indicador diferente para evaluar el rendimiento del sistema: el throughput. En definitiva, la aplicación de la teoría de restricciones al ámbito de la gestión de proyectos implica la adquisición de algunos conceptos, que fundamentalmente no son nuevos pero que tienen un carácter “novedoso” dado que se utilizan por primera vez en el ámbito de la gestión de proyecto.

Al igual que en el método del camino crítico, el método de la cadena crítica se compone de dos fases: la primera, de planificación y programación del proyecto, implica: listar actividades, establecer la secuencia de actividades, reducir las actividades eliminando márgenes de seguridad (Goldratt, 1997), identificar la cadena crítica, explorar la limitación (Leach, 2005), subordinar las cadenas de actividades a la cadena crítica mediante buffers de alimentación actividades, superar la restricción del sistema, iterar, e insertar buffer de coste (Leach, 2005). La fase de seguimiento y control se apoya los reportes o actualizaciones del estado del proyecto y la gestión de los buffers fundamentalmente. El papel de un buffer es absorber las variaciones de duración de las actividades que lo preceden. Así, el buffer del proyecto absorbe las variaciones en las actividades de la cadena crítica, el buffer de alimentación absorbe las variaciones en las actividades no críticas y el buffer de

recurso, las variaciones en las actividades no críticas de un recurso cuando la actividad siguiente de este recurso es una actividad crítica.

La fase de seguimiento y control comprende: el informe del trabajo realizado, el control de los plazos mediante la gestión de del buffer del proyecto y de los buffers de alimentación, y el control de costes, mediante el método del valor ganado. La Teoría de las limitaciones (TOC) y sus conceptos, anteriormente expuestos, tienen su aplicación en proyecto único, pero el 90% de los proyectos se desarrollan en contextos multiproyecto (Turner, 1993). Aplicar la TOC en un entorno multiproyecto implica considerar la cartera de proyectos o portafolio como un sistema con el objetivo de obtener el mayor rendimiento de este sistema.

Se trata de realizar el mayor número de proyectos en el menor tiempo, optimizando la utilización de los recursos disponibles para el conjunto de proyectos.

Así introduce el concepto del “throughput” entendido como la velocidad a la que el sistema genera ingresos económicos (Goldrat, 1997). En el sistema existe una limitación que es el recurso que condiciona la duración de los proyectos. Es el recurso que interviene en varios proyectos y que en caso de tener un retraso en una actividad de un determinado proyecto, influye generando retraso en los demás proyectos. Esta limitación se denomina “DRUM” y define la cadencia de funcionamiento de todo el sistema.

En el contexto de la gestión multiproyecto, el método de la cadena crítica (CCPM), para la fase de planificación y programación, sigue los siguientes pasos: planificar cada proyecto de la cartera, identificar el recurso que limita el sistema (DRUM), y secuenciar los proyectos a partir del DRUM, insertando los buffers de capacidad (entre proyectos) y los buffers de alimentación del DRUM (entre actividades de un proyecto).

Para la fase de seguimiento y control, el objetivo es controlar la cartera de proyectos. Se utiliza la misma representación para reflejar los estados de cada proyecto, lo que permite obtener una visión global del sistema (empresa) y de sus subsistemas (cada proyecto).

Estas representaciones permiten visualizar cuales son los proyectos que necesitan una atención particular e intentar recuperar partes de los buffers manipulando los proyectos que disponen de suficiente margen. La atención de los directores de proyectos se focaliza en dar prioridad a las actividades de la cadena crítica frente a las demás, a las actividades de los proyectos con mayor penetración en el buffer del proyecto y a las actividades de los proyectos con mayor penetración en los buffers de alimentación. La fase de seguimiento y control comprende: el informe del trabajo realizado, el control de los plazos mediante la gestión del buffer del proyecto y de los buffers de alimentación, y el control de costes, mediante el método del valor ganado.

La teoría de las limitaciones (TOC) y sus conceptos, anteriormente expuestos, tienen su aplicación en proyecto único, pero el 90% de los proyectos se desarrollan en contextos multiproyecto (Turner, 1993). Aplicar TOC en un entorno multiproyecto implica considerar la cartera de proyectos o portafolio como un sistema con el objetivo de obtener el mayor rendimiento de este sistema. Se trata de realizar el mayor número de proyectos en el menor tiempo, optimizando la utilización de los recursos disponibles para el conjunto de proyectos (González Cruz, Asencio Cuesta, Diego Más, & Alcaide Marzal, 2009).

### **2.1.6 Odoo**

Open ERP (conocido anteriormente como TinyERP, y ahora conocido como Odoo) es un sistema de ERP integrado de código abierto actualmente producido por OpenERP s.a. cuya sede matriz se encuentra en Bélgica, Actualmente han cambiado el nombre de OpenERP por el de Odoo. El fabricante define su producto como una alternativa de código abierto a SAP ERP y Microsoft Dynamics, así como el ERP de código abierto más sencillo y destacado del momento.

La mayoría de los módulos de OpenERP son lanzados bajo la licencia AGPL (anteriormente la GPL) y algunas partes utilizan una derivada de la Mozilla Public License.<sup>1</sup> como consecuencia directa, OpenERP no requiere ninguna tasa para ser utilizado, a diferencia de los líderes del mercado. Eso también implica que, mientras que se respeten los términos de la licencia, la modificación directa del programa es posible. (OpenERP SA, 2014).

### **2.1.6.1 Arquitectura Odoo**

#### **2.1.6.1.1 Arquitectura cliente-servidor**

OpenERP tiene componentes cliente y servidor separados. El servidor se ejecuta independientemente del cliente y maneja la lógica de negocio y se comunica con la aplicación de base de datos. El cliente presenta la información a los usuarios y les permite interactuar con el servidor. Hay disponibles múltiples aplicaciones cliente.

#### **2.1.6.1.2 Servidor y módulos**

La parte del servidor está escrita en python. El cliente se comunica con él a través de interfaces XML-RPC. La funcionalidad del negocio se organiza en módulos. Los módulos no son más que meras carpetas con una estructura predefinida, con código en python y archivos XML en su interior. Un módulo define la estructura de los datos, formularios, informes, menús, procedimientos, flujos de trabajo, etc. Los módulos se definen mediante una sintaxis independiente del cliente, de tal forma que añadir nuevos objetos, como menús y formularios los hace disponibles para cualquier cliente.

En el servidor se definen los métodos de acceso al motor de base de datos que para esta aplicación solo trabaja con Postgresql.

#### **2.1.6.1.3 Aplicaciones cliente**

Los clientes son livianos porque no contienen la lógica del negocio. Se da soporte a dos aplicaciones oficialmente:

- ❖ Una aplicación web implementada como un servidor HTTP que permite a los usuarios conectarse mediante un navegador de internet.
- ❖ Una aplicación de escritorio escrita en python utilizando el kit de herramientas GTK+ (obsoleta a partir de la versión 7). No obstante, la comunidad ha desarrollado otros clientes alternativos.

#### **2.1.6.1.4 Informes**

OpenERP también incluye un sistema de reportes con integración con openoffice.org, lo que permite personalizar los informes. También hay motores de reportes alternativos utilizando webkit o jaspersoft.

#### **2.1.6.1.5 Código fuente y contribuciones**

El código fuente de OpenERP se aloja en launchpad, utilizando el sistema de control de versiones bazaar. Las contribuciones y la documentación también se administran mediante launchpad. Un sitio web dedicado a recopilar toda la documentación fue lanzado en 2009. A partir del cambio de denominación a Odoo (versión 8) se comenzó a utilizar a github como plataforma para control de versiones (OpenERP SA, 2014).

#### **2.1.6.2 Gestión de Proyectos con Odoo**

Odoo posee módulos relacionados con la administración de proyectos, basados en CPM, que además integran el análisis financiero a través de la contabilidad analítica que posee el ERP.

El modelo de administración propuesto posee las estructuras necesarias para almacenar la siguiente información:

- ❖ Información básica del proyecto
  - Fechas de inicio y fin
  - Administrador del proyecto
  - Recursos asignados
  - Etapas
  - Tareas
- ❖ Recurso asignado
- ❖ Tiempo requerido
- ❖ Etapa a la que pertenece
- ❖ Tareas dependencia(padre)
- ❖ Tareas dependientes(delegaciones)
- ❖ Fecha inicio y fin de la tarea

Entre las características propias de la plataforma se recalca el hecho de que todos los documentos están orientados al concepto de socialización, la mensajería instantánea y automatizada de parte de la plataforma.

En la ilustración # 6 se puede revisar que la plataforma ya posee estructuralmente lo necesario para el registro de un proyecto de manera básica.

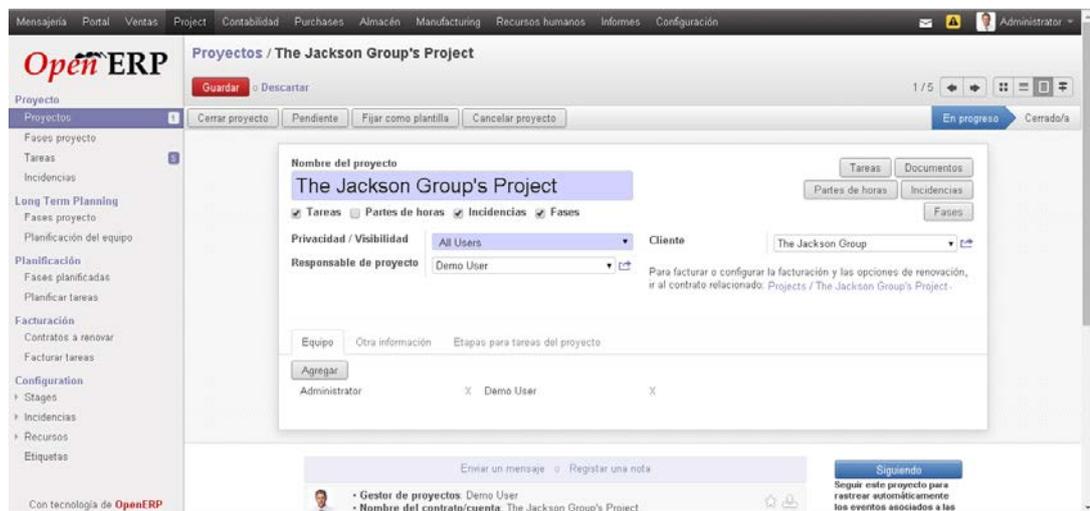


Ilustración 6: Registro de Proyecto en Plataforma Odoo (OpenERP)

Elaborado por: Los Autores

En la ilustración # 7 se puede ver la pantalla que posee actualmente Odoo para el registro de las tareas de un proyecto, y sirve como base para el desarrollo del proyecto.

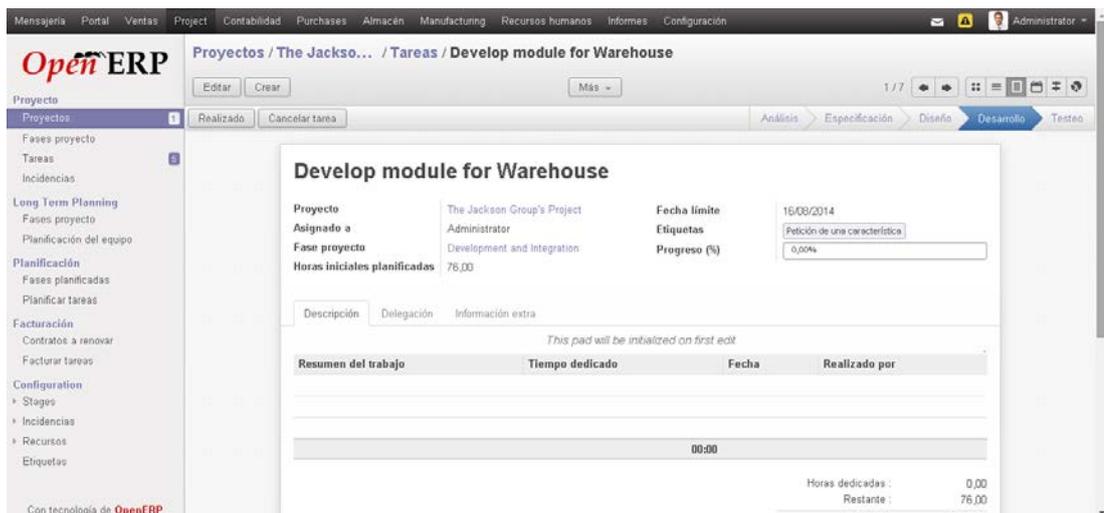


Ilustración 7: Registro de Tarea de Proyecto Plataforma Odoo (OpenERP)

Elaborado por: Los Autores

### **2.1.7 Oerplib**

Esta herramienta sirve para poder programar sobre Python, y hacer lecturas de datos, escrituras, desde scripts simples escritos ya sea en consola o a través de archivos planos que se ejecutan fuera de la plataforma Odoo, también trabaja como herramienta para documentación de las clases creadas en la plataforma Odoo (ABF Osiell, 2014).

### **2.1.8 Postgresql**

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones sus características son similares a otras bases de datos comerciales.

PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando.

A continuación en ilustración # 8 se puede ver de manera general los componentes más importantes en un sistema PostgreSQL. (Martinez Guerrero, 2010).

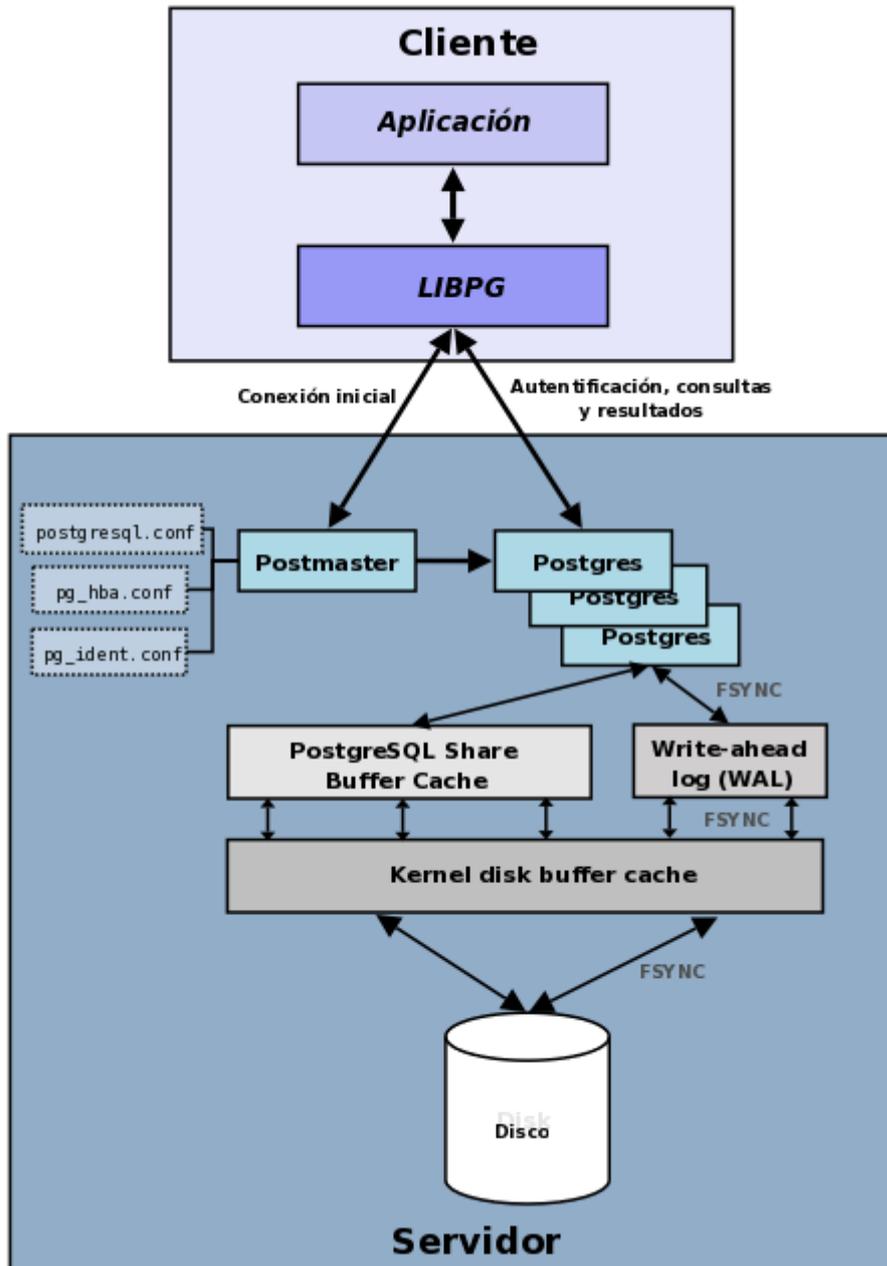


Ilustración 8: Componentes Postgresql

Autor: (Martinez Guerrero, 2010)

### 2.1.8.1 Características

La última serie de producción es la 9.3. Sus características técnicas la hacen una de las bases de datos más potentes y robustas del mercado. Su desarrollo comenzó hace más de 16 años, y durante este tiempo, estabilidad, potencia, robustez, facilidad de administración e implementación de estándares han sido las características que más se han tenido en cuenta durante su desarrollo. PostgreSQL funciona muy bien con

grandes cantidades de datos y una alta concurrencia de usuarios accediendo a la vez al sistema (Martinez Guerrero, 2010).

A continuación se muestra algunas de las características más importantes y soportadas por PostgreSQL: (Martinez Guerrero, 2010)

### **2.1.8.2 Características Generales**

- ❖ Es una base de datos 100% ACID
- ❖ Integridad referencial
- ❖ Tablespaces
- ❖ Nested transactions (savepoints)
- ❖ Replicación asincrónica/sincrónica / Streaming replication - Hot Standby
- ❖ Two-phase commit
- ❖ PITR - point in time recovery
- ❖ Copias de seguridad en caliente (Online/hot backups)
- ❖ Unicode
- ❖ Juegos de caracteres internacionales
- ❖ Regionalización por columna
- ❖ Multi-Version concurrency control (MVCC)
- ❖ Múltiples métodos de autenticación
- ❖ Acceso encriptado via SSL
- ❖ Actualización in-situ integrada (pg\_upgrade)
- ❖ SE-postgres
- ❖ Completa documentación
- ❖ Licencia BSD
- ❖ Disponible para Linux y UNIX en todas sus variantes (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows 32/64bit.
- ❖ Funciones/procedimientos almacenados (stored procedures) en numerosos lenguajes de programación, entre otros PL/pgSQL (similar al PL/SQL de oracle), PL/Perl, PL/Python y PL/Tcl
- ❖ Bloques anónimos de código de procedimientos (sentencias DO)
- ❖ Numerosos tipos de datos y posibilidad de definir nuevos tipos. Además de los tipos estándares en cualquier base de datos, se tiene disponible, entre otros,

tipos geométricos, de direcciones de red, de cadenas binarias, UUID, XML, matrices, etc

- ❖ Soporta el almacenamiento de objetos binarios grandes (gráficos, videos, sonido)
- ❖ APIs para programar en C/C++, Java, .Net, Perl, Python, Ruby, Tcl, ODBC, PHP, Lisp, Scheme, Qt y muchos otros.

Algunos de los límites de PostgreSQL son presentados en la tabla # 1: (Martinez Guerrero, 2010)

Tabla 1: Características Técnicas Postgresql

<b>Límite</b>	<b>Valor</b>
<b>Máximo tamaño base de dato</b>	Ilimitado (Depende de tu sistema de almacenamiento)
<b>Máximo tamaño de tabla</b>	32 TB
<b>Máximo tamaño de fila</b>	1.6 TB
<b>Máximo tamaño de campo</b>	1 GB
<b>Máximo número de filas por tabla</b>	Ilimitado
<b>Máximo número de columnas por tabla</b>	250 - 1600 (dependiendo del tipo)
<b>Máximo número de índices por tabla</b>	Ilimitado

Elaborado por: (Martinez Guerrero, 2010)

### 2.1.9 Python

Python es un lenguaje de programación creado por Guido van Rossum a principios de los años 90 cuyo nombre está inspirado en el grupo de cómicos ingleses “Monty Python”. Es un lenguaje similar a perl, pero con una sintaxis muy limpia y que favorece un código legible.

