



**UNIVERSIDAD DE CHILE**  
**FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS**  
**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

**PLATAFORMA DE GESTIÓN DEL CONOCIMIENTO BASADA EN  
ARQUITECTURA COLABORATIVA E INTERFAZ DE RED SOCIAL**

**TESIS PARA OPTAR AL GRADO DE  
MAGÍSTER EN TECNOLOGÍAS DE LA INFORMACIÓN**

**DAVID NICOLÁS DAVID AGUILERA**

**PROFESOR GUÍA:  
NELSON ANTRANIG BALOIAN TATARYAN**

**MIEMBROS DE LA COMISIÓN:  
SERGIO OCHOA DELORENZI  
JOSÉ ALBERTO PINO URTUBIA  
HECTOR ANTILLANCA ESPINA**

**SANTIAGO DE CHILE  
ABRIL 2015**

# Resumen

Hoy en día, el conocimiento se considera como el activo más importante que pueden poseer las empresas y organizaciones, ya que es fundamental para el crecimiento y productividad de éstas. Aquellas organizaciones que tienden a favorecer el crecimiento del conocimiento, buscan generar ventajas competitivas, para poder sobresalir sobre aquellas que no lo hacen.

Para poder capturar, organizar y almacenar el conocimiento dentro de las organizaciones, se definieron, hace algún tiempo, conceptos como el de gestión del conocimiento o el de memoria organizacional, dentro de los cuales se idearon técnicas, como la de uso de sistemas de gestión del conocimiento o KMS. Estos sistemas tienen como objetivo básicamente traspasar el conocimiento tácito a conocimiento explícito proveyendo principalmente herramientas como repositorios de documentos, bases de datos de expertos o listas de discusiones.

Lo que este trabajo de tesis propone, es el diseño y construcción de una plataforma de gestión del conocimiento con una arquitectura colaborativa y con características de redes sociales, con el fin de incentivar el uso de estas herramientas y poder contar con un sistema mucho más atractivo para las personas, facilitando la transferencia de conocimiento entre los trabajadores, cumpliendo con los objetivos requeridos de una gestión del conocimiento corporativa y facilitando la implantación de un KMS.

El caso es aplicado al holding de empresas de Komatsu en Chile, el cual fue estudiado en detalle y desarrollado en base a los requerimientos y problemas comunes de este cliente, el que tuvo una aceptación positiva de parte de los colaboradores durante la marcha blanca y en donde se pudo concluir que esta implementación permitiría mantener e incrementar el conocimiento de la compañía de manera fácil, mejorar la comunicación entre los colaboradores e incentivar la innovación a través de un sistema corporativo y con imagen propia.

# Tabla de Contenido

|   |           |
|---|-----------|
| <b>Introducción</b> .....                                       | <b>1</b>  |
| 1.1    Conceptos básicos .....                                  | 1         |
| 1.1.1    Contexto de la situación actual .....                  | 1         |
| 1.1.2    Contexto tecnológico .....                             | 2         |
| 1.2    Problemática a resolver .....                            | 3         |
| 1.3    Justificación de la propuesta.....                       | 5         |
| 1.3.1    Solución tecnológica .....                             | 5         |
| 1.3.2    Mejoras en el desempeño de la operación.....           | 5         |
| 1.3.3    Beneficios para el usuario final .....                 | 6         |
| 1.4    Objetivos de la propuesta .....                          | 6         |
| 1.4.1    General.....   | 6         |
| 1.4.2    Específicos.....                                       | 6         |
| 1.5    Metodología .....  | 7         |
| 1.6    Plan de trabajo .....                                    | 9         |
| 1.7    Estructura de la tesis.....                              | 10        |
| <b>Estado del Arte</b> .....                                    | <b>12</b> |
| 2.1    Introducción.....  | 12        |
| 2.2    Memoria organizacional y gestión del conocimiento .....  | 12        |
| 2.3    Aprendizaje Colaborativo.....                            | 14        |
| 2.4    Redes Sociales.....                                      | 14        |
| 2.5    Relación entre los conceptos.....                        | 15        |
| 2.5.1    Memoria organizacional y aprendizaje colaborativo..... | 16        |
| 2.5.2    Aprendizaje colaborativo y redes sociales.....         | 18        |
| 2.5.3    Memoria organizacional y redes sociales .....          | 20        |
| 2.6    Contribución del capítulo .....                          | 22        |
| <b>Análisis de Requisitos</b> .....                             | <b>23</b> |

|                                     |  |           |
|-------------------------------------|--|-----------|
| 3.1                                 | Visión general del proyecto .....  | 23        |
| 3.2                                 | Analogía entre redes sociales y sistemas de gestión del conocimiento ..... | 23        |
| 3.3                                 | Aplicación de metodología .....  | 27        |
| 3.3.1                               | Plantilla OM-1.....  | 27        |
| 3.3.2                               | Plantilla OM-2.....  | 30        |
| 3.3.3                               | Plantilla OM-3.....  | 32        |
| 3.3.4                               | Plantilla OM-4.....  | 34        |
| 3.3.5                               | Plantilla OM-5.....  | 36        |
| 3.4                                 | Requisitos funcionales.....  | 38        |
| 3.5                                 | Requisitos no funcionales .....  | 39        |
| 3.6                                 | Alcances y restricciones del proyecto.....                                 | 40        |
| 3.7                                 | Contribución del capítulo .....  | 42        |
| <b>Diseño de la Plataforma.....</b> |  | <b>43</b> |
| 4.1                                 | Características de la arquitectura del sistema.....                        | 43        |
| 4.2                                 | Arquitectura de la plataforma.....   | 43        |
| 4.2.1                               | Arquitectura física.....   | 43        |
| 4.2.2                               | Servidor de aplicaciones.....  | 44        |
| 4.2.3                               | Servidor de bases de datos.....  | 44        |
| 4.2.4                               | Servidor de repositorios .....   | 45        |
| 4.2.5                               | Conexión con la nube .....   | 45        |
| 4.3                                 | Componentes de la plataforma .....   | 46        |
| 4.3.1                               | Núcleo de la plataforma.....   | 46        |
| 4.3.2                               | Componente de identidad .....  | 47        |
| 4.3.3                               | Componente de sensibilidad .....   | 48        |
| 4.3.4                               | Componente de etiquetado.....  | 48        |
| 4.3.5                               | Buscador semántico .....   | 49        |
| 4.3.6                               | Procesos de back-end.....  | 49        |
| 4.3.7                               | Extensiones y aplicaciones.....  | 50        |
| 4.4                                 | Gestor de colas de mensajes .....  | 50        |

---

|       |   |           |
|-------|---|-----------|
| 4.5   | Componentes del front-end de la plataforma..... | 51        |
| 4.5.1 | Buscador semántico .....                        | 52        |
| 4.5.2 | Identidad .....                                 | 53        |
| 4.5.3 | Sensibilidad.....                               | 54        |
| 4.5.4 | Grupos .....                                    | 55        |
| 4.5.5 | Preguntas y respuestas .....                    | 56        |
| 4.5.6 | Gestión documental .....                        | 57        |
| 4.5.7 | Wiki.....                                       | 57        |
| 4.5.8 | Otros componentes .....                         | 58        |
| 4.6   | Procesos del back-end de la plataforma.....     | 60        |
| 4.6.1 | Motor de indexación de etiquetas.....           | 60        |
| 4.6.2 | Proceso de sensibilización.....                 | 61        |
| 4.6.3 | Proceso de actualización de wikis .....         | 62        |
| 4.6.4 | Proceso de recomendaciones .....                | 63        |
| 4.6.5 | Proceso de actividades .....                    | 64        |
| 4.6.6 | Proceso de estadísticas .....                   | 65        |
| 4.7   | Diseño gráfico.....                             | 66        |
| 4.7.1 | Logo y colores corporativos.....                | 66        |
| 4.7.2 | Estructura encabezado .....                     | 68        |
| 4.7.3 | Estructura del sitio web.....                   | 70        |
| 4.8   | Contribución del capítulo .....                 | 71        |
|       | <b>Implementación de la Plataforma.....</b>     | <b>72</b> |
| 5.1   | Plan de proyecto.....                           | 72        |
| 5.2   | Metodología de desarrollo de software .....     | 73        |
| 5.3   | Revisión de procesos de negocio .....           | 74        |
| 5.4   | Revisión de alternativas tecnológicas.....      | 74        |
| 5.5   | Especificación de requisitos.....               | 74        |
| 5.6   | Diseño de la plataforma .....                   | 75        |
| 5.6.1 | Diseño arquitectónico de los servidores .....   | 75        |

---

|  |  |            |
|--|--|------------|
| 5.6.2  | Diseño base de datos .....                               | 76         |
| 5.6.3  | Diseño MVC.....  | 77         |
| 5.6.4  | Diseño gestor de colas de mensaje.....                   | 83         |
| 5.7  | Implementación de la plataforma.....                     | 83         |
| 5.7.1  | Utilización del Framework .....                          | 83         |
| 5.7.2  | Codificación de la plataforma .....                      | 85         |
| 5.7.3  | Base de datos.....                                       | 88         |
| 5.7.4  | Implementación de la interfaz gráfica.....               | 89         |
| 5.8  | Despliegue de los servidores.....                        | 90         |
| 5.8.1  | Obtención de dominio internet .....                      | 90         |
| 5.8.2  | Instalación de servidor .....                            | 91         |
| 5.8.3  | Instalación y configuración de paquetes en servidor..... | 91         |
| 5.9  | Despliegue de la plataforma.....                         | 94         |
| 5.9.1  | Configuración de la plataforma .....                     | 94         |
| 5.10   | Pruebas de la plataforma .....                           | 94         |
| 5.10.1                                       | Pruebas de aseguramiento de calidad.....                 | 94         |
| 5.10.2                                       | Pruebas funcionales y de sistema .....                   | 95         |
| 5.10.3                                       | Pruebas de aceptación final .....                        | 96         |
| 5.11   | Puesta en producción del sistema.....                    | 97         |
| 5.11.1                                       | Marcha blanca.....                                       | 97         |
| 5.11.2                                       | Encuesta experiencia de usuario.....                     | 99         |
| 5.11.3                                       | Estadísticas de uso en marcha blanca .....               | 100        |
| 5.12   | Contribución del capítulo .....                          | 101        |
| <b>Conclusiones y trabajo a futuro .....</b> |  | <b>102</b> |
| 6.1  | Análisis por capítulo.....                               | 102        |
| 6.2  | Verificación del cumplimiento de objetivos.....          | 105        |
| 6.3  | Contribución de la tesis.....                            | 106        |
| 6.4  | Limitaciones.....  | 106        |
| 6.5  | Trabajo a futuro .....                                   | 107        |

|                          |            |
|--------------------------|------------|
| <b>Bibliografía.....</b> | <b>109</b> |
| <b>Acrónimos.....</b>    | <b>114</b> |
| <b>Anexos.....</b>       | <b>116</b> |
| Anexo A.....             | 116        |
| Anexo B.....             | 116        |
| Anexo C.....             | 118        |
| Anexo D.....             | 120        |
| Anexo E.....             | 120        |
| Anexo F.....             | 121        |
| Anexo G.....             | 121        |
| Anexo H.....             | 124        |
| Anexo I.....             | 125        |
| Anexo J.....             | 126        |
| Anexo K.....             | 126        |
| Anexo L.....             | 128        |

# Índice de Figuras

|  |    |
|--|----|
| Figura 1.1: Modelo de CommonKADS .....   | 8  |
| Figura 1.2: Mapa de trabajo para el nivel de contexto .....                    | 8  |
| Figura 1.3: Carta Gantt del desarrollo del proyecto de Tesis.....              | 10 |
| Figura 2.1: Relación entre los principales conceptos del estado del arte ..... | 16 |
| Figura 3.1: Estructura organizacional del Holding .....                        | 30 |
| Figura 4.1: Arquitectura física de la plataforma .....                         | 43 |
| Figura 4.2: Componentes del servidor de aplicaciones .....                     | 44 |
| Figura 4.3: Servidor de bases de datos.....                                    | 45 |
| Figura 4.4: Servidor de repositorios .....                                     | 45 |
| Figura 4.5: Componentes de la plataforma .....                                 | 47 |
| Figura 4.6: Componentes del núcleo de la plataforma .....                      | 47 |
| Figura 4.7: Componentes de identidad.....                                      | 47 |
| Figura 4.8: Componentes de sensibilidad.....                                   | 48 |
| Figura 4.9: Componentes de etiquetado .....                                    | 49 |
| Figura 4.10: Procesos de back-end.....   | 49 |
| Figura 4.11: Extensiones y aplicaciones .....                                  | 50 |
| Figura 4.12: Gestor de colas de mensaje .....                                  | 51 |
| Figura 4.13: Caso de Uso para Buscador semántico.....                          | 52 |
| Figura 4.14: Wireframe del buscador semántico .....                            | 53 |
| Figura 4.15: Caso de Uso para la Identidad .....                               | 53 |
| Figura 4.16: Wireframe del perfil.....   | 54 |
| Figura 4.17: Caso de Uso para la Sensibilidad .....                            | 55 |
| Figura 4.18: Caso de Uso para Grupos.....                                      | 56 |
| Figura 4.19: Caso de Uso para Preguntas y Respuestas .....                     | 56 |
| Figura 4.20: Caso de Uso para Gestión Documental.....                          | 57 |
| Figura 4.21: Caso de Uso para Wiki .....                                       | 58 |
| Figura 4.22: Wireframe de página principal.....                                | 59 |
| Figura 4.23: Wireframe de sección compartir .....                              | 59 |
| Figura 4.24: Diagrama de proceso de actualización.....                         | 61 |
| Figura 4.25: Diagrama de proceso de sensibilización .....                      | 62 |
| Figura 4.26: Diagrama de proceso de actualización de wiki.....                 | 63 |
| Figura 4.27: Diagrama de proceso de recomendación.....                         | 64 |
| Figura 4.28: Diagrama de proceso de actividad.....                             | 65 |
| Figura 4.29: Diagrama de proceso de estadísticas.....                          | 66 |



|  |    |
|--|----|
| Figura 4.30: Logo corporativo .....                                  | 67 |
| Figura 4.31: Logo propuesto para plataforma .....                    | 68 |
| Figura 4.32: Wireframe del encabezado.....                           | 69 |
| Figura 4.33: Diseño de encabezado propuesto para la plataforma ..... | 69 |
| Figura 4.34: Explicación de diseño de encabezado propuesto .....     | 70 |
| Figura 4.35: Diseño final propuesto para la plataforma .....         | 71 |
| Figura 5.1: Metodología de desarrollo de software incremental.....   | 73 |
| Figura 5.2: Patrón de diseño modelo-vista-controlador .....          | 78 |
| Figura 5.3: Utilización del framework .....                          | 84 |
| Figura 5.4: Estructura de carpetas del proyecto .....                | 84 |
| Figura 5.5: Mapa de navegación de la plataforma .....                | 88 |
| Figura 5.6: Acceso vía SSH a la consola del servidor.....            | 91 |
| Figura 5.7: Método de la caja blanca para pruebas unitarias.....     | 95 |
| Figura 5.8: Actividad reflejada durante la marcha blanca .....       | 97 |
| Figura 5.9: Términos definidos durante la marcha blanca.....         | 98 |
| Figura 5.10: Ejemplo de perfil creado durante la marcha blanca ..... | 98 |