Se trata de un lenguaje interpretado o de script, con tipado dinámico, fuertemente tipado, multiplataforma y orientado a objetos. (González Duque, 2010).

### **Lenguaje interpretado o de script**

Un lenguaje interpretado o de script es aquel que se ejecuta utilizando un programa intermedio llamado intérprete, en lugar de compilar el código a lenguaje máquina que pueda comprender y ejecutar directamente una computadora (lenguajes compilados). La ventaja de los lenguajes compilados es que su ejecución es más rápida, sin embargo los lenguajes interpretados son más flexibles y más portables.

Python tiene, no obstante, muchas de las características de los lenguajes compilados, por lo que se podría decir que es semi interpretado. En Python, como en Java y muchos otros lenguajes, el código fuente se traduce a un pseudo código máquina intermedio llamado bytecode la primera vez que se ejecuta, generando archivos .pyc o .pyo (bytecode optimizado), que son los que se ejecutarán en sucesivas ocasiones. (González Duque, 2010).

### **Tipado dinámico**

La característica de tipado dinámico se refiere a que no es necesario declarar el tipo de dato que va a contener una determinada variable, sino que su tipo se determinará en tiempo de ejecución según el tipo del valor al que se asigne, y el tipo de esta variable puede cambiar si se le asigna un valor de otro tipo. (González Duque, 2010).

### **Fuertemente tipado**

No se permite tratar a una variable como si fuera de un tipo distinto al que tiene, es necesario convertir de forma explícita dicha variable al nuevo tipo previamente. Por ejemplo, si se tiene una variable que contiene un texto (variable de tipo cadena o string) no se puede tratarla como un número (sumar la cadena "9" y el número 8). En otros lenguajes el tipo de la variable cambiaría para adaptarse al comportamiento esperado, aunque esto es más propenso a errores. (González Duque, 2010).

### **Multiplataforma**

El intérprete de Python está disponible en multitud de plataformas (UNIX, Solaris, Linux, DOS, Windows, OS/2, Mac OS, etc.) por lo que si se utiliza librerías específicas de cada plataforma el proyecto podrá correr en todos estos sistemas sin grandes cambios. (González Duque, 2010).

### **Orientado a objetos**

La orientación a objetos es un paradigma de programación en el que los conceptos del mundo real relevantes que para el proyecto se trasladan a clases y objetos en el módulo. La ejecución del módulo consiste en una serie de interacciones entre los objetos. Python también permite la programación imperativa, programación funcional y programación orientada a aspectos. (González Duque, 2010).

#### **2.1.10 Xml**

eXtensible Markup Language(XML) son las siglas del lenguaje de etiquetado extensible. La expresión se forma a partir del acrónimo de la expresión inglesa eXtensible Markup Language. Se trata también de un lenguaje estándar que posee una recomendación del World Wide Web Consortium: Extensible Markup Languages (XML) (<http://www.w3.org/TR/REC-xml/>). Con la palabra "Extensible" se alude a la no limitación en el número de etiquetas, ya que permite crear aquellas que sean necesarias.

Xml es un lenguaje que permite jerarquizar y estructurar la información y describir los contenidos dentro del propio documento, así como la reutilización de partes del mismo. La información estructurada presenta varios contenidos (texto, imágenes, audio, etc.) y formas: hojas de cálculo, tablas de datos, libretas de direcciones, parámetros de configuración, dibujos técnicos, etc. La forma da alguna indicación de qué papel puede jugar el contenido (por ejemplo, el contenido de una sección encabezada con un significado difiere del contenido de una nota a pie de página, lo que significa algo diferente que el contenido de un pie de foto o el contenido de una tabla de datos). Más o menos todos los documentos tienen la misma estructura. (Lamarca Lapuente, 2013)

Según la especificación, los objetivos de diseñar xml fueron los siguientes: (Lamarca Lapuente, 2013)

- ❖ Xml debe ser directamente utilizable en internet
- ❖ Xml debe soportar una amplia variedad de aplicaciones
- ❖ Xml debe ser compatible con sgml
- ❖ Debería se sencillo escribir programas que procesaran documentos xml
- ❖ El número de las características opcionales en xml debería ser el mínimo posible, a ser posible cero
- ❖ Los documentos xml deberían ser legibles por las personas y razonablemente claros
- ❖ El diseño de xml debe ser rápido
- ❖ Xml debería ser simple, pero perfectamente normalizado
- ❖ Los documentos xml deben ser de fácil creación
- ❖ La concisión de las marcas xml tiene una importancia mínima

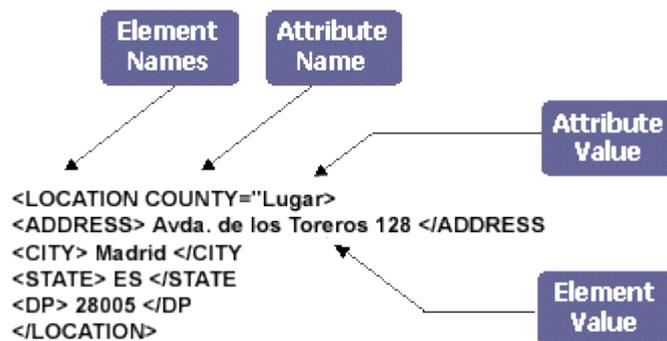


Ilustración 9: Explicación de estructura xml

Elaborado por: (Lamarca Lapuente, 2013)

Entre las ventajas de adoptar xml se puede destacar las siguientes: los autores o proveedores pueden diseñar sus propios documentos a medida diseñando e inventando sus propias etiquetas dependiendo de las funciones que quieran dar a los datos. La información contenida en un documento xml puede ser más rica y fácil de usar porque las habilidades hipertextuales de xml son mayores que las de html. Xml puede ofrecer más facilidades para la representación en los navegadores ya que elimina muchas de las complejidades de sgml en aras de una mayor flexibilidad del modelo, con lo que la escritura de programas para manejar xml es mucho más sencilla. La información será más accesible y reutilizable porque la flexibilidad de las etiquetas de xml puede utilizarse sin tener que amoldarse a las reglas específicas

de un fabricante como es el caso de html. Los archivos xml válidos son válidos también en sgml, por lo que también pueden utilizarse fuera de la web en un entorno sgml. (Lamarca Lapuente, 2013)

Los principales usos de xml son los siguientes: (Lamarca Lapuente, 2013)

- ❖ Xml aplicado a los sitios web: permite separar contenido y presentación, y que los mismos datos se puedan mostrar de varias formas distintas sin demasiado esfuerzo.
- ❖ Xml para la comunicación entre aplicaciones: representación de los datos muy simple, fácil de transmitir por la red, estándar. En los últimos tiempos este uso se está haciendo muy popular con el surgimiento de los Servicios web.
- ❖ Xml para la configuración de programas: representación de los datos simple y estándar, en contraposición con los crípticos formatos propietarios. (Lamarca Lapuente, 2013)

Algunas de las aplicaciones Xml son las siguientes: (Lamarca Lapuente, 2013)

- ❖ Ofrece mecanismos más versátiles para mostrar datos. Los últimos desarrollos como DOM (Document Object Model) desarrollan APIs (Application Programming interface) que soportan tanto los procesadores de xml como html. Se pueden representar (a través de javascripts o applets de Java) documentos xml en los navegadores web de una forma más sofisticada que los documentos html, ya que xml no sólo proporciona una sintaxis, sino también una semántica.
- ❖ Los buscadores inteligentes pueden buscar la información de forma mucho más precisa, ya que xml etiqueta por su significado de forma muy precisa y se puede localizar de una forma más clara que en los documentos html. Con las DTDs (Document Type Declaration) estandarizadas o con los esquemas de xml para distintas aplicaciones, se podría programar buscadores web que recuperen la información sobre un producto de cualquier página web sabiendo que todos tendrán el mismo formato de datos (gracias al dtd o al esquema), aunque no tengan necesariamente la misma representación gráfica (gracias a xml/xls).

- ❖ La creación de una dtd para crear topic maps en xml, denominada xtm ha supuesto un salto importante para el desarrollo de la llamada web semántica.
- ❖ Intercambio de información entre sistemas diversos y heterogéneos. Ya que las dtd y, sobre todo, los esquemas proporcionan un formato estándar para representar la información sobre un tema específico, puede usarse para facilitar y simplificar el intercambio de información entre distintas fuentes.

#### **2.1.10.1 Estructura de XML**

Los documentos xml se componen de unidades de almacenamiento llamadas objetos o entidades (entities), que contienen datos analizados (parsed) o datos sin analizar (unparsed). Los datos analizados se componen de caracteres, algunos de los cuales forman los datos del documento (data) y el resto forman las etiquetas (markups) o marcas. Las etiquetas codifican la descripción de la estructura lógica y de almacenamiento del documento. xml proporciona un mecanismo para imponer limitaciones en la estructura lógica y de almacenamiento.

Para leer los documentos xml y acceder a su contenido y estructura, se utiliza un software o programa procesador de xml. Cualquier aplicación que trabaje sobre xml necesita ese módulo o procesador xml (parser). Dicho módulo lee los documentos y proporciona acceso a su contenido y estructura.

Para poder llevar a cabo esa función, la aplicación debe proporcionar información al procesar xml de cómo se encuentra almacenada esa información a través de un DTD (Document Type Declaration) o de un esquema (schema), también conocido como vocabulario. El dtd o declaración del tipo de documento y los esquemas, proporcionan la gramática para una clase de documentos xml. Estos mecanismos se utilizan para la llamada validación tanto estructural como formal del documento, esto es, enviar un documento a un destinatario junto con las condiciones que deben cumplir los documentos. En la actualidad, los esquemas se utilizan en mayor medida que los dtd's dentro de xml.

Los documentos xml pueden ser de 2 tipos: documentos bien formados y documentos válidos. Un documento está bien formado si, considerado en su conjunto, encaja con la especificación xml de producción. Un documento bien formado es, además, válido, si tiene dtd o Esquema. (Lamarca Lapuente, 2013)

Un archivo xml válido comienza con una dtd o declaración de tipo de documento tal como lo muestra el siguiente ejemplo:

```
<?xml version="1.0"?>
<!DOCTYPE noticia SYSTEM "http://www.hipertexto.info/ad.dtd">
<noticia>
<titulo>...<foto/>...</titulo>
<texto>...</texto>
</noticia>
```

Dado que xml está diseñado para ser un subconjunto de sgml, cualquier documento xml válido debe ser también un documento sgml válido.

Los esquemas indican tipos de datos, número mínimo y máximo de ocurrencias y otras características más específicas. He aquí un ejemplo de esquema (xml schema):

```
<schema targetNamespace="http://www.hipertexto.info/Schema_y_DTDs"
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:HPT="http://www.hipertexto.info/Schema_y_DTDs" >
<element name="Articulo" type="HIP:tArticulo" />
<complexType name="tArticulo">
<element name="Cabecera" type="HPT:tCabecera"/>
<element name="Cuerpo" type="HPT:tCuerpo"/>
<element name="Final" type="HPT:tFinal"/>f
</complexType>
<complexType name="tCabecera">
<element name="Titulo" type="string"/>
<element name="Autor" type="string"/>
</complexType>
```

Un enlace xml puede ser bien un URL (como en los enlaces tradicionales, que aquí se conocen como XLink), bien a un puntero extendido (aquí conocido como XPointer o Xptr), o ambos. Un url en sí misma se asume que tiene que ser un recurso (como en HTML). Si le sigue un XPointer, se asume que es un subrecurso de ese url. Un XPointer en sí mismo se asume que se aplica al documento actual. (Lamarca Lapuente, 2013)

### **2.1.11 Xpath**

XPath es la abreviatura de xml path language(lenguaje de ruta xml), este usa una sintaxis no xml que provee una vía flexible de direccionar o apuntar a cualquier punto dentro de un documento xml.

En Odoo este lenguaje es usado particularmente para determinar en las vistas declaradas en xml, las operaciones de agregado o reemplazo, posibilitando la capacidad de crear definiciones de vistas independientes del código original de las vistas de los módulos pertenecientes a Odoo, manteniendo así los módulos originales sin cambios. (Mozilla Developer Network, 2014).

### **2.1.12 Java**

Es un lenguaje de programación de propósito general, concurrente, basado en clases, y un lenguaje orientado a objetos. Está diseñado para ser lo suficientemente simple para que la mayoría de los programadores puedan alcanzar fluidez en el lenguaje.

Es un lenguaje fuertemente tipado, un lenguaje de alto nivel, también es necesaria la compilación en instrucciones binarias llamadas bytecode, según el formato definido para la máquina virtual java.

La máquina virtual de java le permite ser multiplataforma, manteniendo un ejecutable común llamado java.exe en las plataformas Windows, o java en los archivos basados en unix, por medio de lo cual se permite a través de las consolas intérpretes de comandos su llamada y paso de parámetros (Addison-Wesley Professional, 2000).

### **2.1.13 JavaScript**

JavaScript es un lenguaje ligero, interpretado y orientado a objetos con funciones de primera clase, más conocido como el lenguaje interpretado para las páginas web, pero que también es usado en ambientes no basado en exploradores web tales como node.js o Apache CouchDB. Este es basado en prototipos, un lenguaje interpretado multi-paradigmas que es dinámico, y soporta la orientación a objetos, imperativo y programación funcional (Mozilla Developer Network, 2014).

### **2.1.13.1 Breve historia**

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

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

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

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

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

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

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

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

La organización internacional para la estandarización (ISO) adoptó el estándar ECMA-262 a través de su comisión IEC, dando lugar al estándar ISO/IEC-16262. (Eguiluz, 2013).

## 2.2 Marco legal

Para este proyecto se han usado librerías, y software de código abierto, sin embargo cada uno de ellos posee un licenciamiento diferente el cual se detalla a continuación en la tabla # 2.

Tabla 2: Licencias Usadas en componentes del Proyecto

<b>Software / Librería</b>	<b>Versión</b>	<b>Licencia</b>	<b>Versión de Licencia</b>
<b>Odo</b>	8.0	Agpl	3.0
<b>Postgresql</b>	9.3	Bsd	1.0
<b>Highcharts</b>	4.0.4	Creative Commons	3.0
<b>Python</b>	2.7	Python Software foundation license (PSFL)	1.0
<b>Python-graph</b>	1.8.2	Mit	1.0
<b>Java</b>	1.7	Propietaria(Oracle)	SN
<b>Mpxj</b>	4.5.0	Lgpl	3.0
<b>Commons-Cli</b>	1.2	Apache	2.0
<b>Log4j</b>	1.2.14	Apache	2.0
<b>Super-csv</b>	2.2.0	Apache	2.0

Elaborado por: Los Autores

### **2.2.1 Licencia Pública General Affero de GNU (AGPL) versión 3 (#AGPL) (#AGPLv3.0)**

Esta es una licencia de software libre con copyleft. Sus términos son en la práctica los mismos de la GPLv3, con un párrafo adicional en la sección 13 que permite a los usuarios que interactúan con el software bajo esta licencia en una red, recibir la fuente de tal software. Se recomienda a los desarrolladores que consideren el uso de la AGPL de GNU para todo el software que se ejecute de forma habitual en una red.

Nótese que la AGPL de GNU no es compatible con GPLv2. Tampoco es técnicamente compatible con la GPLv3 en sentido estricto: no se puede tomar código publicado bajo la AGPL de GNU, copiarlo o modificarlo a gusto bajo los términos de la GPLv3, o viceversa. Sin embargo, sí le está permitido combinar módulos separados o archivos fuente publicados bajo ambas licencias y obtener así un único proyecto, lo cual les otorgará a muchos programadores los permisos necesarios para hacer los programas que deseen. (Free Software Foundation, 2015).

### **2.2.2 Licencia BSD Modificada (#ModifiedBSD)**

Esta es la licencia BSD original, modificada por la eliminación de la cláusula de publicidad. Es una licencia de software libre laxa, permisiva, sin copyleft, compatible con la GPL de GNU.

A esta licencia se la llama algunas veces «licencia BSD de 3 cláusulas».

Como licencia laxa, permisiva, la BSD modificada no es tan mala, aunque es preferible la Apache 2.0. Sin embargo, es peligroso recomendar el uso de la «licencia BSD», incluso en casos especiales como por ejemplo para programas pequeños, porque fácilmente se podría producir una confusión que llevaría al uso de la defectuosa licencia BSD original. Para evitar ese riesgo, se puede sugerir el uso de la licencia X11. Las licencias X11 y BSD modificada son más o menos equivalentes. (Free Software Foundation, 2015).

### **2.2.3 Licencia Creative Commons Reconocimiento 4.0 (también conocida como CC BY) (#ccby)**

Esta es una licencia libre sin copyleft, adecuada para obras artísticas, de entretenimiento y educativas. Aunque es compatible con todas las versiones de la GPL de GNU, no se recomienda su uso para software.

(#which-cc) Creative Commons publica muchas licencias que son muy diferentes. Por lo tanto, decir que una obra utiliza «una licencia de Creative Commons» es dejar sin respuesta las preguntas principales acerca del licenciamiento de la obra. (Free Software Foundation, 2015).

### **2.2.4 Licencia de Expat (#Expat)**

Esta es una licencia de software libre laxa, permisiva, sin copyleft, compatible con la GPL de GNU. Algunas veces se la llama ambiguamente «MIT License». (Free Software Foundation, 2015).

### **2.2.5 Licencia Apache, versión 2.0 (#apache2)**

Esta es una licencia de software libre compatible con la versión 3 de la GPL de GNU.

Nótese que esta licencia no es compatible con la versión 2 de la GPL de GNU porque tiene algunos requisitos que no están en esa versión de la GPL, por ejemplo ciertas disposiciones sobre las indemnizaciones y la terminación por patentes. La disposición sobre las patentes es buena, por eso para programas de una cierta envergadura se recomienda usar la licencia Apache 2.0 en lugar de otras licencias laxas, permisivas. (Free Software Foundation, 2015).

### **2.2.6 Python Software Foundation License**

Es una licencia de software libre permisiva, en la línea de la licencia BSD, es decir, que cumple con los requisitos para ser declarada licencia de software libre; por lo que la hace compatible con la licencia GPL. A diferencia de la licencia GPL, y como la mayoría de licencias tipo BSD, la licencia PSFL no es una licencia copyleft, y permite modificaciones del código fuente, así como la creación de trabajos

derivados, sin requerir que ni las modificaciones ni los trabajos derivados tengan que ser a su vez de código abierto. La licencia PSFL está dentro de las listas de licencias aprobadas tanto por la Free Software Foundation como por la Open Source Initiative. (EcuRed, 2014).

## CAPÍTULO III

### ANÁLISIS DEL SISTEMA

#### 3.1 Requerimientos Funcionales:

##### 3.1.1 Proceso de importación de archivo CSV:

Considerando que este módulo está dirigido a empresas, estudiantes, profesionales independientes, que actualmente ya tengan una trayectoria en la administración de proyectos, se ha identificado la necesidad de crear una opción para facilitar el ingreso de datos de proyectos que han sido creados en otras herramientas.

Tabla 3: Requerimiento Funcional Proceso de Importación archivos CSV

<b>ID:</b>	RF1	<b>Relación:</b>	N.A.
<b>Prioridad:</b> Normal			
<b>Descripción:</b> Proceso de importación de archivo CSV			

Elaborado por: Los autores

##### 3.1.2 Creación de un proyecto

Al inicio del proyecto, se deberá considerar que el proyecto deberá tener un responsable o administrador del mismo, quién será encargado de identificar los datos principales que debe tener un proyecto, por lo que el módulo advertirá en caso de que no se ingrese alguno de estos datos al momento de crear un proyecto.

Tabla 4: Requerimiento Funcional Creación de un proyecto

<b>ID:</b>	RF2	<b>Relación:</b>	N.A.
<b>Prioridad:</b> Normal			
<b>3.1.3 Descripción:</b> Creación de un proyecto			

Elaborado por: Los autores

### 3.1.4 Creación de tareas de un proyecto

Un proyecto debe ser creado con actividades o tareas, las mismas que .deben tener una fecha de inicia y de fin definida por la duración, adicionalmente se debe considerar un responsable que ejecutará la tarea.

Al momento de crear una actividad, debe considerarse que una misma tarea puede contener dependencias y puede tener más de un recurso responsable de ejecutar la tarea.

Tabla 5: Requerimiento Funcional Creación de tareas

<b>ID:</b>	RF3	<b>Relación:</b>	N.A.
<b>Prioridad:</b> Normal			
<b>Descripción:</b> Creación de tareas			

Elaborado por: Los autores

### 3.1.5 Cálculo de Ruta Crítica

El ingreso de las tareas ya sea por importación o ingreso manual, el sistema debe estar en capacidad de identificar a estas tareas como parte de la ruta crítica.

El sistema debe tomar a cada una de las tareas del proyecto, en la medida en que se vayan agregando y calcular la ruta crítica, de tal manera que en un campo verdadero/false se asigne el valor de verdadero en caso de pertenecer a la ruta crítica, en caso de no pertenecer este campo debe permanecer en falso.

Por costo en rendimiento la operación tendrá que ser calculada cuando se ejecute el cálculo de la cadena crítica en general, para lo cual existirá un asistente específico para mostrar el resumen del cálculo antes de aplicarlo.

Tabla 6: Requerimiento Funcional Cálculo de ruta crítica

<b>ID:</b>	RF4	<b>Relación:</b>	N.A.
<b>Prioridad:</b> Normal			

**Descripción:** Cálculo de Ruta Crítica

Elaborado por: Los autores

### 3.1.6 Cálculo de Tareas Alimentadoras

Luego de establecer la ruta crítica, es necesario identificar a las tareas que son dependencias para alguna o varias tareas de la ruta crítica.

Tabla 7: Requerimiento Funcional Cálculo de tareas alimentadoras

<b>ID:</b>	RF5	<b>Relación:</b>	N.A.
<b>Prioridad:</b> Normal			
<b>Descripción:</b> Cálculo de tareas alimentadoras			

Elaborado por: Los autores

### 3.1.7 Restricción de Multitareas (Sobre asignaciones)

El sistema debe tener la capacidad de redistribuir los recursos de tal manera que no se sobre asignen, es decir no estén asignados a tareas que se ejecutan en el mismo momento, ni con cortes entre tareas, es decir las tareas deben ejecutarse de corrido, y sin saltos entre tareas.

Tabla 8: Requerimiento Funcional Restricción de multitareas

<b>ID:</b>	RF6	<b>Relación:</b>	N.A.
<b>Prioridad:</b> Normal			
<b>Descripción:</b> Restricción de Multitareas			

Elaborado por: Los autores

### 3.1.8 Reducción de Tiempo de Tareas del Proyecto

Se asume que las tareas son ingresadas con tiempos de holgura, por lo que según la configuración del proyecto, el sistema debe poder ejecutar una acción para reducir el tiempo de las tareas para asignársela posteriormente al amortiguador del proyecto.

Tabla 9: Requerimiento Funcional Reducción de tiempo de tareas

<b>ID:</b>	RF7	<b>Relación:</b>	N.A.
<b>Prioridad:</b> Normal			
<b>Descripción:</b> Reducción de Tiempo de Tareas			

Elaborado por: Los autores

### 3.1.9 Cálculo de Amortiguador de Tareas Alimentadoras

Las tareas alimentadoras de la ruta crítica deben ser identificadas plenamente, son las que son dependencias de alguna o algunas tareas de la ruta crítica, para que estas deban iniciar lo más pronto posible, y su tiempo de holgura será considerado como el amortiguador hasta que se integre con la cadena crítica.

Tabla 10: Requerimiento Funcional Cálculo de amortiguador de tareas alimentadoras

<b>ID:</b>	RF8	<b>Relación:</b>	N.A.
<b>Prioridad:</b> Normal			
<b>Descripción:</b> Requerimiento Funcional Cálculo de amortiguador de tareas alimentadoras			

Elaborado por: Los autores

### 3.1.10 Cálculo de Amortiguador del Proyecto

De todas las tareas pertenecientes a la ruta crítica se hará el cálculo del 50% del tiempo, en el caso de obtener valores decimales, se redondeara hacia el entero inmediato inferior, el tiempo obtenido del cálculo de reducción de todas las tareas se acumulará de tal manera que este tiempo que se calcule de las tareas que pertenecen a la ruta crítica se agregará como un amortiguador general del proyecto, y servirá para al cálculo del reporte de penetración de amortiguador.

Tabla 11: Requerimiento Funcional Cálculo de amortiguador del proyecto

<b>ID:</b>	RF9	<b>Relación:</b>	N.A.
<b>Prioridad:</b> Normal			
<b>Descripción:</b> Cálculo de amortiguador del proyecto			

Elaborado por: Los autores

### 3.1.11 Reporte de Penetración del Proyecto

El sistema calculará en base al día actual, los valores de porcentaje de avance del proyecto y el tiempo de amortiguador.

Tabla 12: Requerimiento Funcional Reporte de penetración del proyecto

<b>ID:</b>	RF10	<b>Relación:</b>	N.A.
<b>Prioridad:</b> Normal			
<b>Descripción:</b> Reporte de penetración del proyecto			

Elaborado por: Los autores

### 3.1.12 Reporte de Tareas Urgentes

En este reporte se mostrará las tareas que necesitan mayor atención, ya sea por su tiempo de atraso o penetración de amortiguador.

Tabla 13: Requerimiento Funcional Reporte de tareas urgentes

<b>ID:</b>	RF11	<b>Relación:</b>	N.A.
<b>Prioridad:</b> Normal			
<b>Descripción:</b> Reporte de tareas urgentes			

Elaborado por: Los autores

## 3.2 Requerimientos No Funcionales:

Para la correcta ejecución del módulo y para garantizar que la aplicación cumpla con las especificaciones funcionales, es necesario especificar las bondades no funcionales que tendrá el módulo.

### 3.2.1 Sistema Operativo del Servidor

Según la recomendación del fabricante del sistema operativo que debe usarse es Ubuntu, para esto es recomendable usar una versión servidor LTS (Long Term Support), soporte largo.

A nivel de Hardware, se requiere como mínimo lo siguiente:

- Procesador: Intel Dual Core de 2.0 Ghz mínimo
- Memoria: 4 Gb DDR2 800 MHZ mínimo
- Disco Duro: 50 Gb mínimo

Tabla 14: Requerimiento no funcional Sistema operativo del servidor

<b>ID:</b>	RNF1	<b>Relación:</b>	N.A.
<b>Descripción:</b> Sistema Operativo del Servidor			

Elaborado por: Los autores

### 3.2.2 Base de datos Postgresql

Se debe poder trabajar en la versión más reciente del motor de base de datos Postgresql 9.3.

Tabla 15: Requerimiento no funcional base de datos postgresql

<b>ID:</b>	RNF2	<b>Relación:</b>	N.A.
<b>Descripción:</b> Base de datos Postgresql			

Elaborado por: Los autores

### 3.2.3 Navegador para Usuarios

El sistema debe ser compatible con las siguientes versiones de navegadores web:

- Internet Explorer: Versión 11
- Mozilla Firefox: Versión 32
- Google Chrome: Versión 38

Tabla 16: Requerimiento no funcional Navegador para usuarios

<b>ID:</b>	RNF1	<b>Relación:</b>	N.A.
<b>Descripción:</b> Navegador para Usuarios			

Elaborado por: Los autores

### 3.3 Casos de Uso

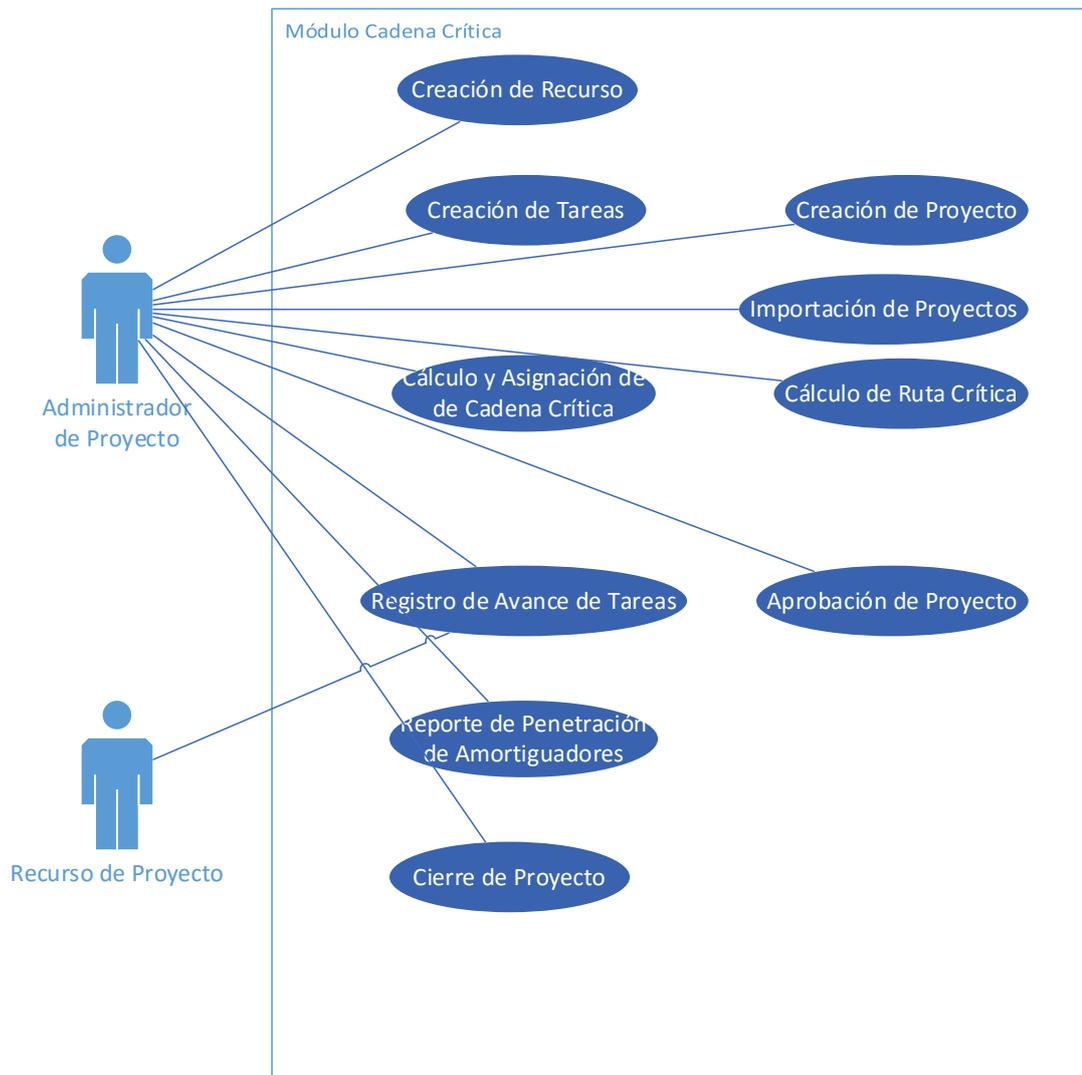


Ilustración 10: Diagrama de Casos de Uso

Elaborado por: Los Autores

### 3.3.1 Importación de Proyectos

En la tabla # 17 se especifica cada detalladamente las funcionalidades que tiene el módulo para el proceso de la importación del archivo CSV.

Tabla 17: Caso de Uso Importación de Proyecto

<b>CU1</b>	Importación de Proyecto
<b>Actores</b>	Administrador del proyecto
<b>Pre- condiciones</b>	<p>El archivo CSV deberá tener los siguientes campos:</p> <ol style="list-style-type: none"> <li>1. Identificador de la Tarea</li> <li>2. Descripción de la Tarea</li> <li>3. Recursos Asignado</li> <li>4. Tiempo de la Tarea</li> <li>5. Dependencias</li> </ol> <p>No debe haber nombres de recursos repetidos diferenciados por mayúsculas y minúsculas.</p> <p>No debe existir otro proyecto con el mismo nombre ni código de proyecto</p>
<b>Escenarios:</b>	
<ol style="list-style-type: none"> <li>1. Ingresar fecha de inicio y final del proyecto, una descripción general del proyecto y un responsable del proyecto que podría ser el administrador del proyecto</li> <li>2. Escoger el tipo de archivo csv</li> <li>3. Consultar un archivo demostrativo es decir una plantilla de datos de ejemplo</li> <li>4. Adjuntar el archivo a procesar</li> <li>5. Procesar para que el sistema evalúe la integridad de los datos</li> <li>6. En caso de no existir algún dato, el sistema muestra un mensaje de advertencia. Además mostrará los datos faltantes por cada actividad</li> <li>7. El sistema al procesar el archivo y pasar todas las verificaciones, mostrará un resumen de los datos leídos             <ul style="list-style-type: none"> <li>○ Identificador de la Tarea</li> <li>○ Descripción de la Tarea</li> </ul> </li> </ol>	

<ul style="list-style-type: none"> <li>○ Recursos Asignado</li> <li>○ Tiempo de la Tarea</li> <li>○ Dependencias</li> </ul> <p>Una vez que el sistema procese toda la información el sistema crea un nuevo proyecto, con toda la información importada y se muestra el formulario donde se encuentra el proyecto importado</p>	
<b>Post- condiciones</b>	En cada proceso de importación el módulo creará un proyecto nuevo, validando la condición de proyectos repetidos

Elaborado por: Los autores

### 3.3.2 Creación de un recurso

Al inicio del proyecto, se deberá considerar un equipo de recursos que van a ejecutar las tareas, incluyendo al líder o administrador del proyecto quién será encargado de registrar calendario general del proyecto.

En la tabla # 18 se puede ver los pasos necesarios para la creación de un proyecto:

Tabla 18 Caso de Uso Creación de Recurso

<b>CU2</b>	Creación de equipo de proyecto
<b>Actores</b>	Administrador del proyecto
<b>Escenarios:</b>	
<ol style="list-style-type: none"> <li>1. En la opción de creación de equipos se deben registrar las siguientes acciones:</li> <li>2. Agregar los usuarios ya registrados en el sistema</li> <li>3. En caso de crear nuevo recurso, registrar el nombre del recurso</li> <li>4. Registrar el correo, el cuál será el link para autenticar su registro</li> <li>5. Registrar los números telefónicos</li> <li>6. Asignar un calendario que por defecto traerá el estándar del proyecto, pero se puede crear otro calendario con los mismos datos del calendario estándar</li> <li>7. En el recurso, también se puede seleccionar días de ausencia</li> </ol>	

Elaborado por: Los autores

### 3.3.3 Creación de Proyecto

La tabla # 19 muestra el proceso de ingreso manual a través de la plataforma de un proyecto.

Tabla 19: Caso de Uso Creación de Proyecto

<b>CU3</b>	Creación de un nuevo proyecto
<b>Actores</b>	Administrador del proyecto
<b>Escenarios:</b>	
<ol style="list-style-type: none"><li>1. Especificar el nombre único para el proyecto</li><li>2. Escoger el administrador o responsable del proyecto</li><li>3. Ingresar los recursos que tendrá el proyecto</li><li>4. Asignar las fechas de inicio y fin</li><li>5. Asignar un calendario de trabajo</li></ol>	

Elaborado por: Los autores

### 3.3.4 Creación de Tareas

La tabla # 20 muestra como la plataforma permite el ingreso y edición de tareas dentro de la plataforma.

Tabla 20: Caso de uso creación de tarea

<b>CU4</b>	Creación de tareas
<b>Actores</b>	Administrador del proyecto
<b>Escenarios:</b>	
<ul style="list-style-type: none"><li>● Agregar tareas contenedoras/agrupadoras<ul style="list-style-type: none"><li>○ Tiene una sangría a nivel de vista de árbol</li><li>○ Se visualiza en la vista Gantt</li><li>○ Se calcula automáticamente el total de las horas de sus tareas hijas</li></ul></li><li>● En cada tarea se pondrá la unidad de medida</li><li>● Cada tarea podrá escoger la tarea contenedora</li><li>● Se debe considerar que una tarea pueda tener dependencia de otra(en el caso actual, sería dentro del contexto de la delegación)</li></ul>	

- En cada tarea debería indicarse el porcentaje avanzado, según el tiempo dedicado.
- El tiempo dedicado debe también tener la unidad de medida
- El estado de la tarea debe ser automática según el porcentaje de avance
- No debe permitir registrar el tiempo invertido sin antes registrar la duración planificada de la tarea
- En el escenario de que una tarea se haya concluido antes de lo planificado, deberá tener un botón de cierre anticipado y automáticamente se debe cambiar de etapa, y no debe permitir mover de una etapa a otra manualmente
- Una tarea debe tener una fecha de inicio y la fecha fin deberá calcularse en base al tiempo planificado.
- Las fecha de inicio debe ser automáticas tendrán estos criterios de asignación:
  - Fecha de inicio del proyecto
  - Se prioriza por secuencia de ingreso de la tarea
  - Cuando tenga dependencia
  - Cuando el recurso no esté asignado en otra tarea con la misma fecha, deberá mostrar un mensaje y que se recalcule a fecha disponible

Elaborado por: Los autores

### 3.3.5 Cálculo de Ruta Crítica

Tabla 21: Caso de Uso Cálculo de Ruta Crítica

<b>CU5</b>	Cálculo de Ruta Crítica
<b>Actores</b>	Administrador del Proyecto
<b>Escenarios:</b>	
<b>Acciones</b>	
<ul style="list-style-type: none"> <li>● El usuario sobre un proyecto abierto en el formulario de edición del proyecto, deberá ejecutar la acción de cálculo de la ruta crítica</li> <li>● El sistema internamente se encargará de acuerdo al orden de creación o importación de la tareas, asignar las fechas de ejecución de cada una de ellas, deberá tomar en cuenta:           <ul style="list-style-type: none"> <li>○ El calendario general del proyecto, para solo usar días laborables</li> <li>○ El calendario asignado a los recursos, para que las tareas en las que participes se extiendan o se muevan según sea el caso</li> </ul> </li> </ul>	

- Luego de terminar este calculo

Elaborado por: Los autores

### 3.3.6 Cálculo y Asignación de Cadena Crítica

Tabla 22: Caso de Uso Cálculo y Asignación de Cadena Crítica

<b>CU6</b>	Cálculo y Asignación de Cadena Crítica
<b>Actores</b>	Administrador de Proyecto
<b>Pre-condiciones</b>	Proyecto debe estar creado, y debe haberse calculado la ruta crítica del proyecto
<b>Escenarios:</b>	
<b>Acciones</b>	<ul style="list-style-type: none"> <li>• Se debe seleccionar el formulario de edición de un proyecto</li> <li>• Dentro del formulario debe ejecutarse un asistente como acción que muestre un asistente con el resumen del cálculo de la cadena crítica</li> <li>• El asistente calculará en base a las tareas de la ruta crítica tomando de cada una la mitad del tiempo total disponible, haciendo un redondeo del valor al entero inmediato inferior, todos estos valores deben sumarse, para hacer el cálculo del amortiguador del proyecto</li> <li>• Las tareas alimentadores, serán movidas para que sean ejecutadas lo más pronto posible</li> <li>• Además se informará sobre las tareas que poseen conflictos por multitareas, y las moverá según su precedencia en el tiempo para evitar estos conflictos</li> <li>• Una vez cerrado el asistente se guardará la información de cadena crítica en el proyecto</li> </ul>

Elaborado por: Los autores

### 3.3.7 Aprobación del Proyecto

Tabla 23: Caso de Uso Aprobación del Proyecto

<b>CU7</b>	Aprobación del proyecto
<b>Actores</b>	Administrador del Proyecto
<b>Pre-condiciones</b>	Ya se debe haber realizado el cálculo y asignación de cadena crítica
<b>Escenarios:</b>	
<b>Acciones</b>	<ul style="list-style-type: none"><li>• Una vez realizado el cálculo de fechas de la cadena crítica junto con sus amortiguadores, el usuario debe confirmar el inicio del proyecto</li><li>• Esta acción cambiara el estado del proyecto a un estado en planeación que permitirá posteriormente registrar el avance de las tareas</li></ul>

Elaborado por: Los autores

### 3.3.8 Registro de Avance en Tareas

Tabla 24: Caso de uso Registro de Avance en Tareas

<b>CU8</b>	Registro de Avance en Tareas
<b>Actores</b>	Administrador del proyecto, Recurso del Proyecto
<b>Pre-condiciones</b>	El proyecto debe estar aprobado
<b>Escenarios:</b>	
<b>Acciones</b>	<ul style="list-style-type: none"><li>• El administrador o ya sea el recurso del sistema puede registrar la cantidad de horas de trabajo en las tareas propias o asignadas a si mismo</li><li>• Un recurso de proyecto no debe poder registrar avance en tareas en las que no esté asignado</li><li>• Al momento de completar el ingreso total del tiempo asignado, o superar el tiempo, la tarea debe pasar a estado de</li></ul>

	terminado o 100% completo
--	---------------------------

Elaborado por: Los autores

### 3.3.9 Reporte de Penetración de Amortiguadores

Tabla 25: Caso de uso Reporte de Penetración de Amortiguadores

<b>CU9</b>	Reporte de Penetración de Amortiguadores
<b>Actores</b>	Administrador del Proyecto
<b>Pre- condiciones</b>	Debe ejecutarse sobre un proyecto que se encuentre abierto, también se puede sacar como historial el reporte sobre proyectos terminados
<b>Escenarios:</b>	
<b>Acciones</b>	<ul style="list-style-type: none"> <li>• El usuario deberá ejecutar sobre la acción sobre un proyecto abierto</li> <li>• En un asistente debería consultar sobre cómo debe mostrar el asistente</li> <li>• El avance del tiempo tener la capacidad de mostrarse en días, semanas o meses, de tal manera que se pueda ver de manera más ampliada la penetración del amortiguador del proyecto</li> </ul>

Elaborado por: Los autores

### 3.3.10 Cierre de Proyecto

Tabla 26: Caso de Uso Cierre de Proyecto

<b>CU10</b>	Cierre de Proyecto
<b>Actores</b>	Administrador del Proyecto
<b>Pre- condiciones</b>	El proyecto debe tener al menos una tarea con registro de avance
<b>Escenarios:</b>	
	<ul style="list-style-type: none"> <li>• El cierre del proyecto supone el termino de todas las tareas, en caso de que existan tareas abierta, el sistema debe mostrar un detalle de las mismas, antes</li> </ul>

de confirmar el cierre

- Al momento de mostrar el resumen el usuario podrá modificar las fechas de termino de las tareas
- Se asume que las tareas terminaron a tiempo, es decir el sistema sugerirá la fecha de término de la tarea como la fecha planificada de término

Elaborado por: Los autores

### 3.3.11 Creación de calendario

Tabla 27: Caso de Uso: Creación de Calendario

<b>CU8</b>	Creación de calendario
<b>Actores</b>	Administrador de Proyecto
<b>Pre- condiciones</b>	Ninguna
<b>Escenarios:</b>	
<b>Acciones</b>	
<ul style="list-style-type: none"><li>• Se debe poder registrar nuevos calendarios para asignar a los recursos por proyecto</li><li>• Existirá un calendario precargado en la aplicación, con los días laborables regulares con un tiempo de trabajo de 8 horas diarias</li><li>• Se debe poder registrar días feriados, y asignarse al calendario precargado</li></ul>	

Elaborado por: Los autores

### 3.4 Definición de Roles en los Módulos:

Todos los usuarios del sistema, tendrán un acceso autenticado, por medio de un usuario y contraseña, además de un listado de preferencias como el lenguaje, y la posibilidad de cambiar su contraseña.