# Índice de Tablas

|   |     |
|---|-----|
| Tabla 3.1: Tabla de analogía entre redes sociales y SGC .....                         | 24  |
| Tabla 3.2: Tabla de Plantilla OM-1 de CommonKADS.....                                 | 27  |
| Tabla 3.3: Tabla de Plantilla OM-2 de CommonKADS.....                                 | 30  |
| Tabla 3.4: Tabla de Plantilla OM-3 de CommonKADS.....                                 | 32  |
| Tabla 3.5: Tabla de Plantilla OM-4 de CommonKADS.....                                 | 34  |
| Tabla 3.6: Tabla de Plantilla OM-5 de CommonKADS.....                                 | 36  |
| Tabla 4.1: Tabla de colores utilizados en la plataforma.....                          | 67  |
| Tabla 4.2: Tabla de encabezados de redes más populares .....                          | 68  |
| Tabla 5.1: Controladores de la plataforma .....                                       | 85  |
| Tabla 5.2: Tabla con el detalle de las vistas de la plataforma.....                   | 86  |
| Tabla 5.3: Tabla con el script de creación de colecciones .....                       | 88  |
| Tabla 5.4: Tabla con el detalle de librerías y hojas de estilo.....                   | 89  |
| Tabla 5.5: Descripción de paquetes instalados en servidor.....                        | 91  |
| Tabla 5.6: Detalle de la configuración del servidor FTP .....                         | 93  |
| Tabla 5.7: Detalle de la configuración del servidor MongoDB.....                      | 93  |
| Tabla 5.8: Pruebas funcionales .....  | 95  |
| Tabla 5.9: Pruebas de aceptación final .....  | 96  |
| Tabla 5.10: Encuesta de experiencia de usuario.....                                   | 99  |
| Tabla 5.11: Estadísticas de uso en marcha blanca.....                                 | 100 |
| Tabla 6.1: Promedios por objetivos específicos.....                                   | 105 |
| Tabla 7.1: Tabla con la especificación de la estructura de carpetas del proyecto..... | 116 |
| Tabla 7.2: Contenido y enrutamiento del archivo index.php .....                       | 116 |
| Tabla 7.3: Contenido del controlador KM.php .....                                     | 118 |
| Tabla 7.4: Contenido del controlador Chat.php .....                                   | 120 |
| Tabla 7.5: Contenido del controlador Documents.php.....                               | 120 |
| Tabla 7.6: Contenido del controlador Images.php .....                                 | 121 |
| Tabla 7.7: Contenido del modelo Activities.php .....                                  | 121 |
| Tabla 7.8: Contenido del modelo Entries.php.....                                      | 122 |
| Tabla 7.9: Contenido del modelo Galleries.php .....                                   | 122 |
| Tabla 7.10: Contenido del modelo Groups.php.....                                      | 123 |
| Tabla 7.11: Contenido del modelo Logs.php.....  | 123 |
| Tabla 7.12: Contenido del modelo Messages.php .....                                   | 123 |
| Tabla 7.13: Contenido del modelo Sensibilities.php .....                              | 123 |
| Tabla 7.14: Contenido del modelo Users.php .....                                      | 124 |

|  |     |
|--|-----|
| Tabla 7.15: Detalle de la configuración del servidor DNS.....          | 124 |
| Tabla 7.16: Detalle de la configuración del sitio web.....             | 125 |
| Tabla 7.17: Detalle de la configuración del servidor MySQL .....       | 126 |
| Tabla 7.18: Contenido de las carpetas subidas por FTP .....            | 126 |
| Tabla 7.19: Detalle del archivo de configuración de la plataforma..... | 128 |
| Tabla 7.20: Contenido del archivo .htaccess.....                       | 128 |

# Introducción

## 1.1 Conceptos básicos

### 1.1.1 Contexto de la situación actual

El conocimiento es un valor relevante dentro de una organización, dado que su adecuada administración, y el impacto que tiene en la competitividad, son aspectos fundamentales para lograr el éxito sostenible dentro de las empresas.

La memoria organizacional, consiste en el almacenamiento y preservación del conocimiento dentro de las organizaciones con la finalidad de reutilizarlo e ir mejorando los procesos que la experiencia va dando. Sin embargo, las empresas no siempre manejan de manera correcta la memoria organizacional, lo que puede provocar una fuga de conocimiento, que si no es controlada, traerá problemas complejos de eficacia y de ventajas competitivas.

Es importante mencionar que el conocimiento reside en las personas, por lo tanto depende de éstas y de su capacidad de liderazgo, debido a que es una actividad de gestión. Dicha actividad está íntimamente vinculada con la cultura organizacional, la cual resulta complicada, lenta y ardua al momento de atravesar por un proceso de mejoramiento. El conocimiento para que sea útil debe estar instaurado en cada una de las actividades cotidianas de la organización, tanto en las personas como en la interacción entre éstas, lo que implica la modificación en las rutinas organizativas, transformando el trabajo individual a un trabajo en equipo, enfocándose en producir resultados.

La gestión del conocimiento implica la evolución de la cultura organizacional, significa un cambio de estructuras organizativas y transformación de procesos. Lamentablemente, a pesar que en muchas organizaciones existe una carencia en saber cómo implementar un sistema de gestión del conocimiento, muchas de éstas se han quedado sólo las ganas de realizar un método para gestionar el conocimiento, y de las

que lo han logrado, algunas han fracasado en el proceso de implementación o no han utilizado herramientas realmente efectivas.

Las organizaciones no siempre relacionan este proceso con una gestión de cambio, lo cual implica que no se produce la transformación deseada respecto a la gestión del conocimiento, así como tampoco dan énfasis al proceso de aprender a aprender, es decir, ampliar la capacidad de aprendizaje de las personas.

### 1.1.2 Contexto tecnológico

Dentro de las organizaciones existen sistemas muy estructurados, muchos de ellos con interfaces obsoletas, lo que da como resultado que las personas se sientan poco atraídas para interactuar con ellas. Generalmente, las implementaciones de sistemas de gestión del conocimiento que se han realizado, desarrollan sistemas con una visión muy poco orientada al usuario, provocando poca participación de los mismos y un escaso aprendizaje. Es muy difícil encontrar sistemas que estén enfocados realmente en las personas y sean atractivos. Por lo general, éstos sólo se encuentran en plataformas de ocio, muy lejanos de la formalidad que requiere el rubro empresarial.