#### 3.4.1 Administrador del Proyecto:

- ❖ Crear Proyecto Nuevo
- ❖ Importar Archivo de Proyecto
- ❖ Calcular Cadena Crítica

- ❖ Creación de Tareas
  - Asignación de Recursos
  - Reasignación de Recursos
- ❖ Creación de Recursos
- ❖ Creación de Calendarios
- ❖ Registrar Avance en Tareas de Proyecto
- ❖ Reporte de Amortiguadores del Proyecto x Tarea
- ❖ Reporte de Amortiguador General x Proyecto

#### **3.4.2 Recurso del Proyecto:**

- ❖ Registrar Avance en Tareas de Proyecto

### 3.4.3 Definición de operaciones de base de datos permitidas

En la tabla # 28 se define que operaciones puede hacer cada rol sobre las tablas o clases del sistema, en la misma se coloca 1 para indicar que posee el permiso para realizar la acción, y el 0 para indicar que no lo tiene.

Tabla 28: Definición de operaciones de base de datos permitidas

Descripción	Nombre de Clase	Administrador de Proyecto				Recurso de Proyecto			
		Crear	Actualizar	Leer	Borrar	Crear	Actualizar	Leer	Borrar
<b>Proyecto</b>	project.project	1	1	1	1	0	0	1	0
<b>Tarea</b>	project.task	1	1	1	1	0	1	1	0
<b>Recurso</b>	project.resource	1	1	1	1	0	0	1	0
<b>Calendario</b>	resource.calendar	1	1	1	1	0	0	1	0
<b>Registro de Horas en Tareas</b>	project.task.work	1	1	1	1	1	1	1	1
<b>Calendario x Recurso en Proyecto</b>	project.resource.assignment	1	1	1	1	0	0	1	0
<b>Unidad de Medida</b>	product.uom	1	1	1	1	0	0	1	0
<b>Compañía</b>	res.company	0	0	1	0	0	0	1	0
<b>Usuarios</b>	res.users	0	0	1	0	0	0	1	0
<b>Etapas de las Tareas</b>	project.task.type	1	1	1	1	0	0	1	0
<b>Etiquetas del proyecto</b>	project.category	1	1	1	1	0	0	1	0
<b>Delimitadores de</b>	file.encoding.text.delimiter	1	1	1	1	0	0	1	0

<b>Texto</b>									
<b>Delimitadores de Campo</b>	file.encoding.field.delimiter	1	1	1	1	0	0	1	0
<b>Juego de Caracteres</b>	file.encoding.character.set	1	1	1	1	0	0	1	0

Elaborado por: Los autores

## CAPÍTULO IV DISEÑO DEL SISTEMA

### 4.1 Diseño de la arquitectura del sistema

#### 4.1.1 Diseño Arquitectónico

La ilustración # 11 muestra de manera general la interacción entre los grandes componentes de este proyecto, que consisten en la base de datos, su comunicación con el servidor de aplicaciones, y su interface a través del portal web para el manejo de cadena crítica.

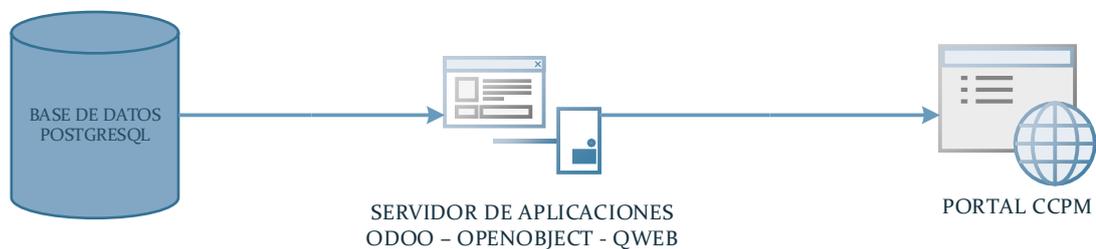


Ilustración 11: Modelo conceptual de la aplicación

Elaborado por: Los Autores

#### 4.1.1.1 Base de Datos

El motor de base de datos se encargará de la persistencia de los datos, en su versión 9.3 es la base de datos de código abierto más avanzada en la actualidad, odoo usa este motor únicamente para la persistencia, es decir de la estructuras de datos, para todas la tablas que se generan a través del servidor de aplicaciones, genera automáticamente los siguientes campos mostrador en la tabla # 29:

Tabla 29: Lista de Campos creados automáticamente por el framework de Odoo

Nombre	Tipo	Relación	Descripción
Id	Integer(secuencial automático)		Identificador de base de datos, de cualquier registro del sistema
create_uid	Integer(Foreign Key)	res_users	Usuario que realiza la creación del registro
create_date	Datetime		Fecha en la que se realizó la acción de creación

Nombre	Tipo	Relación	Descripción
write_uid	Integer(Foreign Key)	res_users	Usuario que realizó la última actualización del registro
write_date	Datetime		Fecha en la que se realizó la última actualización o escritura del registro
__last_update	Datetime		Este dato es usado para resolver problemas de concurrencia cuando 2 o más usuarios editan el mismo registro

Elaborado por: Los autores

#### 4.1.1.2 Servidor de Aplicaciones

Odoo es un motor de aplicaciones escrito en python, xml, javascript, que posee los siguientes componentes:

- ❖ **Core:** Es el encargado de la comunicación de datos con la base de datos, posee una implementación de O.R.M. propia, que posee métodos para efectuar las operaciones C.R.U.D., también se encarga de trabajar como servicio para los clientes a través del protocolo XML-RPC y XML-RPCS (Cifrado). También posee las siguientes características:
  - Gestor de tareas planificadas propio de la aplicación, multihilos
  - Motor de Traducciones, el sistema permite usar archivos de traducción, editables con el software de código abierto Poedit
  - Gestión de Usuarios, Perfiles/Grupos, reglas de acceso especificable por operación C.R.U.D., la asignación de permisos se realiza a los grupos, y estos se asignan a los usuarios del sistema
- ❖ **Cliente Web:** Este cliente ha sido la evolución más importante de Odoo en los últimos años, gracias a este la personalización de la interface con el usuario puede ser personalizada completamente, e integrar el uso de librerías JavaScript y tecnologías más actuales.
- ❖ **Addons:** Estos son módulos desarrollados para modelar procesos usados por las empresas de manera genérica, como el proceso de ventas, manejo de inventario, crm, compras, recursos humanos, contabilidad, gestión de proyectos, etc. Este

proyecto se basa específicamente en el módulo actual de este componente que se conoce como 'project' que es usado para la administración de proyectos.

#### **4.1.1.3 Importación de Archivos de Microsoft Project**

Para la implementación de esta funcionalidad en el sistema, se va a usar las siguientes librerías de java de código abierto, e implementador a través de un proyecto Maven:

- ❖ **Mpxj:** Es una librería encargada de interpretar archivos de Project con extensión mpp, xml
- ❖ **Commons-cli:** Es usada para poder permitir a la aplicación recibir parámetros a través de llamadas por consola
- ❖ **Super-csv :** Es usada para la elaboración de los archivo CSV
- ❖ **Log4j :** Es usada para hacer el log a travez del sistema operativo y asi poder ser tomado por la aplicación Python que ejecuta el proyecto

Este proyecto se compila como un único archivo .jar que es ejecutado a través de un wrapper Python que se encarga de realizar la ejecución a través de una llamada en consola.

#### **4.1.1.4 Cálculo de Cadena Crítica**

Debido a que el administrador de proyectos de Odoo, no posee actualmente un cálculo de ruta crítica necesario para obtener la información necesaria para el cálculo de la cadena crítica, amortiguadores y demás, es necesario usar una Libreria Python llamada Python-Graph, que posee implementado el cálculo de la ruta crítica a través de diágrafos los cuales son estructuras en las cuales se indican la etiqueta, el valor de duración, y las conexiones que existen entre ella o dependencias, de esta manera esta librería nos devuelve marcadas en un estructura de datos llamada diccionario a todas las tareas que pertenecen a la ruta crítica.

#### **4.1.1.5 Portal**

Usando la posibilidad de extender y modificar la interface de la plataforma a través del desarrollo de un módulo independiente, en el cual se ha integrado librerías de

JavaScript necesarias para agregar los reportes gráficos que se implementaron en este proyecto, entre estas se encuentran las librerías:

- ❖ Highcharts – Versión. 4.0.4
- ❖ Dhtmlxgantt -. Versión 3.0.

#### **4.1.2 Módulos del Sistema**

Este proyecto se ha construido con los siguientes componentes:

- ❖ Módulo Odoo: Siguiendo las instrucciones de desarrollo propuestas por la plataforma Odoo, se desarrolló un módulo que tiene dependencias al módulo llamado “project” que trae originalmente la plataforma, de tal manera que se cambió el comportamiento original del módulo. Entre los principales cambios que se realizó al módulo se detalla lo siguiente:
  - Proyectos:
    - ❖ El sistema originalmente trataba a los recursos de un proyecto como usuarios del sistema, por lo cual se creó una clase independiente, donde los recursos no necesariamente sean usuarios del sistema.
    - ❖ Se agregó por defecto la creación de los proyectos en estado de nuevo, para luego ser aprobados, originalmente el sistema creaba el proyecto en estado en progreso, obviando la fase de planificación que se necesita en cadena crítica.
    - ❖ Se agregó la especificación del calendario con el que trabajaran los recursos dentro del proyecto.
    - ❖ Se implementó la librería Python-graph para realizar el cálculo de la cadena crítica con las tareas dadas.
  - Tareas:
    - ❖ Se agregó la posibilidad de planificar las tareas en unidades de medida de tiempo configurables, por ejemplo días, horas, minutos, etc. Originalmente el sistema asumía que la duración de las tareas estaba expresada en horas.

- ❖ Se agregó la estructura adicional, para cumplir con el contexto de cadena crítica, en este caso poder identificar que son tareas críticas, grupos, hitos, información de amortiguadores, tipo de amortiguadores, esto se explica con detalle en el diccionario de datos ANEXO 1 – DICCIONARIO DE DATOS.

- ❖ Módulo Java:

- Para facilitar la adopción de la aplicación en Odoo, se optó por permitir la importación de 3 tipos de archivos:

- ❖ **Archivos Separados por Comas (CSV):** Odoo se encarga de interpretar el archivo csv a través de una librería nativa de python, correspondientes a los siguientes campos en la tabla # 30:

Tabla 30: Campos solicitados para importación de archivo csv

Nombre Campo	Tipo	Descripción
Id	Cadena de Texto	Para guardar mayor compatibilidad se espera que sea cualquier texto, pero en general se espera que sea un dígito, y que no se repita por proyecto
nombre_tarea	Cadena de Texto	Este texto es libre para describir que lo que se espera obtener o realizar con la tarea
Recurso	Cadena de Texto	Aquí se puede seleccionar uno o varios separados por comas
Tiempo	Cadena de Texto	El tiempo tiene que ser expresado en el formato número y añadido la unidad de medida del tiempo que puede ser: m ó M: para minutos h ó H: para horas d ó D: para Días
dependencias	Cadena de Texto	Aquí se debe poner el listado de dependencias, debe ser el id de la tarea, en caso de ser más de una debe estar separada

		por comas
--	--	-----------

Elaborado por: Los autores

- ❖ **Archivos de Microsoft Project (MPP, XML):** Para la interpretación de los archivos de Ms Project se usó la librería Mpxj la cual provee soporte para realizar la lectura o escritura los archivos de project, la idea general es que desde Odoos se solicite el archivo para la importación de datos, se especifique el formato a importar, se escriba de manera temporal el archivo en el sistema de archivos, el archivo jar lo reciba como un parámetro de ejecución, otro parámetro de ejecución es el tipo, y el archivo de salida, de tal forma que desde Python se envíe la ejecución del archivo jar con los parámetros adecuados, y de esta manera el módulo java se encarga de entregarme un archivo csv que puedo interpretar tal como lo haría con el asistente anteriormente mencionado.

## 4.2 Diseño de las Interfaces

**Pantalla de Inicio del Sistema:** en esta se realizará el acceso por usuario y contraseña asignados previamente como se puede ver en la ilustración #12.



Ilustración 12: Pantalla de Acceso Odoo

Elaborado por: Los autores

**Interfaz general del Sistema:** Esta es la pantalla general que se le muestra al usuario una vez que ingresa al sistema.

**Menús del Sistema:** Este es el listado de Menús que va a poseer el sistema de administración de proyectos.

**Pantalla general de búsqueda del Sistema:** Odoo posee una definición general de vistas, en las cuales permite tener una para realizar búsquedas de registros, y sobre la cual también se pueden aplicar varios filtros, y esquemas de agrupación, y sobre todo permite tener búsquedas avanzadas, es decir por cualquier campo de la tabla sobre la cual se consulta, como podemos ver en la ilustración # 13.



Grupo	Nombre del proyecto	Responsable de proyecto	Contacto	Tiempo estimado	Tiempo total	Tiempo dedicado	Progreso	Estado
▼ Administrator (2)				00:00	00:00	00:00	<div style="width: 0%;"></div>	
<input type="checkbox"/>	test	Administrator		00:00	00:00	00:00	<div style="width: 0%;"></div>	Nueva
<input type="checkbox"/>	Proyecto UPS	Administrator		00:00	00:00	00:00	<div style="width: 0%;"></div>	Nueva

Ilustración 13: Pantalla general de búsqueda del sistema

Elaborado por: Los Autores

**Asistente para importación de proyectos:** En este asistente se permitirá escoger el archivo a importar, el administrador del proyecto, la fecha de inicio, las especificaciones para cuando es un archivo csv, luego de importar el archivo deberá presentar un resumen de la importación, de la forma en que se presenta en la ilustración #14.

**Importar Proyecto**

**Información del Archivo**

Nombre del Proyecto:  Responsable del Proyecto:

Fecha de Inicio del Proyecto: 08/03/2015 Tipo de Archivo: CSV(Separado por Comas)

**Configuración del Archivo**

Codificación: utf-8 Delimitador de Texto: " Delimitador de Campo: ,

**Archivo a Importar**

Archivo Binario:

o

Ilustración 14: Asistente de Importación de Archivo CSV

Elaborado por: Los autores

La ilustración # 15 muestra el cambio que debe tener la pantalla cuando se trata de una importación de un archivo de MS Project, sea en formato mpp o xml, en este caso no es necesario solicitar al usuario la configuración de delimitadores del archivo CSV.

**Importar Proyecto**

**Información del Archivo**

Nombre del Proyecto:  Responsable del Proyecto:

Fecha de Inicio del Proyecto: 08/03/2015 Tipo de Archivo: MS Project File

**Configuración del Archivo**

**Archivo a Importar**

Archivo Binario:

o

Ilustración 15: Asistente de Importación de Archivos de Project MPP, y XML

Elaborado por: Los autores

La ilustración # 16 muestra el resumen de los datos leídos, que debe generarse, tal como se muestra los datos principales que servirán para crear junto con el proyecto.

### Información del Archivo

Nombre del Proyecto | Proyecto UPS    Responsable del Proyecto | Administrator    Fecha de Inicio del Proyecto | 08/03/2015  
 Archivo Binario | [Descargar](#)

### Datos Importados

ID	Nombre Tarea	Resource Names	Tiempo de Trabajo	UdM Tiempo	Dependencias	Observaciones
1	TAREA A	CHRISTOPHER	2,0000	Día(s)		
2	TAREA B	MERCEDES	1,0000	Día(s)	1	
3	TAREA C	MERCEDES	4,0000	Día(s)	2	
4	TAREA D	MERCEDES	2,0000	Día(s)	3	
5	TAREA E	CHRISTOPHER	3,0000	Día(s)	4	
6	TAREA F	CHRISTOPHER	1,0000	Día(s)	5	
7	TAREA G	CHRISTOPHER	4,0000	Día(s)	4, 5	
8	TAREA H	MERCEDES	5,0000	Día(s)	7	
9	TAREA I	CHRISTOPHER	1,0000	Día(s)	8	

[Añadir un elemento](#)

[Crear Proyecto](#) o [Cerrar](#)

Ilustración 16: Asistente de importación de archivo, resumen de importación

Elaborado por: Los autores

**Pantalla de Creación de Proyectos:** En esta pantalla se puede crear los nuevos proyectos, o el asistente de importación también lleva a esta pantalla, como se puede apreciar en la ilustración #17.

Nombre del proyecto  
Proyecto UPS

Use Tasks     Partes de horas

Responsable de proyecto | Administrator ▼

Customer |   ▼

9 Tareas    0 Documentos    Partes de horas

Para facturar o configurar la facturación y las opciones de renovación, ir al contrato relacionado: [Projects / Proyecto UPS](#).

Equipo    Resource's Calendar    Otra información    Etapas para tareas del proyecto

Resource	Calendar	Critical?
CHRISTOPHER	Calendar General	<input type="checkbox"/>
MERCEDES	Calendar General	<input type="checkbox"/>

[Añadir un elemento](#)

Ilustración 17: Pantalla de creación de proyectos

Elaborado por: Los autores

**Pantalla de Edición/Creación de Tareas:** En esta pantalla se puede realizar el ingreso o modificación de los registros de tareas, y también de sus respectivos registros de avances, tal como se muestra en la ilustración #18.

**TAREA A**

Proyecto: Proyecto UPS  
 Revisor: Administrator  
 Tiempo Planificado: 2,000  
 Fecha límite:   
 Etiquetas:   
 UdM de Tiempo: Día(s)

Descripción: Añadir una descripción...