La revolución de la Web 2.0 o web social se produjo con la llegada de una nueva generación de tecnologías y aplicaciones interactivas que permiten una fácil publicación, edición y difusión de contenidos, así como la creación de redes personales y comunidades en línea. A su vez, la Web 3.0 o web semántica, es un salto con respecto a la 2.0, creando una base semántica y cualitativa del conocimiento, lo que da como resultado una herramienta esencial para la fidelización del usuario. Esta generación, a diferencia de la anterior, le da protagonismo a motores y procesadores de información, para conseguir una personalización de datos más eficiente.

Adicionalmente, existe una nueva tendencia de incluir el uso de redes sociales dentro de las empresas, sin embargo, pocas de ellas se han dado cuenta del potencial que genera el *social media* y el *social business*, los cuales añaden valor a la empresa y mejoran la productividad, la innovación, la colaboración, la reputación y sobre todo, el involucramiento de los empleados en ellas. Además, la mayoría de las organizaciones no cuentan con políticas establecidas sobre el uso de redes sociales en el trabajo, lo que sugiere que es necesario desarrollar también políticas de regulación en este sentido. El punto importante sobre la implementación de estos sistemas, es que es posible enlazar las redes sociales con la gestión del conocimiento, creando comunidades interactivas, de colaboración y conectadas a través de intereses comunes.

Otro de los retos en las organizaciones es pensar en la estrategia de aprendizaje dentro de ellas, el cual debe estar enfocado en la colaboración e informalidad, elementos que ahorran dinero e incrementan la cobertura general del aprendizaje organizacional. Es aquí donde el uso de la tecnología regala sus ventajas, una de ellas es que permite interactuar con una amplia diversidad de personas, incluso aquellas que se encuentren geográficamente distantes (distribuidas) y/o a distintas horas (asíncronamente). Otra es el intercambio de la información, el que facilita el desarrollo de una cultura de aprendizaje colaborativo. Este aprendizaje colaborativo es un proceso social que trae como resultados la generación de conocimiento compartido, posibilita la solución de problemas en conjunto, representa el entendimiento común de un grupo con respecto al contenido de un dominio específico y permite la distribución del conocimiento, además de que permite compartir experiencias.

## 1.2 Problemática a resolver

Lo que este trabajo propone, es plantear una mejora en los sistemas de gestión del conocimiento que no han sido lo suficientemente eficaces en las organizaciones debido a sus debilidades en la implementación, rescatando las ventajas y beneficios que ofrecen las redes sociales y el aprendizaje colaborativo, proponiendo un diseño de solución que sea adaptable, efectivo y aplicable a la mayoría de las organizaciones.

Este proyecto, fue aplicado al caso específico del holding de empresas de Komatsu en Chile, el que es estudiado en detalle y el trabajo es desarrollado en base a sus principales problemáticas y requerimientos específicos, como se muestran a continuación.

Komatsu es líder mundial en la fabricación de equipos para minería, construcción y forestal. Se encuentra presente en Chile por más de 40 años a través de distribuidoras, y en 1999 se establece como un holding, incorporando experiencia, tecnología de vanguardia y volumen organizacional.

En un escenario donde la actividad minera se constituye como una de las principales actividades económicas del país, Komatsu se ha consolidado como uno de los principales socios estratégicos en proveer de equipos, repuestos y servicios a la pequeña, mediana y gran minería chilena, transformándose además en la compañía con mayor experiencia en contratos de servicios de mantención a flotas de equipos mineros.

Debido al crecimiento en la industria minera, Komatsu ha debido aumentar su número de trabajadores, desde 4000 hasta 5000 personas en el último año (2014) y sigue aumentando (20% anual promedio). Su estructura organizativa, se encuentra repartida en la mayoría de las regiones del país, debido a que más de la mitad de sus trabajadores se encuentran en faenas mineras, distribuyéndose en múltiples cargos. Adicionalmente, como la industria minera ha sido tan competitiva, existe una rotación de personal importante, por lo que la experiencia y el conocimiento se escapan muy fácilmente. Es por esto que se visualiza la necesidad de contar con un sistema de gestión del conocimiento corporativo, que independiente del lugar de trabajo y la rotación del personal, se pueda mantener y mejorar una memoria organizacional que no desaparezca con el tiempo. Cabe mencionar, que los activos intangibles constituyen uno de los principales factores del éxito presente y futuro de la compañía, incrementándose cada vez más las inversiones en éstos.

Una vez dado el contexto anterior, en el análisis de detección de problemas y necesidades del cliente se concluyó lo siguiente:

- Existen 6 empresas dentro del holding que funcionan independiente una de otra, por lo que la comunicación y el manejo de conocimiento no es transversal.
- Los trabajadores se encuentran distribuidos en distintas faenas, oficinas o se encuentran en traslados frecuentes, por lo que la relación entre ellos es poco estable.
- Existe una pérdida de conocimiento importante debido a la alta rotación del personal capacitado generado por la competencia en el mercado minero.
- La compañía necesita generar una base de conocimiento común para todas las empresas del holding con el fin de transferir de forma rápida y efectiva el conocimiento adquirido.
- Se requiere que se tenga mayor control y visibilidad sobre las tareas que competen a las distintas empresas.
- La compañía requiere que las personas tengan mayor interacción para hablar de temas en específico.
- Existe la necesidad de innovación permanente, en un mercado altamente competitivo.
- Existe la necesidad de identificar problemas, estrategias y soluciones.
- Existe gran cantidad de información estructurada y no estructurada que poseen los trabajadores, la que no se encuentra ordenada.
- Los sistemas que existen actualmente son manejados por pequeños grupos de usuarios y son poco incentivados.

## 1.3 Justificación de la propuesta

Dados los problemas anteriormente señalados, se justifica el desarrollo de este proyecto para que proporcione mejoras y beneficios en distintos aspectos:

### 1.3.1 Solución tecnológica

**La solución tecnológica consiste en la implementación de una plataforma de gestión del conocimiento que genere las siguientes mejoras:**

- 1) **Fomentar la colaboración:** con el fin de mejorar el aprendizaje, el sistema estará basado en un modelo de aprendizaje colaborativo para poder transmitir el conocimiento disponible de una manera más fluida.
- 2) **Incentivo de uso:** para lograr una mejora en la utilización de la plataforma, se considerarán características típicas de las redes sociales.
- 3) **Usabilidad y alta disponibilidad:** la solución deberá cumplir con todos los estándares de usabilidad con el fin de mejorar la experiencia de usuario final y deberá mejorar la disponibilidad actual de los sistemas.
- 4) **Integración con otros sistemas:** con el fin de tener acceso a más sistemas, se dispondrá de un mecanismo que posibilite la integración con otros sistemas o aplicaciones disponibles.
- 5) **Fomento al desarrollo de nuevas ideas:** la solución debe ser un camino para que otras áreas se involucren y puedan generar nuevas ideas a través de esta misma plataforma o propuesta de nuevas aplicaciones de la plataforma.

### 1.3.2 Mejoras en el desempeño de la operación

Se espera que la solución provea distintas mejoras en el desempeño de la operación. Para evaluar esto, se realizará una encuesta a los usuarios, con el fin de medir las siguientes mejoras:

- Permitir transformar el conocimiento de las personas en competencias corporativas y ventajas competitivas.
- Permitir centralizar la información técnica esencial que se encuentra disponible en distintas aplicaciones.
- Permitir capturar, organizar y almacenar el conocimiento de los trabajadores.
- Acelerar el flujo de la información que tiene valor.
- Crear valor para los clientes.
- Promover actividades de mejora continua.