Resumen del trabajo	Tiempo Trabajado	UdM de Tiempo	Fecha	Realizado por
TEST	1,000	Día(s)	15/03/2015 16:40:03	Administrator

Añadir un elemento

Ilustración 18: Pantalla de Ingreso/Edición de Tareas

Elaborado por: Los autores

**Vista de Gantt de Tareas:** En esta vista se podrá ver gráficamente la duración de las tareas y sus respectivas dependencias, como se puede ver en la ilustración #19.

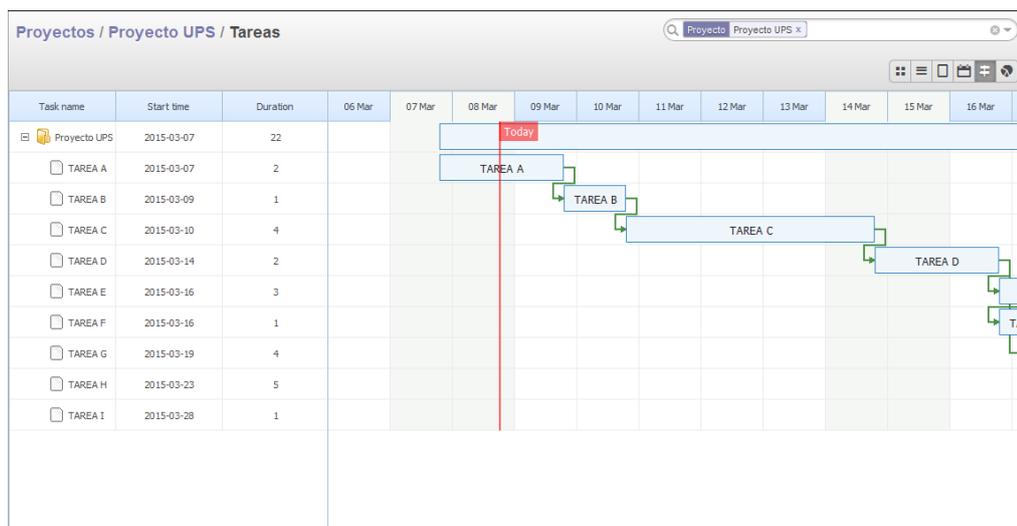


Ilustración 19: Vista de Gantt de Tareas

Elaborado por: Los autores

**Pantalla de Edición/Creación de Recursos:** En esta pantalla se puede hacer el ingreso de los recursos a usar en un proyecto, o también se puede editar los que se importan con el asistente de importación, tal como se visualiza en la ilustración #20.

Nombre: **CHRISTOPHER**

Código:  Correo Electrónico:

Calendario:  Usuario del sistema:

Activo?

Proyectos Actuales | Historial de Tareas

Nombre del proyecto	Responsable de proyecto	Contacto	Tiempo estimado	Tiempo total	Tiempo dedicado	Progreso	Estado
Projecto UPS	Administrator		00:00	24:00	24:00	<div style="width: 100%; height: 10px; background-color: green;"></div>	En progreso

Ilustración 20: Pantalla de Ingreso/Edición de Recursos

Elaborado por: Los autores

**Pantalla de Edición/Creación de Calendarios:** En este se puede especificar las horas disponibles por día, para posteriormente asignarse en los proyectos o recursos directamente, como se puede ver en la ilustración #21.

Nombre: **Calendar General**

Responsable del grupo de trabajo: **Administrator**

Días de Trabajo | Días Feriados/Libres

Nombre	Día de la semana	Trabajar desde	Trabajar hasta	Fecha de inicio
Monday	Lunes	08:00	17:00	
Tuesday	Martes	08:00	17:00	
Wednesday	Miércoles	08:00	17:00	
Thursday	Jueves	08:00	17:00	
Friday	Viernes	08:00	17:00	

[Añadir un elemento](#)

Ilustración 21: Pantalla de Edición/Creación de Calendarios

Elaborado por: Los autores

**Reporte de Penetración de Amortiguadores:** En este reporte se mostrará el cálculo de la penetración de amortiguadores, que será calculado por proyecto, como se muestra en el modelado de la ilustración #22.

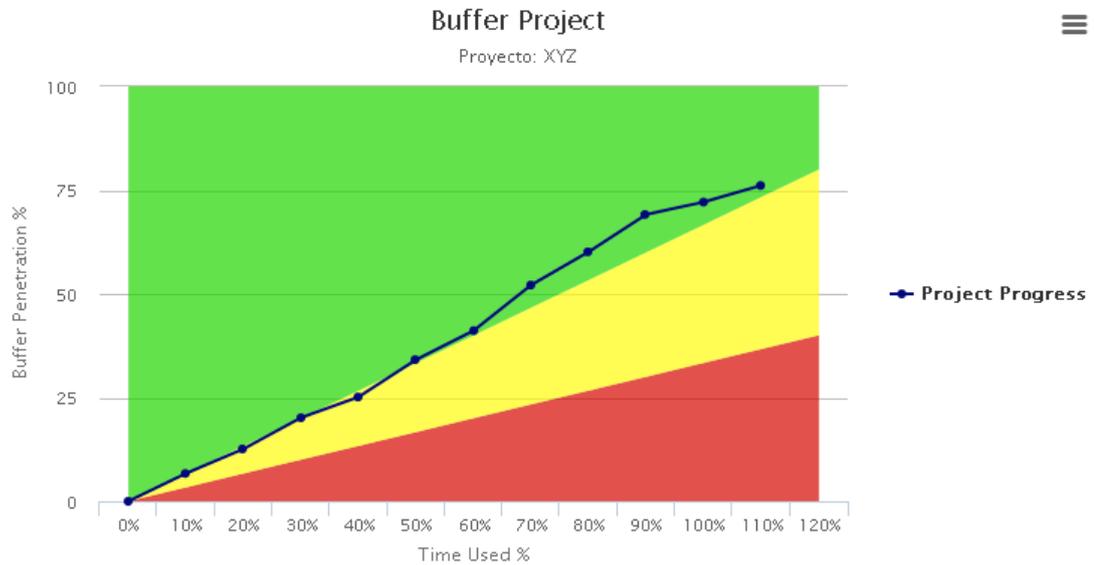


Ilustración 22: Reporte de Penetración de Amortiguadores

Elaborado por: Los autores

### 4.3 Diagrama de Clases del Sistema

Gracias al uso de la herramienta oerplib, se realizó la gráfica de las clases y sus relaciones, el siguiente es el listado de clases usadas para este proyecto en la componente parte Lógica de la aplicación:

- ❖ project.category

project.category	
[Attributes]	
- __last_update	datetime
- create_date	datetime
- display_name	char
- id	integer
- name [R]	char
- write_date	datetime

Ilustración 23: Diagrama de Clase project.category

Elaborado por: Los Autores

❖ project.config.settings

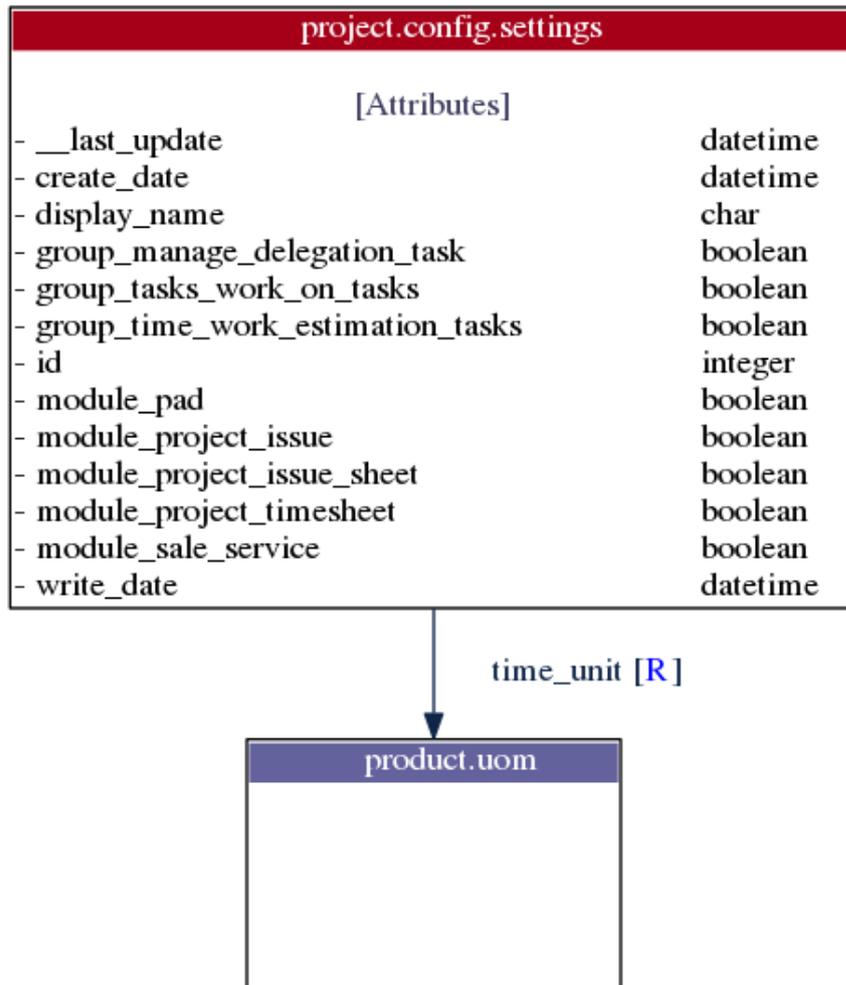


Ilustración 24: Diagrama de Clase project.config.settings

Elaborado por: Los Autores

❖ project.project

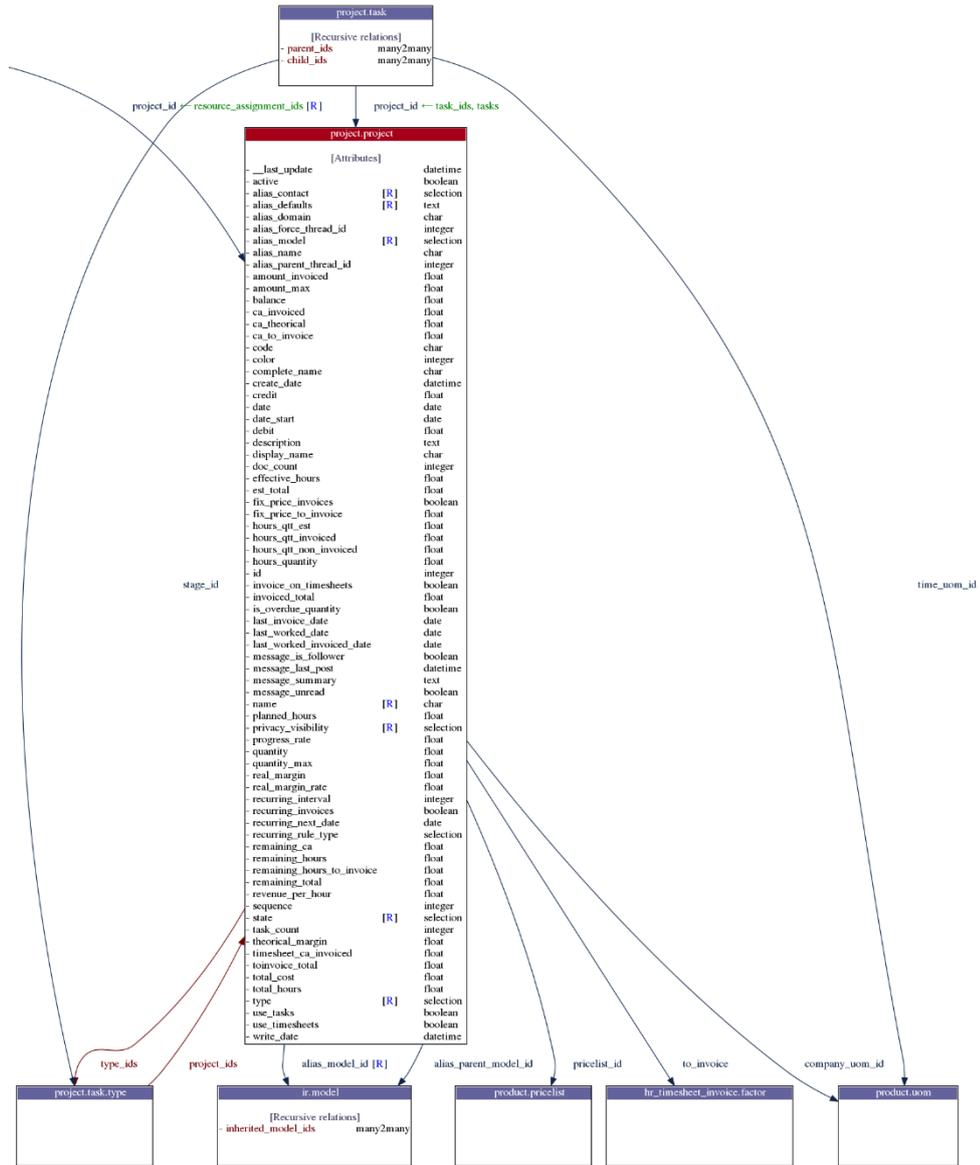


Ilustración 25: Diagrama de Clase project.project

Elaborado por: Los Autores

❖ project.resource

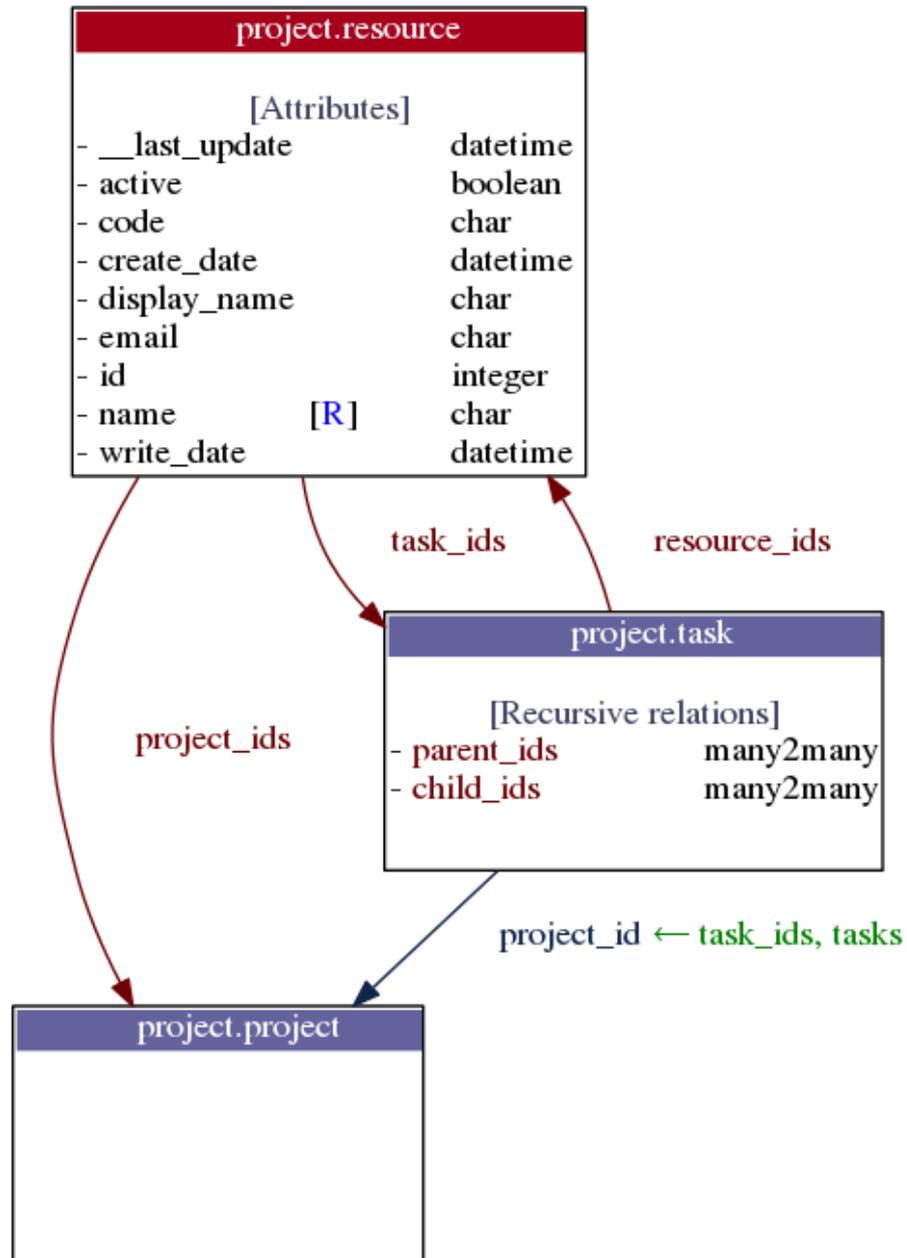


Ilustración 26: Diagrama de Clase project.resource

Elaborado por: Los Autores

❖ project.resource.assignment

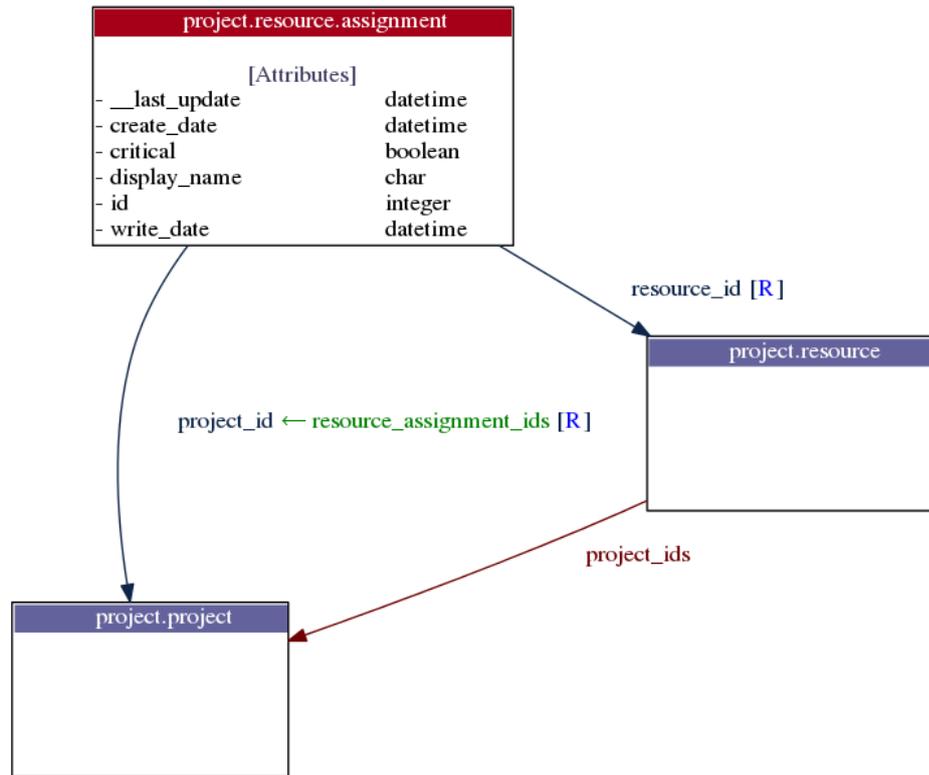


Ilustración 27: Diagrama de Clase project.resource.assignment

Elaborado por: Los Autores

❖ project.resource.calendar.holiday

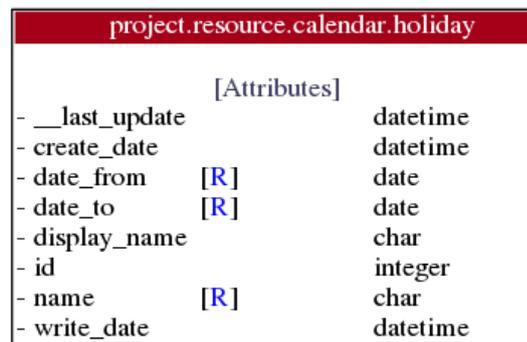


Ilustración 28: Diagrama de Clase project.resource.calendar.holiday

Elaborado por: Los Autores

❖ project.task

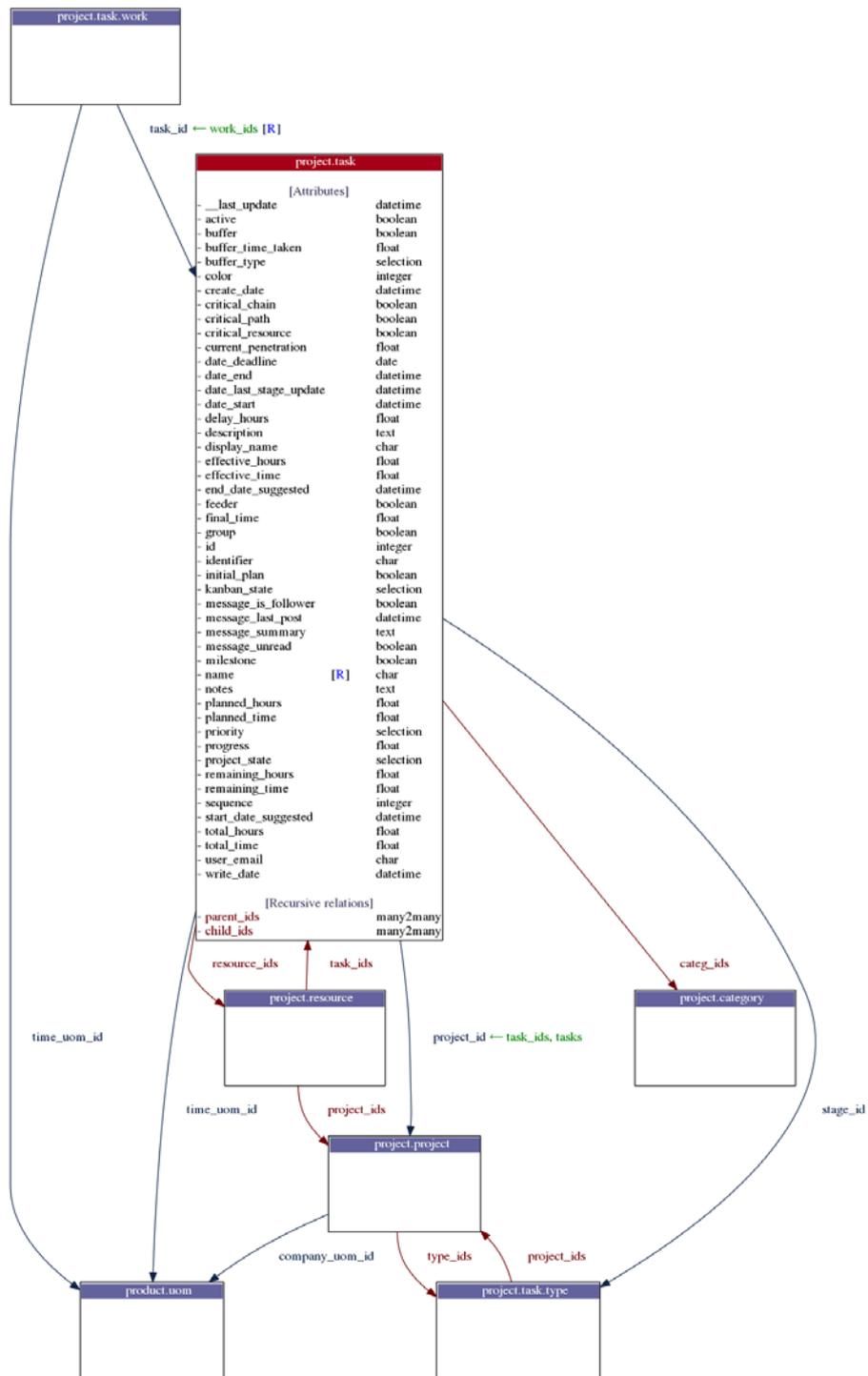


Ilustración 29: Diagrama de Clase project.task

Elaborado por: Los Autores

❖ project.task.delegate

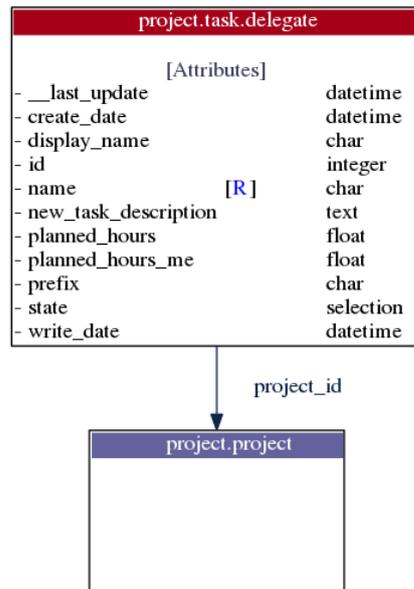


Ilustración 30: Diagrama de Clase project.task.delegate

Elaborado por: Los Autores

❖ project.task.history

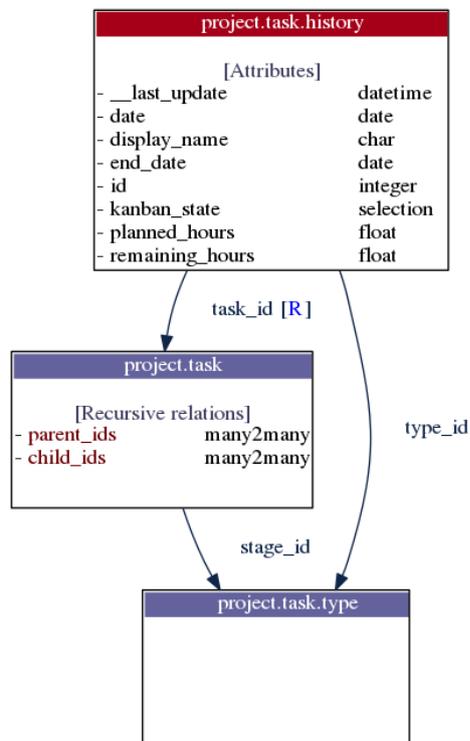


Ilustración 31: Diagrama de Clase project.task.history

Elaborado por: Los Autores

❖ project.task.history.cumulative

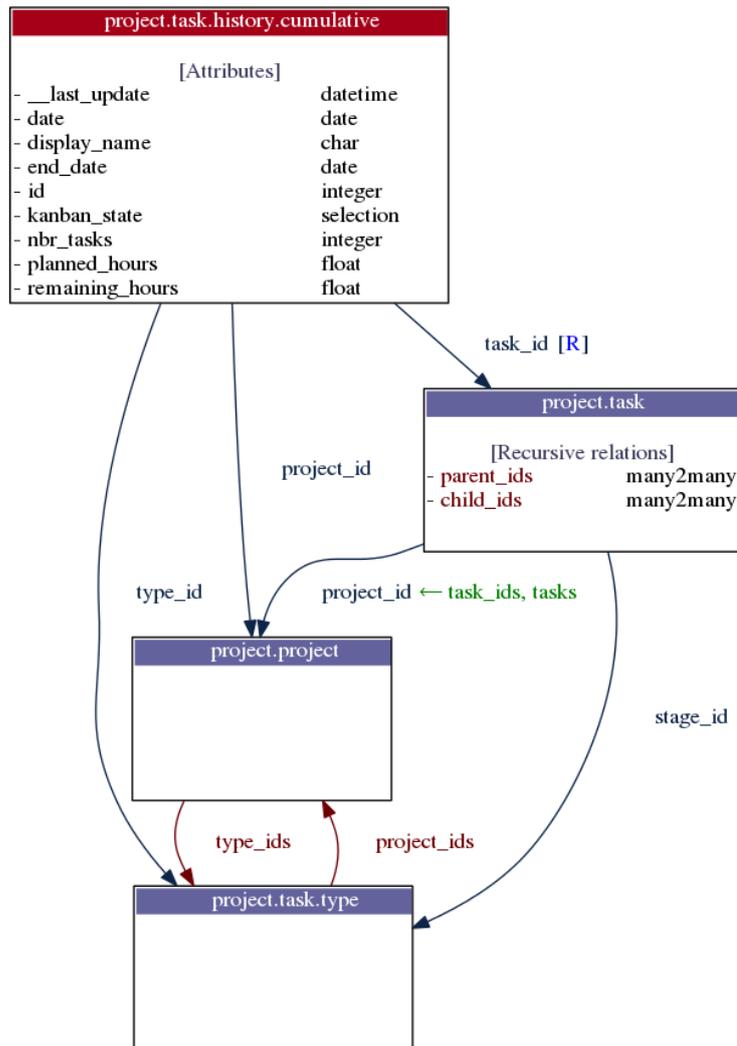


Ilustración 32: Diagrama de Clase `project.task.history.cumulative`

Elaborado por: Los Autores

❖ project.task.type

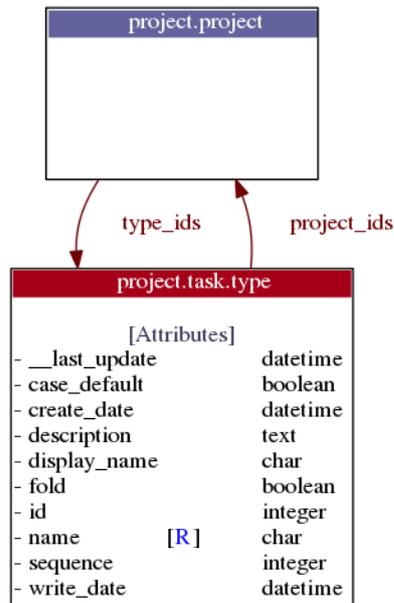


Ilustración 33: Diagrama de Clase project.task.type

Elaborado por: Los Autores

❖ project.task.work

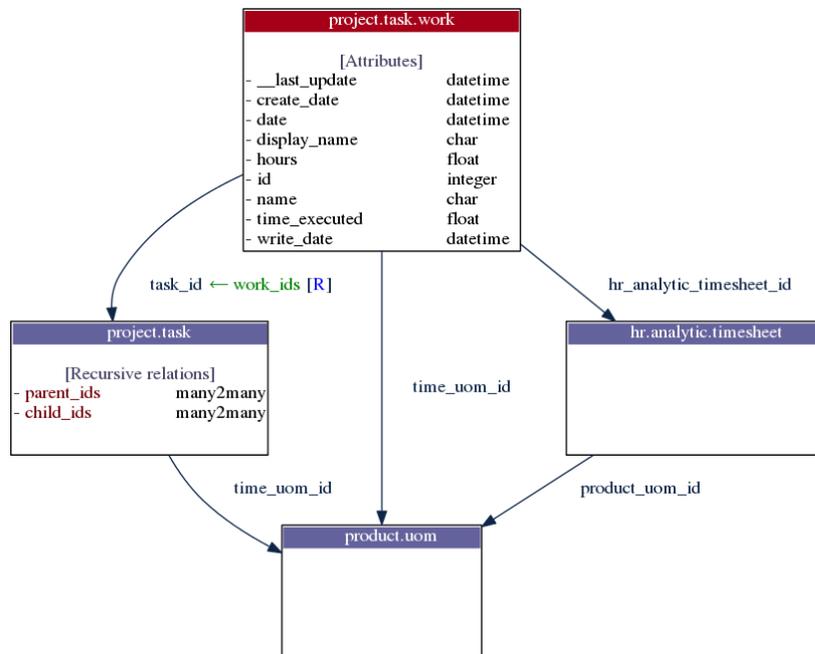


Ilustración 34: Diagrama de Clase project.task.work

Elaborado por: Los Autores

❖ report.project.task.user

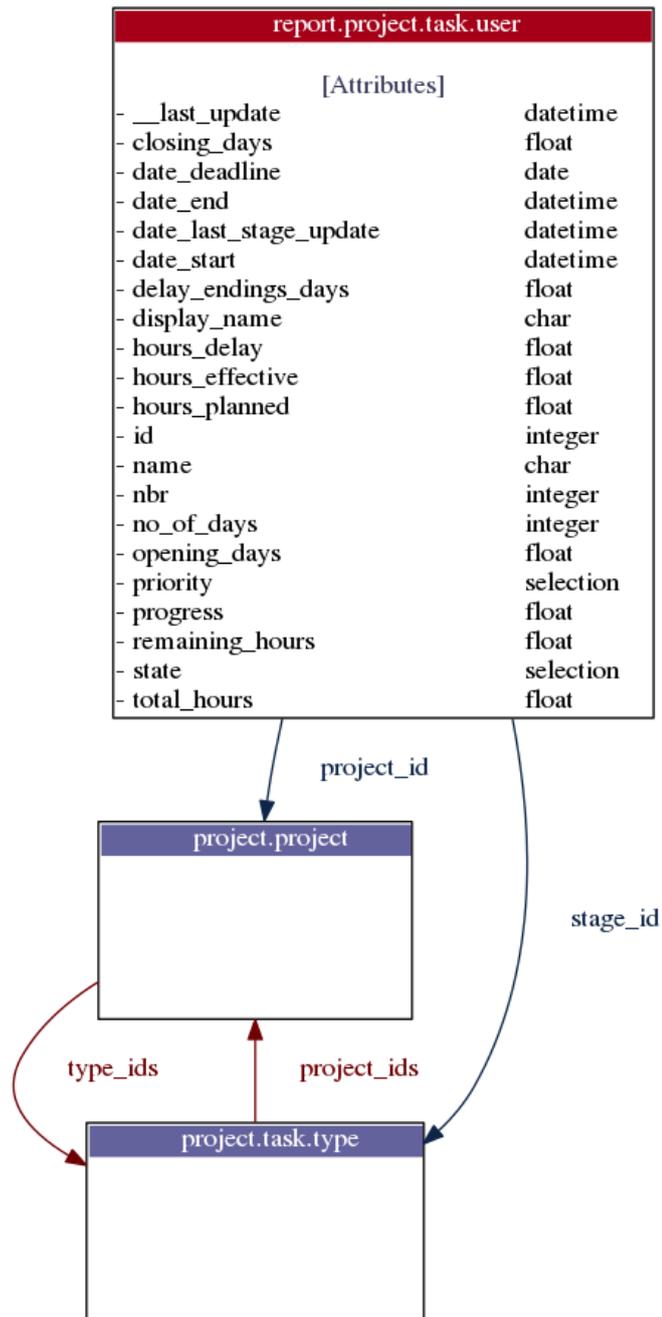


Ilustración 35: Diagrama de Clase report.project.task.user

Elaborado por: Los Autores

❖ wizard.import.project.line

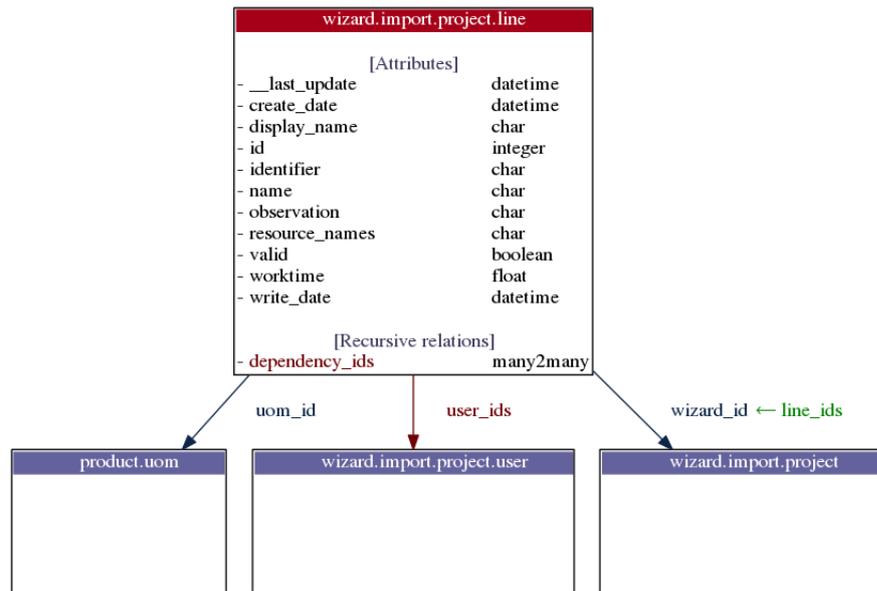


Ilustración 36: Diagrama de Clase wizard.import.project.line

Elaborado por: Los Autores

❖ wizard.import.project.user

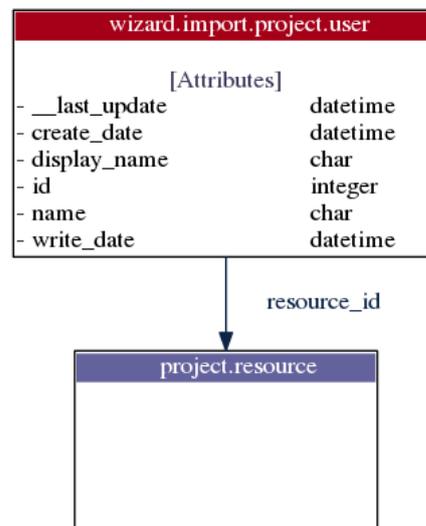


Ilustración 37: Diagrama de Clase wizard.import.project.user

Elaborado por: Los Autores

## 4.4 Modelo Lógico de la Base de Datos

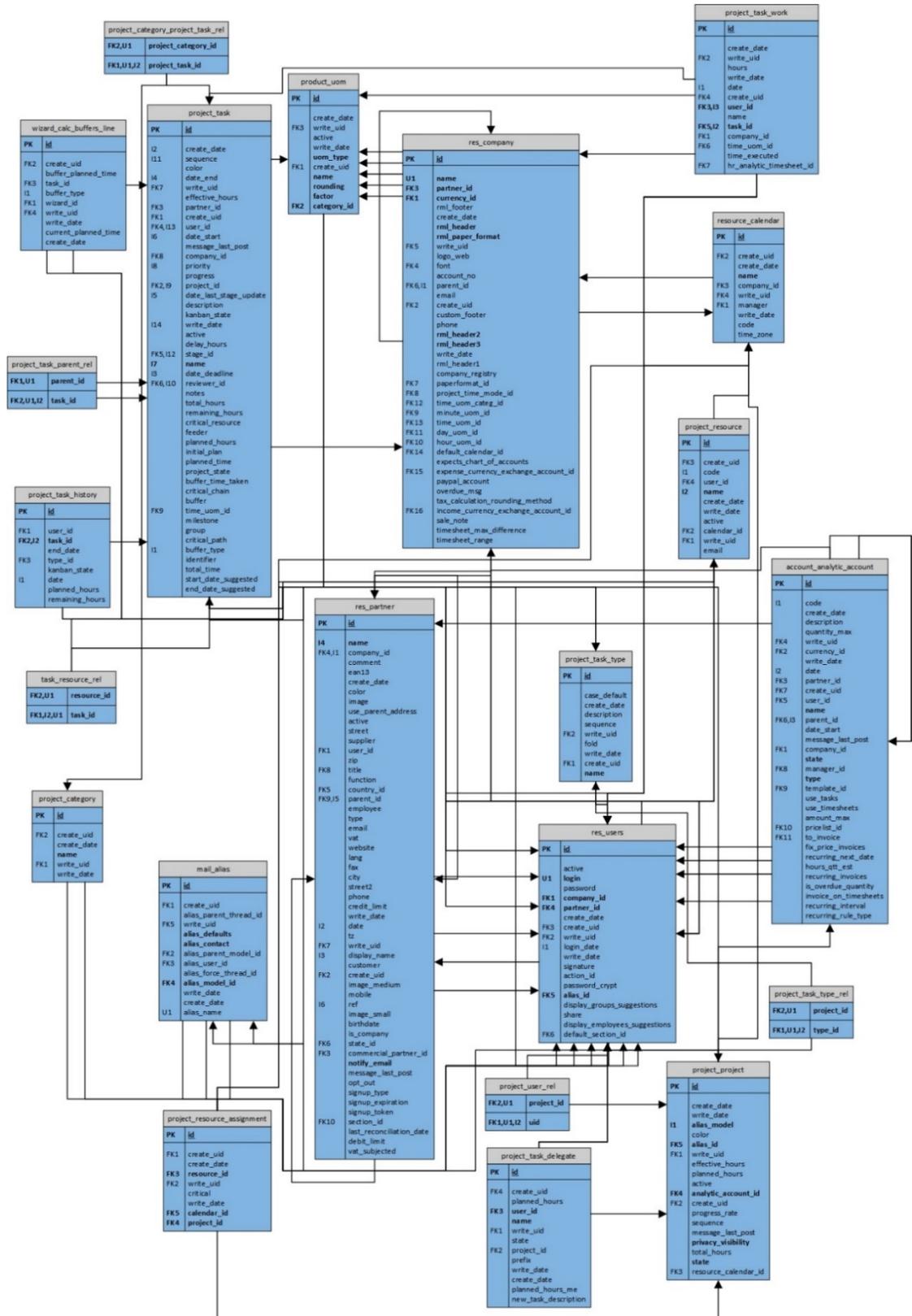


Ilustración 38: Diagrama Entidad Relación

Elaborado por: Los Autores

## CAPÍTULO V

### IMPLEMENTACIÓN Y PRUEBAS

#### 5.1 Capas del Sistema y Comunicación entre Capas

Como se puede ver en la ilustración # 39, el sistema se ha dividido en 4 capas, dentro de sí mismas también se tiene comunicación entre distintos lenguajes de programación.