### 1.3.3 Beneficios para el usuario final

Debido a que esta plataforma está orientada a las personas, se espera que la solución propuesta provea a los usuarios diversos beneficios que se detallan a continuación:

- Facilitar la extracción de información para la construcción de informes.
- Guías de consulta rápida construidas por expertos.
- Disponer de foros internos para discusiones de temas en específicos.
- Facilitación del intercambio y el flujo de ideas, dudas, experiencias e inquietudes.
- Facilitar la notificación de información relevante.
- Asignación de tareas específicas a las personas.
- Aprender más rápidamente y sacar provecho de la experiencia de otros.

## 1.4 Objetivos de la propuesta

### 1.4.1 General

Diseñar y crear una plataforma que permita la gestión del conocimiento basada en una arquitectura colaborativa y con características de red social.

### 1.4.2 Específicos

Los objetivos específicos definidos para el alcanzar el objetivo general son los siguientes:

- 1) Gestionar el conocimiento de manera que sea posible crear una memoria organizacional.
- 2) Demostrar que las personas puedan participar de manera colaborativa dentro de la plataforma.
- 3) Determinar los módulos de la plataforma que permitan la creación de conocimiento y sean relevantes para los usuarios.
- 4) Considerar una implementación de la plataforma que mejore la comunicación en el trabajo.
- 5) Mejorar la usabilidad, navegación, experiencia de usuario y estabilidad en relación a otros sistemas informáticos de la compañía.

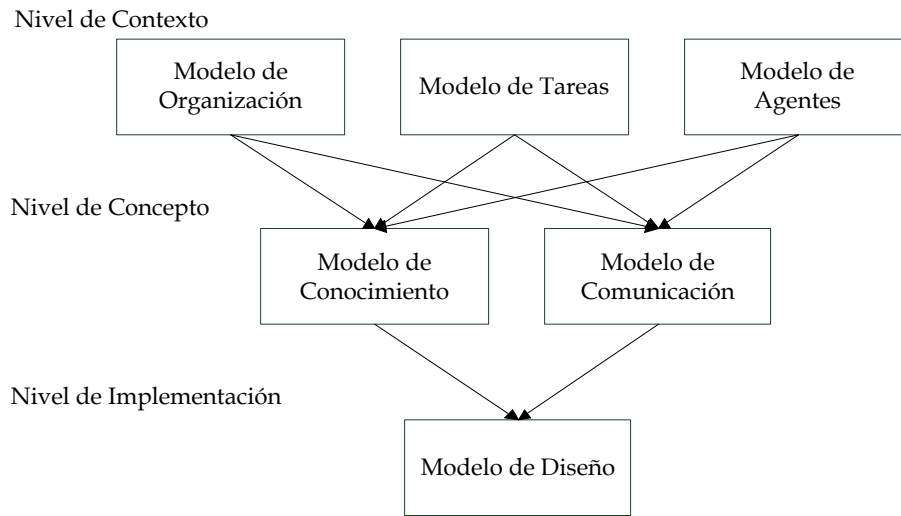
## 1.5 Metodología

Para este trabajo se usará la metodología CommonKADS. La metodología CommonKADS [44] se utiliza como un estándar por los responsables de la gestión del conocimiento e ingenieros del conocimiento para el análisis y construcción de sistemas en este sentido.

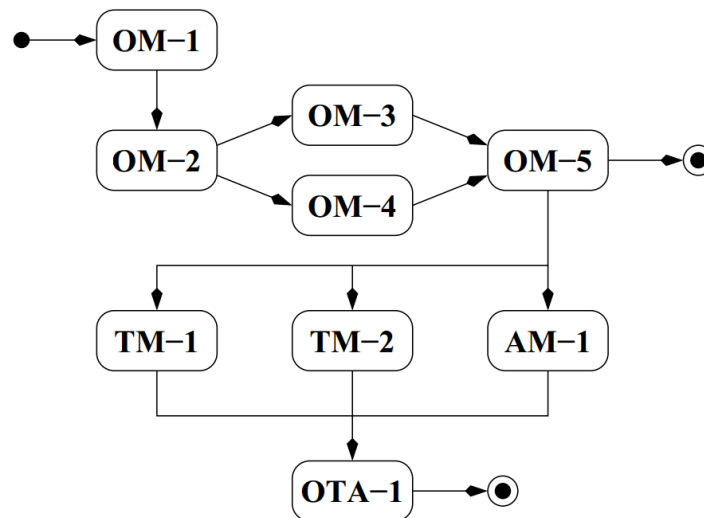
Su fortaleza radica en dos conceptos: primero, muestra lo importante que es el análisis de la organización para las estrategias de introducción y desarrollo de sistemas de tecnologías de la información, punto que se aprovechará en el desarrollo de este proyecto. En segundo lugar, CommonKADS se muestra cómo una metodología estándar del conocimiento, que soporta las aplicaciones de ingeniería y de gestión del conocimiento a la hora de construir los sistemas.

El principio que se utiliza para aplicar esta metodología se basa en el principio de nivel de conocimiento: que para modelar el conocimiento, primero hay que concentrarse en la estructura del conocimiento conceptual, y dejar los detalles de la programación para después.

El modelo de CommonKADS, se compone en tres submodelos: modelo de contexto, modelo conceptual y modelo de diseño para su construcción, como se muestra en la figura 1.1. Para este proyecto se aplicará el modelo de contexto, que responde a las preguntas del porqué del sistema a diseñar. Justifica la razón del sistema de gestión del conocimiento que se pretende construir. Determina los problemas y oportunidades a resolver, considerando el impacto que se produce en la organización. En este modelo, lo más importante es entender el contexto y el entorno de trabajo. Para ello, se ayuda del modelo de la organización, el modelo de las tareas y el modelo de los agentes.



**Figura 1.1: Modelo de CommonKADS**



**Figura 1.2: Mapa de trabajo para el nivel de contexto**

En la figura 1.2, se detalla la primera parte del modelo de la organización, que soporta el análisis de la organización para descubrir los problemas y oportunidades en los sistemas de conocimiento, establecer su factibilidad y valorar los impactos que tendrán las acciones de conocimiento sobre la organización. Utiliza las plantillas OM-1, donde se detectan los problemas y oportunidades de la organización, que pretenden ser resueltos; la plantilla OM-2, proporciona una visión de la estructura de la organización;

en la plantilla OM-3, se descompone el proceso en sus elementos principales; la plantilla OM-4, muestra los componentes de conocimiento; y por último, la plantilla OM-5, donde se pone de manifiesto los criterios que justifican la viabilidad del sistema de conocimiento. Para este proyecto, se han desarrollado las plantillas OM-1, OM-2, OM-3, OM-4 y OM-5 del modelo de la organización, que se desarrollan en los distintos capítulos de esta tesis.

## 1.6 Plan de trabajo

Este proyecto se implementó en una sola fase de 7 meses que contempló la investigación detallada, diseño la solución e implementación de la plataforma. Las actividades realizadas para cumplir con los objetivos de la tesis se enumeran a continuación:

- a. Investigación inicial: Corresponde a la etapa de investigación previa que finaliza con la entrega de esta propuesta.
- b. Revisión de requerimientos de negocio: Definición de las restricciones y alcances del proyecto.
- c. Investigación detallada: Revisión en profundidad de los métodos y principios existentes en la literatura para poder diseñar una solución acorde al problema. Investigación de herramientas reutilizables que se encuentren disponibles en la web para la construcción del proyecto.
- d. Diseño del motor de la solución: Selección de métodos y principios para diseñar el núcleo de la solución.
- e. Diseño de la arquitectura técnica: Elección y diseño de una solución técnica que sea aplicable para la solución del problema, considerando escalabilidad, funcionalidad y requerimientos de negocio.
- f. Diseño de la interfaz de la plataforma: Propuesta de una interfaz acorde a los diseños anteriores.
- g. Implementación de la solución: Codificación de la solución según diseños desarrollados.
- h. Capacitación: Capacitación de los usuarios seleccionados para la evaluación.
- i. Evaluación: Evaluación y obtención de resultados de la solución implementada a través de la realización de una encuesta. Comparación con procedimientos anteriores a la plataforma construida y conclusión en la mejora del proceso de gestión del conocimiento.
- j. Finalización documento: Revisión y conclusión final del documento de Tesis.

El detalle de las tareas realizadas se refleja en la siguiente figura 1.3 de la Carta Gantt:

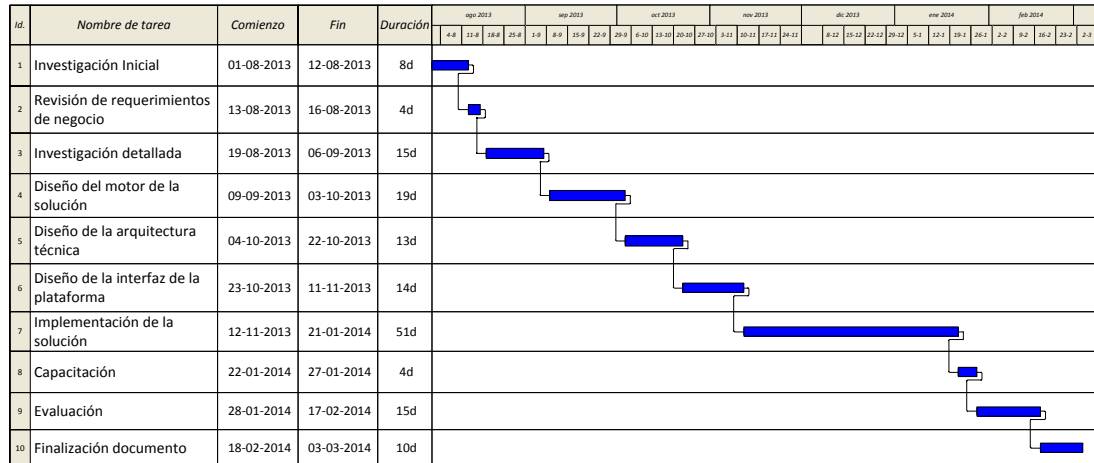


Figura 1.3: Carta Gantt del desarrollo del proyecto de Tesis

## 1.7 Estructura de la tesis

La tesis se estructuró en cada uno de los capítulos que se describen a continuación:

**Capítulo 1:** Introducción en que se explica el contexto de negocio y tecnológico donde se desarrolla el proyecto, los objetivos que se pretenden cumplir, la metodología utilizada y el plan de trabajo seguido.

**Capítulo 2:** Revisión del estado del arte con respecto a la materia que se analiza en esta tesis. Se realiza un estudio de alto nivel de los términos de aprendizaje colaborativo, redes sociales y memoria organizacional, así como la relación entre ellos.

**Capítulo 3:** Especificación y análisis de los requisitos que debe cumplir la plataforma. Se desarrolla en detalle la metodología utilizada y las diferentes plantillas que la componen.

**Capítulo 4:** Muestra la arquitectura general de la plataforma y los principales componentes involucrados en la concepción de la misma.

**Capítulo 5:** Se describe el plan de proyecto, el proceso de despliegue de los servidores y la implementación del sistema, así como todas las tareas necesarias para hacerlo. Se

realizan las pruebas y se pone en marcha la plataforma, para obtener los resultados de la experiencia de uso.

**Capítulo 6:** Conclusiones, verificación del cumplimiento de objetivos, aporte de la tesis, análisis autocrítico del desarrollo del proyecto, sugerencias y lineamientos para futuros trabajos.

## Estado del Arte

### 2.1 Introducción

Los temas relevantes que se deben revisar en el estado del arte para el desarrollo de este trabajo son: 1) memoria organizacional y gestión del conocimiento; 2) aprendizaje colaborativo y; 3) redes sociales. A continuación se presentan las relaciones que existen entre estos tres conceptos, con el objetivo de demostrar que existen correlaciones significativas que se acercan a la finalidad de este proyecto.

### 2.2 Memoria organizacional y gestión del conocimiento

Diversos autores han investigado acerca del conocimiento, a continuación se presentan algunos de ellos y los conceptos claves que serán necesarios para el desarrollo del presente capítulo.

Nonaka y Takeuchi [38], define al conocimiento, como una creencia verdadera y justificada, mientras que Drucker [19], por su parte, destaca que el conocimiento está especializado y hace distinción entre los datos, la información y el conocimiento. Naeve [36], define el conocimiento como fantasías eficientes, las cuales poseen un contexto, un objetivo y un grupo objetivo. Finalmente, Siemens [46], señala que el conocimiento se apoya en un individuo y reside en el colectivo.

Lo anterior, hace evidente que existen diversas definiciones acerca del conocimiento, sin embargo, la mayoría de los textos hacen mención al conocimiento explícito y al conocimiento tácito, los cuales corresponden al conocimiento sistemático que se puede codificar con facilidad, y al conocimiento difícil de formalizar y de comunicar, y que es subjetivo [38]. En este sentido, es importante mencionar, que existen otros autores que definen al conocimiento como una distinción entre lo estructurado y lo menos estructurado [17]; el saber qué y saber cómo [9]; y el conocimiento duro y el blando [27].

La memoria organizacional, es el medio por el cual el conocimiento del pasado puede ser reutilizado en actividades actuales. A través de ella, las organizaciones pueden beneficiarse de su propia experiencia. Conklin [15], indica que la memoria organizacional no es sólo una habilidad para el almacenamiento y preservación, sino también para el intercambio de conocimiento. Spiller y Cunha [47], afirman que la memoria organizacional es formada a base de experiencias por las cuales atraviesa una organización a lo largo de los años, las cuales son acumuladas en sus personas, cultura, procesos y en sus documentos. Ackerman y Halverson [1], por su parte, mencionan que este concepto debe ser socialmente construido, mantenido y dirigido.

En el artículo “Understanding Organizational Memory” [23] se muestran las diferentes formas de llevar a cabo una memoria organizacional a partir de diversos autores, como por ejemplo Pautzke, presenta un modelo de capas de conocimiento accesible o no dentro de la organización; Romero, propone recoger discusiones asincrónicas para preparar las reuniones como base para la construcción de un OM; Walsh y Ungson, consideran lo que se recuerda diferente a cómo es recordado, haciendo hincapié en la importancia de los contenidos; Watson define la OM como una entidad tecnológica, donde las tareas requeridas son muy similares a la de gestión de datos.