**Base de Datos:** Los datos son guardados en el motor de base de datos Postgresql, solo se usa para operaciones CRUD, creación de vistas.

**Modelos:** Odoo posee un framework llamado openobject el cual posee una implementación de ORM que permite que la programación sea más sencilla y entendible, aquí se definen las clases que openobject traduce en tablas generadas a nivel de base de datos, y el mismo se encarga de la interacción en las operaciones de consulta, creado, actualizado y borrado.

**Lógica de Negocios:** En esta capa se programó los cálculos correspondientes a cadena crítica, así como las reglas generales de la aplicación, también se encuentra una comunicación a nivel de Shell con un módulo desarrollado en java, que permite la lectura de los archivos de Project con extensiones en XML, o MPP, esta comunicación puede ser apreciada en la ilustración #40.

**Cliente Web:** Odoo posee un framework llamado qweb con el cual se pueden definir dentro de la aplicación sus propias interfaces a través de JavaScript puro o como es en este caso con el uso de librerías de código abierto para la creación de gráficas de Gantt y los reportes de penetración de amortiguadores.

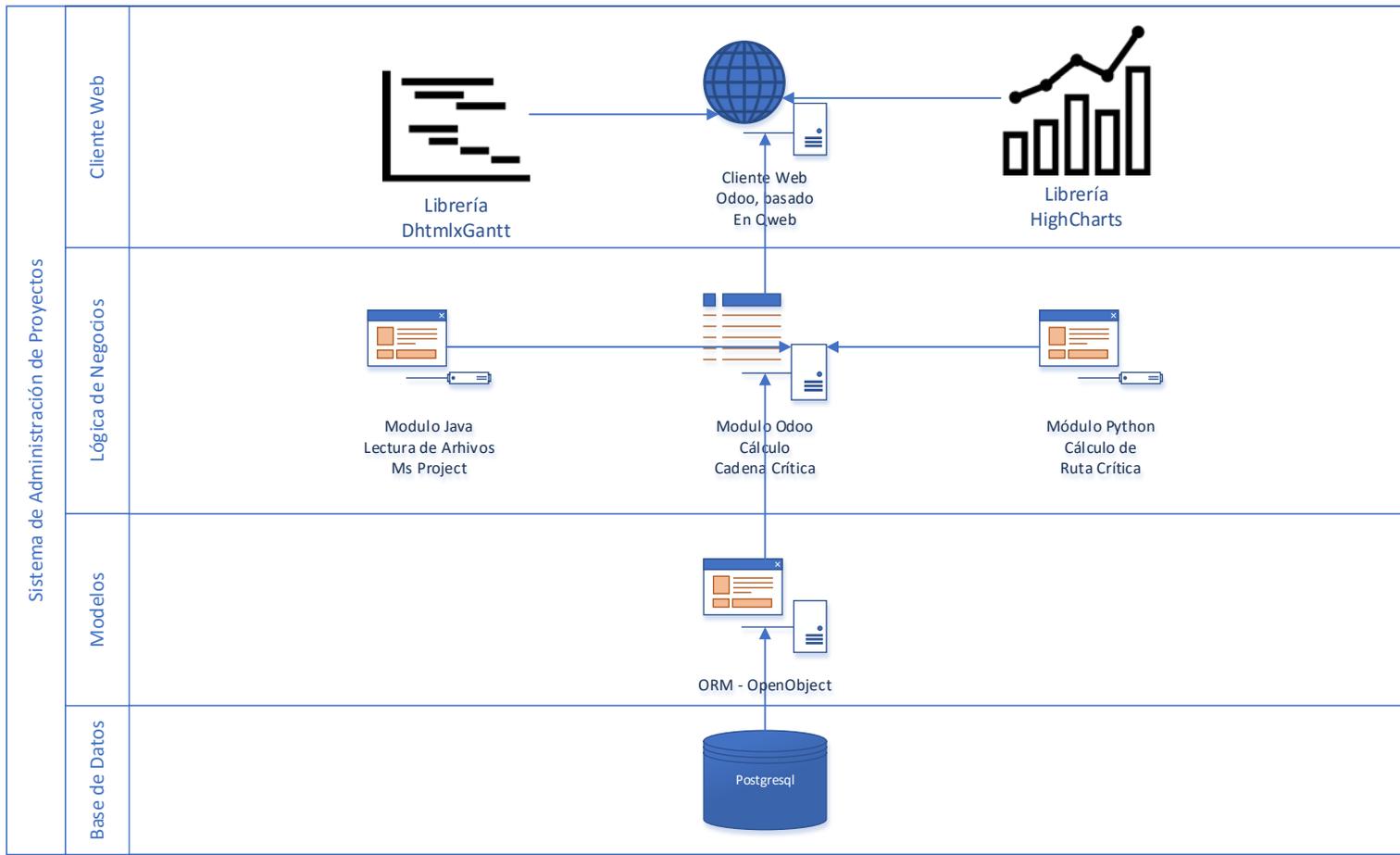


Ilustración 39: Arquitectura General del Sistema

Elaborado por: Los Autores

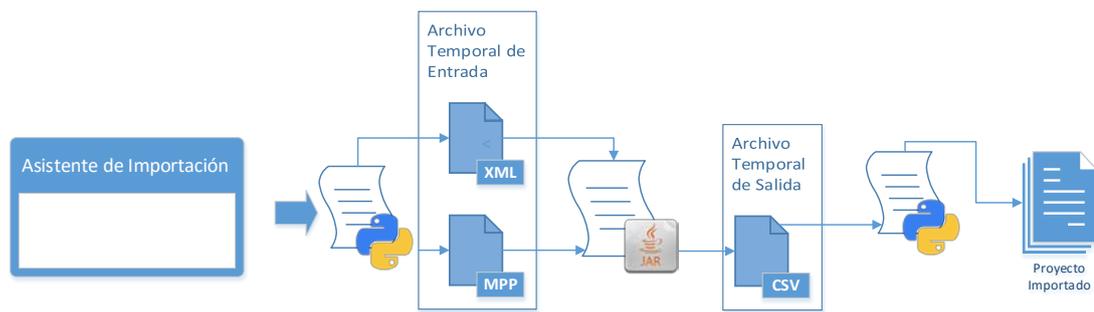


Ilustración 40: Modelo Conceptual de Comunicación entre Odoo y módulo Java

Elaborado por: Los Autores

## 5.2 Plan de Pruebas

Para este proyecto dado que el desarrollo trata de un módulo independiente que se integra en una plataforma preexistente, es necesario hacer pruebas de tipo unitarias y de integración, para evaluar el impacto que tiene el módulo desarrollado en la plataforma.

Como principal objetivo de estas pruebas es verificar la estabilidad de la plataforma, luego de su desarrollo, su usabilidad y seguridad.

La tabla # 31 muestra el listado de pruebas a realizar en el módulo desarrollado

Tabla 31: Planificación de casos de prueba

Caso de prueba	Tipo	Descripción del caso
<b>Acceso por Credenciales</b>	Unitaria	Verificar que los usuarios tengan que autenticarse para acceder al sistema
<b>Importación del Proyecto</b>	Unitaria	Se debe hacer ejemplos con archivos de extensiones csv, mpp y xml
<b>Creación de equipo de proyecto</b>	Unitaria	Crear varios recursos para un proyecto
<b>Creación de un nuevo proyecto</b>	Unitaria	Crear un proyecto desde la plataforma, independiente de la importación
<b>Creación de tareas</b>	Unitaria	Crear de tareas
<b>Cálculo de Ruta</b>	Unitaria	Algoritmo implementado de ruta crítica

<b>Crítica</b>		
<b>Aprobación del proyecto</b>	Unitaria	Calculo de Cadena Crítica
<b>Registro de Avance en Tareas</b>	Unitaria	Registro de tiempo de tareas
<b>Reporte de Penetración de Amortiguadores</b>	Unitaria	Reporte gráfico de penetración de amortiguadores
<b>Cierre de Proyecto</b>	Unitaria	Marcar código como cierre
<b>Instalación de Modulo</b>	Integración	Verificar que el modulo sea compatible con una instalación estándar de Odoo
<b>Proceso Completo de Proyecto</b>	Integración	Verificar que los usuarios pueden realizar las operaciones completas de todas las tareas unitarias

Elaborado por: Los autores

Los responsables de las pruebas del software y su calificación en este proyecto son los autores, ya que la implementación sigue lineamientos generales con respecto a la teoría de restricciones.

Para estas pruebas se realizó una instalación en un VPS en el cual se instaló el software y se accedía a través de su IP pública.

Se establece como criterios de aceptación para las pruebas:

- ❖ Rendimiento
- ❖ Estabilidad

### 5.3 Resultado de pruebas y métricas tomadas

Como métricas iniciales, se realizó la evaluación de los tiempos de respuesta en la importación de los archivos de proyecto, para esto se realizó la prueba con los siguientes tipos de archivos, en cada uno de ellos se indican proyectos con sus respectivos recursos y dependencias.

El entorno de red es local, es decir estas pruebas se realizaron con el cliente, el servidor y la base de datos dentro del mismo equipo, estas medidas no incluyen latencias de red.

La tabla # 32 muestra los tiempos de respuesta tomados en la ejecución de la aplicación y pruebas de importación con diferentes tipos de archivos y cantidad de tareas.

Tabla 32: Archivos usados en pruebas de importación

<b>Formato del Archivo</b>	<b>Cantidad de Tareas</b>	<b>Peso en Bytes</b>	<b>Tiempo de Respuesta(ms)</b>
Csv	10	412	1434
Csv	100	3687	3252
Csv	250	9318	8664
Ms Project .mpp	10	8901	2495
Ms Project .mpp	100	880640	7264
Ms Project .mpp	250	2196800	25019
Ms Project .xml	10	5443	2292
Ms Project .xml	100	555784	7134
Ms Project .xml	250	1374731	23021

Elaborado por: Los autores

Las pruebas realizadas unitarias y de integración se resumen en la tabla # 33.

Tabla 33: Tabla resumen de resultado de prueba

<b>Caso de prueba</b>	<b>Resultado Esperado</b>	<b>Resultado Obtenido</b>
<b>Acceso por Credenciales</b>	Acceso solo con credenciales	Se bloquea el usuario que no tiene credenciales

<b>Importación del Proyecto</b>	Importación de los 3 tipos de archivos	Importación correcta
<b>Creación de equipo de proyecto</b>	Creación de recursos y sus respectivos calendarios asignados con sus datos correctos	Creación satisfactoria
<b>Creación de un nuevo proyecto</b>	Creación de un proyecto, asignación de recursos y calendarios	Creación satisfactoria
<b>Creación de tareas</b>	Creación de tareas con sus datos de tiempo y asociación a un proyecto	Creación satisfactoria
<b>Cálculo de Ruta Crítica</b>	Marcar las tareas como ruta crítica las tareas que calcule el algoritmo	Calculo satisfactorio
<b>Aprobación del proyecto</b>	Calculo de cadena crítica, y cambio de estado del proyecto	Cálculo satisfactorio
<b>Registro de Avance en Tareas</b>	Registro del recurso y del administrador	Registro exitoso
<b>Reporte de Penetración de Amortiguadores</b>	Gráfica con puntos indicando la penetración del amortiguador	Gráfica con cálculos correctos
<b>Cierre de Proyecto</b>	Cambio de estado del proyecto y tareas abiertas	Proceso Correcto
<b>Instalación de Modulo</b>	Instalación de dependencias	Proceso Correcto
<b>Proceso Completo</b>	Administrador haga su proceso	Proceso correcto

<b>de Proyecto</b>	completo sobre proyectos importados	
--------------------	-------------------------------------	--

Elaborado por: Los autores

De los tipos de pruebas que se realizó se desprende la siguiente estadística del tipo de pruebas que se realizaron para este proyecto, como se puede evaluar en la ilustración # 41



Ilustración 41: Estadística de tipos de pruebas

Elaborado por: Los autores

El resultado de las pruebas se puede resumir como exitoso en función de que no se encontraron errores dentro de las pruebas de funcionalidades requeridas.



Ilustración 42: Estadística de Resultados de Pruebas

Elaborado por: Los Autores

## CAPÍTULO VI

### CONCLUSIONES Y RECOMENDACIONES

#### 6.1 Conclusión

Este proyecto ha tenido su grado de dificultad, en el aprendizaje de los cálculos pero que son necesarios para la correcta funcionalidad del módulo.

Al inicio de este estudio se determinó que existía cierta complejidad en lo que se trataba de lograr, fue necesario hallar la forma de sentar la bases para el cálculo de cadena crítica, de los cuales carecía Odoo, entre estos está el cálculo de la ruta crítica, la gráfica de Gantt que permita visualizar de manera correcta las dependencias de un proyecto, la capacidad para importar o interpretar archivos de Ms Project.

En este proyecto se debe recalcar que existen soluciones informáticas para problemas específicos, para este caso, cálculo de ruta crítica, gráficas de Gantt, lectura de archivos de Ms Project, archivos csv, sin embargo ha sido necesaria la interacción entre ellas y desarrollo de una aplicación específica, que combine todas estas tecnologías.

La conclusión más importante a la que se llegó es que no existe un lenguaje de programación que de la solución a todos los problemas, sin embargo tener una base sólida de la ingeniería de sistemas y su respectiva aplicación hace capaz de llevar a cabo este tipo de proyectos, acortando caminos al no tener que escribir desde cero la implementación de ciertas funcionalidades específicas que se necesitaron en este proyecto, y escritas en diferentes lenguajes.

La capacidad de interpretar archivos de Ms Project no es algo que se encuentra implementado en todos los lenguajes de programación y menos en forma de librerías de código abierto, fue complicado al principio tratar de interpretar directamente los archivos, sin embargo se encontró que una librería como MPXJ escrita en Java permitió el ahorro del largo camino que hubiera sido el tener que desarrollar esa funcionalidad desde cero.

Odoo posee un esquema de desarrollo rápido, que permite una fácil adaptación y modificación, es decir con un módulo y pocas líneas es posible modificar parcial o

completamente el comportamiento de un módulo determinado, en este proyecto esto facilitó la alteración y comportamiento de estructuras ya existente, como lo es el módulo de proyecto.

En la versión que se trabajó de Odoo incluye una capacidad de integración con librerías de JavaScript, lo cual hace que la interface que interactúa con el usuario sea más versátil y moderna, permite la implementación de proyectos independientes de JavaScript, lo cual permite el correcto graficado de reportes, y de proyectos en formato Gantt.

La fácil adopción que posee Odoo en la actualidad se debe en gran parte al uso de python, que es un lenguaje sencillo de entender y mantener, su sintaxis es limpia, y sobre todo el ordenamiento que poseen los módulos desarrollados en la plataforma son legibles, lo que posibilita a personas entusiastas aventurarse a desarrollar en esta plataforma que en realidad es bastante amplia, pero como demuestra este proyecto es posible simplificar y reutilizar solo el componente que fue necesario para cumplir el objetivo.

## **6.2 Recomendación**

Este proyecto puede ser base para que una comunidad grande como Odoo pueda integrarlo dentro de sus especializaciones, dando de esta manera una puerta de entrada para que ya sea entusiastas o empresas interesadas en términos comerciales puedan pulir y mejorar esta herramienta desarrollada.

Las características deseables que se espera de esta herramienta, se lista a continuación, para su profundización y desarrollo posterior

- ❖ Reportes relacionados con los avances individuales de las tareas de los proyectos
- ❖ Notificaciones automáticas de amortiguadores en rojo
- ❖ Reporte gráfico para portafolios de proyectos
- ❖ Restricción de multitasking al tener varios proyectos corriendo a la vez, es decir evitar que al momento de la planificación se asigne recursos de manera sobrepuesta en el tiempo

- ❖ Implementar librerías JavaScript como bootstrap 3 que permita la correcta visualización en dispositivos móviles como smartphones o tablets

## BIBLIOGRAFÍA

- ABF Osiell. (2014). *OerpLib Documentation*. Obtenido de Python Hosted:  
<http://pythonhosted.org/OERPLib/>
- Addison-Wesley Professional. (2000). *Java Language Specification, 2nd Edition* .  
Mountain View: Addison-Wesley.
- Aguilera C., C. I. (2007). Un enfoque gerencial de la teoría de restricciones. *Estudios Gerenciales*, 53-69.
- Barros, A. (05 de Enero de 2010). *Comportamiento de Proyectos TI. ¿Estan en Deuda!* Obtenido de El Escritorio de Alejandro BARros:  
<http://www.alejandrobarrros.com/content/view/691759/Comportamiento-de-proyectos-TI-Estan-en-deuda.html>
- Bloch, M., Blumberg, S., & Laartz, J. (Octubre de 2012). *Delivering large-scale IT projects on time, on budget, and on value*. Obtenido de McKinsey & Company Web Site:  
[http://www.mckinsey.com/insights/business\\_technology/delivering\\_large-scale\\_it\\_projects\\_on\\_time\\_on\\_budget\\_and\\_on\\_value](http://www.mckinsey.com/insights/business_technology/delivering_large-scale_it_projects_on_time_on_budget_and_on_value)
- EcuRed. (2014). *Python Software Foundation License - EcuRed*. Obtenido de Python Software Foundation License:  
[http://www.ecured.cu/index.php/Python\\_Software\\_Foundation\\_License](http://www.ecured.cu/index.php/Python_Software_Foundation_License)
- Eguiluz, J. (2013). *Breve historia(Introducción a JavaScript)*. Obtenido de Librosweb: [http://librosweb.es/libro/javascript/capitulo\\_1/breve\\_historia.html](http://librosweb.es/libro/javascript/capitulo_1/breve_historia.html)
- Free Software Foundation. (2015). *Lista de licencias con comentarios-Proyecto GNU-Free Software Foundation*. Obtenido de Licencias de software libre compatibles con la GPL: <https://www.gnu.org/licenses/license-list.es.html#GPLCompatibleLicenses>
- González Cruz, M. C., Asencio Cuesta, S., Diego Más, J. A., & Alcaide Marzal, J. (2009). Análisis del Método de Cadena Crítica vs Método del Camino Crítico. Viabilidad y Conceptos. *XII CONGRESO INTERNACIONAL DE*

*INGENIERÍA DE PROYECTOS* (págs. 56-67). Valencia, España: Universidad Politécnica de Valencia.

González Duque, R. (2010). *Python para Todos*. Obtenido de <http://mundogeek.net:https://launchpadlibrarian.net/18980633/Python%20para%20todos.pdf>

Gulla, J. (Febrero de 2012). *Seven Reasons IT Projects Fail*. Obtenido de IBM System Magazine: [http://www.ibmssystemsmag.com/power/Systems-Management/Workload-Management/project\\_pitfalls/](http://www.ibmssystemsmag.com/power/Systems-Management/Workload-Management/project_pitfalls/)

Kerzner, H. R. (2013). *Project management: a systems approach to planning, scheduling, and controlling*. John Wiley & Sons.