De esta forma, para mantener una memoria organizacional existe un proceso: el de gestionar el conocimiento. Éste tiene como fin mantener el conocimiento en el tiempo y lograr que se incremente progresivamente. La gestión del conocimiento, aunque no tiene una definición precisa, apunta a distintas perspectivas y modelos, donde todas las interpretaciones señalan que el conocimiento se identifica, captura, almacena, intercambia y se vuelve a utilizar [21]. La gestión del conocimiento tiene como objetivo, aportar a las personas conocimiento de calidad, con el fin de lograr un máximo rendimiento. Autores más recientes dan mucha más importancia al lado humano en la gestión del conocimiento [2].

Todos los aportes sobre gestión del conocimiento giran en torno a tres componentes principales: 1) Repositorios de información o conocimiento, 2) Redes y comunidades y 3) Expertos y conocedores. A continuación se definen estos tres componentes.

Los repositorios, contienen información explícita y codificada que es fácil de conseguir. Rosenberg [43], indica que los repositorios deben tener una gestión de búsqueda, usabilidad y aportes de fácil manejo. Por su parte, las redes se definen como las conexiones entre entidades para crear un todo integrado, mientras que las comunidades se definen como una agrupación de áreas de interés similares, que permiten interacción y oportunidad de compartir, dialogar y pensar juntos [46]. Un tipo especial de estas



últimas, son las comunidades de práctica, que son conformadas por personas que comparten una preocupación o pasión por algo que hacen y aprenden cómo hacerlo mejor [48]. Finalmente, los expertos y conocedores, son personas que manejan el *know-how*, y que pueden ayudar a lograr los mejores resultados. Rosenberg [43] complementa esta definición, indicando que los expertos son también aquellos asesores que brindan orientación, dan retroalimentación correctiva y evalúan el desempeño.

### 2.3 Aprendizaje Colaborativo

El aprendizaje está directamente relacionado con el conocimiento. Por ejemplo, Allee [5] dice que aprender es la base del conocimiento. Siemens [46] dice que conocer es haber aprendido. Chatti [13], indica que el aprendizaje, al igual que el conocimiento, tiene distintas facetas que consisten en: qué aprender, por qué aprender, cómo aprender, dónde aprender y quién aprende. El aprendizaje es entonces un procedimiento orientado a la acción y a la actividad social, por lo que no es estático ni es un proceso mecánico. Es por el contrario, un flujo muy humano, dinámico y complejo que se asemeja a una estructura orgánica más que a un flujo mecánico [28].

Chatti [12], en el artículo “The future of e-learning: a shift to knowledge networking and social software”, indica que en cuanto a los sistemas de aprendizaje, la mayoría de los contenidos de aprendizaje hoy en día se han diseñado, escrito, entregado y administrado a través de sistemas de gestión de aprendizaje (LMS), sistemas de gestión de contenido (CMS) como cursos en línea estáticos que son envasados y presentados en módulos. Asimismo, el autor indica que en la mayoría de estos casos, el material de aprendizaje basado en papel se convierte simplemente en una forma digital y las aulas en cursos en línea. Estos sistemas de contenido a menudo tratan de entregar contenido compatible con SCORM.

### 2.4 Redes Sociales

O’Reilly [39] define la Web 2.0 como una nueva generación de web que está centrada en el usuario, es abierta, dinámica y con una producción entre pares, y se basa en el intercambio, la colaboración, la inteligencia colectiva y el contenido distribuido. El software social [4], ha surgido como un componente importante del movimiento de la Web 2.0, y está definido como una herramienta para aumentar las habilidades sociales y de colaboración humana, siendo un medio para facilitar la conexión social, y el intercambio de información, además de ser visto como una ecología que habilita un sistema de personas, prácticas, valores y tecnologías en un entorno local concreto. Los

ejemplos más claros de lo anterior, pueden ser las wikis, blogs, RSS, podcasts, etiquetado social, entre otros.

La interacción y la colaboración, son cada vez más intervenidas por el medio social de los entornos basados en web. Los espacios de redes sociales como Facebook, MySpace, Youtube y otros, ofrecen a las personas oportunidades sin precedentes para compartir ideas, creatividad y recibir retroalimentación inmediata de sus compañeros [49]. Las redes sociales permiten a las personas a participar en entornos digitales, y los incitan a compartir experiencias productivas y agradables [37].

Por su parte, las comunidades de práctica [30], sirven para ampliar conocimientos y mejorar la práctica profesional en una especialidad. Deben ser consideradas importantes para la educación, ya que el conocimiento se basa en la interacción entre experiencias individuales y estructuras de conocimiento socialmente definidas [48].

Sumado a lo anterior, la herramienta de software social wiki, se está desarrollando como uno de los favoritos en el sector de educación [37]. Esta herramienta permite una rápida y fácil creación de páginas webs, incluyendo texto, imágenes, enlaces, además de permitir “hacer retroceder” a versiones anteriores si es necesario.

Otro de los ejemplos importantes es el etiquetado social o tags, que constituyen una forma eficaz de promover la construcción de comunidades, donde los usuarios pueden compartir, organizar, descubrir o buscar lo que otros han marcado y encontrar personas con los mismos intereses.

Las tecnologías mencionadas, no son las únicas relacionadas con redes sociales o software social, existen otras, como software que aprovecha la inteligencia colectiva, motores de búsqueda inteligente o buscadores semánticos (a través de datos de redes semánticas, ligadas a la Web 3.0), bases de datos No-SQL (bases de datos documentales), etc.

## **2.5 Relación entre los conceptos**

Una vez investigado los términos “Memoria Organizacional y Gestión del Conocimiento”, “Aprendizaje Colaborativo” y “Redes Sociales”, se estudia el estado del arte respecto de las relaciones que se producen entre estos conceptos, permitiendo

abordar de manera más concreta la situación del problema planteado, como se representa en la figura 2.1.



**Figura 2.1: Relación entre los principales conceptos del estado del arte**

### 2.5.1 Memoria organizacional y aprendizaje colaborativo

En el artículo “The future of e-learning: a shift to knowledge networking and social software” [12] se sostiene que la gestión del conocimiento y la gestión de aprendizaje se pueden ver como las dos caras de la misma moneda, donde ambos conceptos se relacionan en términos de insumos, resultados, procesos, actividades, componentes, herramientas, conceptos y terminologías. Este trabajo muestra la razón de por qué estos conceptos tienen que fusionarse para conseguir la implementación del proyecto, además de que esta argumentación va en sentido amplio, es decir, mostrando la fusión desde dos campos, en lugar de una intersección o relación de complementariedad entre ellos.

Se dice que el aprendizaje y el conocimiento comparten la misma naturaleza, ambos son complejos, dinámicos, humanos y sociales. Al igual que el proceso de creación de conocimiento, el aprendizaje es un proceso orientado a la acción y es una actividad social. En términos de resultados, ambos tienen como objetivos principales la producción de conocimiento y recursos de aprendizaje, la forma de conectar a las personas al conocimiento de calidad, así como de personas a personas, mejorar el rendimiento personal y profesional y aumentar la capacidad cognitiva de las personas, los equipos u organizaciones.