Lamarca Lapuente, M. (2013). *eXtensible Markup Language (XML)*. Obtenido de HIPERTEXTO: EL NUEVO CONCEPTO DE DOCUMENTO EN LA CULTURA DE LA IMAGEN : <http://www.hipertexto.info/documentos/xml.htm>

Martinez Guerrero, R. (02 de Octubre de 2010). *Sobre PostgreSQL*. Obtenido de PostgreSQL-es: [http://www.postgresql.org.es/sobre\\_postgresql](http://www.postgresql.org.es/sobre_postgresql)

Mozilla Developer Network. (2014). *JavaScript | MDN*. Obtenido de Mozilla Developer Network: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>

Mozilla Developer Network. (2014). *Xpath | MDN*. Obtenido de Mozilla Developer Network - MDN: <https://developer.mozilla.org/en-US/docs/Web/XPath>

O'Reilly Media, Inc. (s.f.). *What is XML?* Obtenido de XML.com: <http://www.xml.com/pub/a/98/10/guide0.html?page=2#AEN58>

OpenERP SA. (2014). *Odoo Open Source ERP and CRM*. Obtenido de [https://doc.odoo.com/7.0/book/1/1\\_1\\_Inst\\_Config/1\\_1\\_Inst\\_Config\\_architecture/](https://doc.odoo.com/7.0/book/1/1_1_Inst_Config/1_1_Inst_Config_architecture/)

Pérez de Eulate, U., & Oyarbide Zubillaga, A. (2005). La aportación de la "Cadena Crítica" frente a la gestión clásica de proyectos. *IX Congreso de Ingeniería de Organización*, (págs. 8-9). Gijón.

- Poggioli, P. (1976). *Aplicación Práctica del Método P.E.R.T.* Barcelona: Editores Técnicos Asociados S.A.
- Project Management Institute. (2000). *PMBOK, A Guide to the project Management body of knowledge.* Pennsylvania USA.
- Santiago, J., & Desirae, M. (2009). *Critical Path Method: Stanford University.*  
Obtenido de <http://web.stanford.edu/class/cee320/CEE320B/CPM.pdf>
- Santoyo González, F., Tenorio González, A. L., & Tenorio González, A. L. (2011).  
TÉCNICAS DE PLANEACIÓN Y CALENDARIZACIÓN PARA LA ADMINISTRACIÓN DE PROYECTOS EFICIENTES Y EFICACES.  
*Inceptum, Revista para la investigación en Ciencias de la Administración.*
- Tan, S. (27 de Enero de 2011). *How to Increase Your IT Project Success Rate.*  
Obtenido de Gartner Web Site: <https://www.gartner.com/doc/1531025>

### ANEXO 1: MATRIZ DE CAUSA Y EFECTO

<b>PROBLEMA GENERAL</b>	<b>PREGUNTAS GENERALES</b>	<b>OBJETIVO GENERAL</b>	<b>HIPOTESIS</b>	<b>VARIABLES DEPENDIENTES X</b>	<b>VARIABLES INDEPENDIENTES Y</b>
Al método cadena crítica para la administración de proyecto, le faltan herramientas informáticas accesibles para su implementación	¿Por qué en los proyectos no es viable aplicar cadena crítica para cumplir con los objetivos de terminar a tiempo, dentro del presupuesto esperado y con la calidad esperada?	Diseñar módulo para gestión de proyectos a través del método de cadena crítica en la plataforma de software libre Odoos, para facilitar la administración de los proyectos a través de este método de manera eficiente.	La elaboración de un módulo que implemente los algoritmos, reportes e indicadores propuestos por el método cadena crítica en una herramienta de código abierto, permitirá que se pueda tener el apoyo tecnológico adecuado para la implementación del método cadena crítica en la administración de proyectos	No es posible tener el apoyo tecnológico para usar cadena crítica en la gestión de proyectos	Falta de herramientas de código abierto que tengan implementado cadena crítica
<b>SUBPROBLEMAS</b>	<b>SISTEMATIZACION</b>	<b>OBJETIVOS ESPECIFICOS</b>	<b>HIPOTESIS</b>	<b>VARIABLES DEPENDIENTES X</b>	<b>VARIABLES INDEPENDIENTES Y</b>
No es posible determinar la cadena crítica en la fase de planificación del proyecto	¿Por qué no es posible determinar la cadena crítica en la fase de planificación del proyecto?	Desarrollar mecanismos para cálculo y sugerencia de amortiguadores para las distintas categorías que identifica la teoría, amortiguador de proyecto, amortiguadores de alimentación, y amortiguadores de recursos.	El desarrollo de los cálculo de amortiguadores, y de la cadena crítica, permitirá tener la información necesaria para realizar los indicadores propuestos por la teoría de cadena crítica	No es posible tener la información necesaria para realizar los indicadores propuestos por la teoría	La falta de cálculo de amortiguadores, y de la cadena crítica
No se puede tener información del estado del proyecto y de su avance según cadena crítica, los valores de penetración de amortiguadores	¿Por qué no es posible tener los reportes necesarios para el seguimiento del avance del proyecto que propone la cadena crítica para la medición?	Desarrollar los reportes necesarios para dar seguimiento a la penetración del amortiguador general del proyecto, y de las tareas específicas que posean amortiguadores de alimentación o de recursos.	Al elaborar los reportes necesario para establecer que tareas requieren atención adicional, se evitará su retraso y el de todo el proyecto	No es posible identificar las tareas del proyecto que necesitan atención	La falta de reportes no permite identificar que tareas del proyecto necesitan atención adicional

<p>No es posible identificar cuáles son las tareas que necesitan atención inmediata, y que ponen en riesgo el cumplimiento de los objetivos del proyecto</p>	<p>¿Por qué no es posible identificar cuáles son las tareas que tienen un riesgo alto de retrasar todo el proyecto en general?</p>	<p>Adicionar los esquemas de semáforos propuestos por las teorías en los reportes de penetración de amortiguadores, y gráficas Gantt del proyecto.</p>	<p>Al desarrollar el mecanismo para cálculo del indicador de penetración de amortiguadores de las tareas del proyecto, será posible evaluar y actuar para mitigar su riesgo</p>	<p>No es posible evaluar y actuar para mitigar su riesgo</p>	<p>Al no poseer un mecanismo de cálculo del indicador de penetración de amortiguadores de las tareas del proyecto</p>
<p>La adopción de herramientas de código para administración de proyectos, posee poca aceptación debido a la difusión y uso de herramientas comerciales como Microsoft Project</p>	<p>¿Por qué poseen poca aceptación las herramientas de código abierto para la administración de proyectos?</p>	<p>Construir un mecanismo para importar proyectos desde Microsoft Project (Archivos XML) y archivos separados por comas (.csv) en el gestor de proyectos de Odo, que permitirá migrar proyectos actualmente gestionados a través de estas herramientas.</p>	<p>Al desarrollar un mecanismo para importar proyectos desde Microsoft Project (Archivos XML) y archivos separados por comas (.csv) permitirá de compatibilidad de adaptación con las herramientas de código abierto</p>	<p>Existe una lenta adaptación con las herramientas de código abierto para administración de proyectos</p>	<p>La falta de compatibilidad con las herramientas mas utilizadas de administración de proyectos</p>

## ANEXO 2: DICCIONARIO DE DATOS

TABLA	MAIL_ALIAS		
Nombre_Campo	Tipo	Requerido	Descripción
alias_contact	Varchar	NO	Contacto
alias_defaults	Text	NO	Valor por defecto
alias_force_thread_id	integer	SI	Record ID
alias_model_id	integer	NO	Id del Modelo
alias_name	varchar	SI	Nombre de alias
alias_parent_model_id	Integer	SI	Modelo padre
alias_parent_thread_id	Integer	SI	Padre ID
alias_user_id	integer	SI	Id Usuario

TABLA	PRODUCT_UOM		
Nombre_Campo	Tipo	Requerido	Descripción
active	boolean	SI	Activo
category_id	integer	NO	Categoría de Producto
factor	numeric	NO	Factor
name	varchar	NO	Unidad de medida
rounding	numeric	NO	Precisión de redondeo
uom_type	varchar	NO	Tipo

TABLA	PROJECT_ACCOUNT_ANALYTIC_LINE		
Nombre_Campo	Tipo	Requerido	Descripción
from_date	date	SI	Fecha inicial
to_date	date	SI	Fecha final

TABLA	PROJECT_CATEGORY		
Nombre_Campo	Tipo	Requerido	Descripción
name	varchar	NO	Nombre

TABLA	PROJECT_ACCOUNT_ANALYTIC_LINE		
Nombre_Campo	Tipo	Requerido	Descripción
group_manage_delegation_task	boolean	SI	Tarea delegada
group_tasks_work_on_tasks	boolean	SI	Log de actividades realizadas
group_time_work_estimation_tasks	boolean	SI	Manage time estimation on tasks
module_pad	boolean	SI	Módulo

module_project_issue	boolean	SI	Asunto
module_project_issue_sheet	boolean	SI	Invoice working time
module_project_timesheet	boolean	SI	Record timesheet lines per tasks
module_sale_service	boolean	SI	Generate tasks from sale orders
time_unit	integer	NO	Working time unit

<b>TABLA</b>	<b>PROJECT_PROJECT</b>		
<b>Nombre_Campo</b>	<b>Tipo</b>	<b>Requerido</b>	<b>Descripción</b>
active	boolean	SI	Activo
alias_id	integer	NO	Alias
alias_model	varchar	NO	Alias Model
analytic_account_id	integer	NO	Contract/Analytic
color	integer	SI	Color Index
effective_hours	numeric	SI	Time Spent
message_last_post	timestamp without time zone	SI	Last Message Date
planned_hours	numeric	SI	Tiempo planificado
privacy_visibility	varchar	NO	Visibilidad Privada
progress_rate	numeric	SI	Progreso
resource_calendar_id	integer	SI	Tiempo laborable
sequence	integer	SI	Secuencia
state	varchar	NO	Estatus
total_hours	numeric	SI	Tiempo Total

<b>TABLA</b>	<b>PROJECT_RESOURCE</b>		
<b>Nombre_Campo</b>	<b>Tipo</b>	<b>Requerido</b>	<b>Descripción</b>
active	boolean	SI	Activo
calendar_id	integer	SI	Calendario
code	varchar	SI	Código
email	varchar	SI	Correo
name	varchar	NO	Nombre
user_id	integer	SI	Usuario System

<b>TABLA</b>	<b>PROJECT_RESOURCE_ASSIGNMENT</b>		
<b>Nombre_Campo</b>	<b>Tipo</b>	<b>Requerido</b>	<b>Descripción</b>
calendar_id	integer	NO	Calendario
Critical	boolean	SI	Crítico
project_id	integer	NO	Proyecto
resource_id	integer	NO	Recurso

<b>TABLA</b>	<b>PROJECT_RESOURCE_CALENDAR_HOLIDAY</b>		
<b>Nombre_Campo</b>	<b>Tipo</b>	<b>Requerido</b>	<b>Descripcion</b>
calendar_id	integer	NO	Calendario
date_from	date	NO	Fecha Inicio
date_to	date	NO	Fecha Fin
Name	varchar	NO	Nombre

<b>TABLA</b>	<b>PROJECT_TASK</b>		
<b>Nombre_Campo</b>	<b>Tipo</b>	<b>Requerido</b>	<b>Descripcion</b>
Active	boolean	SI	Activo
Buffer	boolean	SI	Buffer
buffer_time_taken	numeric	SI	Buffer Time Taken
buffer_type	varchar	SI	Buffer Type
Color	integer	SI	Color Index
company_id	integer	SI	Compañía
critical_chain	boolean	SI	Cadena Crítica
critical_path	boolean	SI	Critical Path
critical_resource	boolean	SI	Critical Resource
date_deadline	date	SI	Deadline
date_end	timestamp without time zone	SI	Ending Date
date_last_stage_update	timestamp without time zone	SI	Last Stage Update
date_start	timestamp without time zone	SI	Starting Date
delay_hours	numeric	SI	Delay Hours
description	text	SI	Description
effective_hours	numeric	SI	Hours Spent
end_date_suggested	timestamp without time zone	SI	Start Date Suggested
feeder	boolean	SI	Feeder
group	boolean	SI	Group
identifier	varchar	SI	Identifier
initial_plan	boolean	SI	Initial Planned
kanban_state	varchar	SI	Kanban State
message_last_post	timestamp without time zone	SI	Last Message Date
milestone	boolean	SI	Milestone

name	varchar	NO	Task Summary
notes	text	SI	Notes
partner_id	integer	SI	Customer
planned_time	numeric	SI	Planned Time
priority	varchar	SI	Priority
progress	numeric	SI	Working Time Progress (%)
project_id	integer	SI	Project
project_state	varchar	SI	Project State
remaining_hours	numeric	SI	Remaining Hours
reviewer_id	integer	SI	Reviewer
sequence	integer	SI	Sequence
stage_id	integer	SI	Stage
start_date_suggested	timestamp without time zone	SI	Start Date Suggested
time_uom_id	integer	SI	Time UoM
total_hours	numeric	SI	Total
total_time	numeric	SI	Total Time
user_id	integer	SI	Assigned to

<b>TABLA</b>	<b>PROJECT_TASK_DELEGATE</b>		
<b>Nombre_Campo</b>	<b>Tipo</b>	<b>Requerido</b>	<b>Descripcion</b>
name	varchar	NO	Delegated Title
new_task_description	text	SI	New Task Description
planned_hours	double precision	SI	Planned Hours
planned_hours_me	double precision	SI	Hours to Validate
prefix	varchar	SI	Your Task Title
project_id	integer	SI	Project
state	varchar	SI	Validation State
user_id	integer	NO	Assign To

<b>TABLA</b>	<b>PROJECT_TASK_HISTORY</b>		
<b>Nombre_Campo</b>	<b>Tipo</b>	<b>Requerido</b>	<b>Descripcion</b>
date	date	SI	Date
end_date	date	SI	End Date
kanban_state	varchar	SI	Kanban State
planned_hours	numeric	SI	Planned Time
remaining_hours	numeric	SI	Remaining Time
task_id	integer	NO	Task
type_id	integer	SI	Stage
user_id	integer	SI	Responsible

<b>TABLA</b>	<b>PROJECT_TASK_TYPE</b>		
<b>Nombre_Campo</b>	<b>Tipo</b>	<b>Requerido</b>	<b>Descripcion</b>
case_default	boolean	SI	Default for New Projects
description	text	SI	Description
fold	boolean	SI	Folded in Kanban View
name	varchar	NO	Stage Name
sequence	integer	SI	Sequence

<b>TABLA</b>	<b>PROJECT_TASK_WORK</b>		
<b>Nombre_Campo</b>	<b>Tipo</b>	<b>Requerido</b>	<b>Descripcion</b>
company_id	Integer	SI	Company
date	timestamp without time zone	SI	Date
hours	double precisión	SI	Time Spent
hr_analytic_timesheet_id	Integer	SI	Related Timeline Id
name	varchar	SI	Work summary
task_id	integer	NO	Task
time_executed	numeric	SI	Time Excecuted
time_uom_id	Integer	SI	Time UoM
user_id	Integer	NO	Done by

<b>TABLA</b>	<b>PROJECT_TASK_WORK</b>		
<b>Nombre_Campo</b>	<b>Tipo</b>	<b>Requerido</b>	<b>Descripción</b>
account_no	varchar	SI	Account No.
company_registry	varchar	SI	Company Registry
custom_footer	boolean	SI	Custom Footer
day_uom_id	integer	SI	UoM equal at Day
default_calendar_id	integer	SI	Default Calendar
Email	varchar	SI	Email
expects_chart_of_accounts	boolean	SI	Expects a Chart of Accounts
expense_currency_exchange_account_id	integer	SI	Loss Exchange Rate Account
font	integer	SI	Font
hour_uom_id	integer	SI	UoM equal at Hour
income_currency_exchange_account_id	integer	SI	Gain Exchange Rate Account
logo_web	bytea	SI	Logo Web
minute_uom_id	Integer	SI	UoM equal at minute
overdue_msg	Text	SI	Overdue Payments Message
paperformat_id	Integer	SI	Paper format
parent_id	Integer	SI	Parent Company

paypal_account	varchar	SI	Paypal Account
phone	varchar	SI	Phone
project_time_mode_id	integer	SI	Project Time Unit
rml_footer	text	SI	Report Footer
rml_header	text	NO	RML Header
rml_header1	varchar	SI	Company Tagline
rml_header2	text	NO	RML Internal Header
rml_header3	text	NO	RML Internal Header for Landscape Reports
rml_paper_format	varchar	NO	Paper Format
sale_note	text	SI	Default Terms and Conditions
tax_calculation_rounding_method	varchar	SI	Tax Calculation Rounding Method
timesheet_max_difference	double precision	SI	Timesheet allowed difference(Hours)
timesheet_range	varchar	SI	Timesheet range
time_uom_categ_id	integer	SI	Default Time UoM Category
time_uom_id	integer	SI	Default Time UoM

<b>TABLA</b>	<b>RESOURCE_CALENDER</b>		
<b>Nombre_Campo</b>	<b>Tipo</b>	<b>Requerido</b>	<b>Descripción</b>
code	varchar	SI	Code
company_id	integer	SI	Company
manager	integer	SI	Workgroup Manager
name	varchar	NO	Name
time_zone	varchar	SI	Time Zone

<b>TABLA</b>	<b>RES_USERS</b>		
<b>Nombre_Campo</b>	<b>Tipo</b>	<b>Requerido</b>	<b>Descripción</b>
action_id	Integer	SI	Home Action
alias_id	Integer	NO	Alias
default_section_id	Integer	SI	Default Sales Team
display_employees_suggestions	Boolean	SI	Display Employees Suggestions
display_groups_suggestions	Boolean	SI	Display Groups Suggestions
login_date	Date	SI	Latest connection
password_crypt	varchar	SI	Encrypted Password
share	boolean	SI	Share User
signature	Text	SI	Signature

<b>TABLA</b>	<b>WIZARD_CALC_BUFFERS</b>
--------------	----------------------------

Nombre_Campo	Tipo	Requerido	Descripción
project_buffer	numeric	SI	Buffer del proyecto
project_id	Integer	SI	Proyecto
time_uom_id	integer	SI	Tiempo de Unidad de medida

TABLA	WIZARD_CALC_BUFFERS_LINE		
Nombre_Campo	Tipo	Requerido	Descripción
buffer_planned_time	numeric	SI	Tiempo de buffer planificado
buffer_type	varchar	SI	Tipo de Buffer
current_planned_time	numeric	SI	Tiempo planificado
task_id	integer	SI	Tarea
wizard_id	integer	SI	Wizard

TABLA	WIZARD_IMPORT_PROJECT		
Nombre_Campo	Tipo	Requerido	Descripción
encoding_id	Integer	SI	Codificación
field_delimiter_id	Integer	SI	Delimitador de Campo
file_data	Bytea	NO	Archivo Binario
name	varchar	SI	Nombre del Proyecto
start_date	Date	NO	Fecha de Inicio del Proyecto
text_delimiter_id	Integer	SI	Delimitador de Texto
type	varchar	NO	Tipo de Archivo
user_id	integer	SI	Responsable del Proyecto

TABLA	WIZARD_IMPORT_PROJECT_LINE		
Nombre_Campo	Tipo	Requerido	Descripción
identifier	varchar	SI	ID
name	varchar	SI	Nombre Tarea
observation	varchar	SI	Observaciones
resource_names	varchar	SI	Recursos
uom_id	integer	SI	UdM Tiempo
user_id	integer	SI	Usuario
valid	boolean	SI	Válido
wizard_id	integer	SI	Asistente
worktime	numeric	SI	Tiempo de Trabajo

<b>TABLA</b>	<b>WIZARD_IMPORT_PROJECT_USER</b>		
<b>Nombre_Campo</b>	<b>Tipo</b>	<b>Requerido</b>	<b>Descripción</b>
name	varchar	SI	Nombre
resource_id	integer	SI	Recurso
user_id	integer	SI	Usuario

### ANEXO 3: FORMATO DE PRUEBAS UNITARIAS

<b>Caso de Uso:</b>		<b>N°</b>	
<b>Escenario:</b>			
<b>Responsable:</b>		<b>Fecha:</b>	
<b>Precondiciones:</b>			
<b>Datos de Entrada:</b>			
<b>Descripción de Pasos</b>			
<b>Resultado Esperado</b>		<b>Cumplimiento</b>	
<b>Resultado Obtenido</b>		<b>Fallas Provocadas:</b>	
<b>Recomendación y Observación</b>			

**ANEXO 4: FORMATO DE PRUEBAS DE INTEGRACIÓN**

<b>Escenario:</b>		<b>N°</b>	
<b>Lista de Módulos:</b>			
<b>Responsable:</b>			<b>Fecha:</b>
<b>Precondiciones:</b>			
<b>Datos de Entrada:</b>			
<b>Descripción de Pasos</b>			
<b>Resultado Esperado</b>		<b>Cumplimiento</b>	
<b>Resultado Obtenido</b>		<b>Fallas Provocadas:</b>	
<b>Recomendación y Observación</b>			