Uno de los principales problemas en las organizacionales es su pérdida del conocimiento, considerando que el conocimiento y la experiencia de los empleados no quedarán a disposición de la organización en el largo plazo. Ghahfarokhi y Zakaria, [22] indican que para retener ese conocimiento son necesarios proyectos concretos en las organizaciones. Una idea similar indica Ozdemir [40], al aseverar que la memoria organizacional es un activo fundamental para las organizaciones, y éstas tienden a perder y olvidar qué y por qué lo han hecho en el pasado.

Pollard [42], indica que la mayoría de las iniciativas de gestión del conocimiento han fracasado, debido a que no ha demostrado ninguna ventaja competitiva para las organizaciones que han invertido en ella. Tales fallos dan lugar a partir de que ven a la gestión del conocimiento como un problema de tecnología y no de interacción social entre las personas [18]. Los sistemas de gestión del conocimiento no están manejando realmente el conocimiento, sino la información, y una gran parte de los que se presentan son simplemente de gestión de información bajo un nuevo sello [29][33][50], donde el concepto de dato se caracteriza generalmente como un conjunto de hechos claros respecto a diferentes eventos; el concepto de información como un conjunto de datos que constituyen un mensaje; y el conocimiento como un conjunto más complejo de información que representa conceptos y experiencias. Sólo una pequeña fracción de un valioso conocimiento es explícita, siendo que existe una enorme cantidad de conocimientos de alta calidad incorporado en las personas, que no es fácilmente expresable.

Así, muchas empresas están descubriendo que el oro en la gestión del conocimiento no es en la distribución de documentos o combinación de bases de datos, sino en el intercambio de ideas y puntos de vista que no están documentados, y que son difíciles de documentar [35]. Éste es el conocimiento tácito, del cual Nonaka y Takeuchi [38] hacen mención, argumentando que es el que reside en las personas y que sólo puede ser creado, sostenido y que puede surgir a través de la socialización, mientras que Siemens [46], dice que hoy en día, el conocimiento sobrevive solamente un período corto de tiempo antes de ser obsoleto.

Al igual que en la gestión del conocimiento, en los contextos de aprendizaje colaborativo, como los LMS y CMS mencionados inicialmente, se pone un fuerte énfasis en la forma de centralizar y estandarizar la experiencia de aprendizaje, sin embargo, la clave no es la interoperatividad ni la reutilización de contenido, sino más bien el apoyo al aprendizaje como un proceso cognitivo y constructivo [32]. Es por eso que muchas de las soluciones de gestión del conocimiento y aprendizaje colaborativo han fracasado,

por enfocarse en el contenido y tecnología, por tanto, éstos deben ser reemplazados con nuevos modelos que reflejen la naturaleza del aprendizaje y el conocimiento del lado humano.

Así, en las organizaciones, la mayoría del conocimiento es tácito, inaccesible en algunos casos y concentrado principalmente en las personas el uso y la distribución del mismo depende en buena medida de las decisiones y los procesos que se hayan definido en cada organización. Los sistemas de gestión del conocimiento en las organizaciones se utilizan para lograr una memoria organizacional incremental, aunque este vínculo no incluye necesariamente un aprendizaje colaborativo como tal. Sólo en algunos casos, como en las redes de conocimiento, es utilizado el aprendizaje colaborativo, por ejemplo, en instancias de investigación y desarrollo, las cuales recuperan la creatividad y el potencial de las personas [34].

Dado lo anterior, en un entorno colaborativo de trabajo utilizado como un sistema de gestión del conocimiento, el proceso de aprendizaje es permanente y evolutivo, tanto en personas como en la organización [6], donde éstas se perfilan según la experiencia y conocimientos, definiendo al grupo de expertos y a los aprendices correspondientes, lo que genera por ende un proceso de memoria organizacional en el corto plazo.

### 2.5.2 Aprendizaje colaborativo y redes sociales

Cross [16] indica que hoy en día las organizaciones ponen énfasis en la forma de obligar a los usuarios a acceder a ambientes cerrados y a que se unan a pequeñas comunidades. Esta forma a menudo falla, por lo que necesitan tomar un enfoque emergente impulsado por las personas, sobre la base de compartir y colaborar en vez de controlar. Para los sistemas de aprendizaje, existe la necesidad de proporcionar ambientes que apoyen la captura eficaz del conocimiento y de la calidad del mismo.

El conocimiento creado por varias personas es más probable que sea de mayor valor que el creado individualmente, es por eso que herramientas de redes sociales tales como clasificación, comentarios, reseñas, críticas y recomendaciones pueden ser muy útiles. Ésta es la inteligencia colectiva que actualmente es utilizada por grandes compañías de internet como Google, Amazon, Youtube, Flickr y que demuestran su éxito. En el artículo “Towards formal evaluation of collaborative work” [51] se mide el aporte que puede realizar un grupo de personas en comparación al aporte de una sola persona, concluyendo que varios miembros pueden contribuir con una mayor riqueza del trabajo en conjunto.

Por otro lado, el aprendizaje colaborativo, como proceso cognitivo, posee un fuerte vínculo dentro de las comunidades virtuales, en las llamadas comunidades de aprendizaje, las cuales consisten en un grupo de personas que son vinculadas por intereses comunes y que están comprometidas con el proceso de aprendizaje continuo por medio de herramientas tecnológicas para compartir [45], que incrementan la eficiencia y eficacia del proceso de aprendizaje general, aprovechando el espacio electrónico [31], orientando la dinámica de interacción en la co-construcción del conocimiento entre docentes y alumnos, no sólo intercambiando dudas, información y conocimiento, sino manteniendo una fuerte interacción social que permite en esencia hacer más dinámico el proceso. El conocimiento se genera a través de la interacción de los miembros y por tanto es compartido socialmente [11]. Algunas de las características de estas comunidades de aprendizaje son los glosarios, muros, foros, chats, espacios de trabajo, preguntas frecuentes [11] y donde más se han utilizado estos tipos de herramientas ha sido en el área de la educación [45] como una forma de apoyo al proceso educativo.

Cabe mencionar, la existencia de estudios [20] que reflejan que el uso de redes sociales de ocio, aunque no estén orientadas al estudio, puede ser un apoyo para el aprendizaje colaborativo, otorgando beneficios adicionales a docentes y alumnos. También se indica que son utilizadas otras herramientas de redes sociales de apoyo, como blogs y wikis, donde se potencia la participación de los alumnos, la motivación por el estudio, la interacción y la creatividad.

Otro estudio [14] que evalúa el rendimiento de aprendizaje en una comunidad CSCL (Computer-Supported Collaborative Learning) utilizando un análisis de redes sociales, concluye que los alumnos poseen una alta disposición para comunicarse, ventaja que contribuye en el rendimiento respecto al aprendizaje y a la construcción del capital intelectual.

Es de vital importancia tener en cuenta que la participación de las personas en una plataforma virtual de aprendizaje debe tener como fin incrementar su uso en el tiempo y no caer en el abandono de ésta. Es por eso que los autores del artículo “Boosting Participation in Virtual Communities” [24] proponen una serie de principios y funcionalidades para impulsar la participación con el fin de lograr un alto grado de actividad en la comunidad virtual, promoviendo la contribución de aportes. Se basan en principios y teorías de distintos autores para proponer un modelo basado en tres secciones: modelo de red social, tecnologías y 5 factores sociales: rankings, moderación de pares, desafíos, matchmaking y notificaciones.

Es entonces que los autores mencionados, se centran en probar los 5 factores, utilizando un sistema de aprendizaje colaborativo, basado en una plataforma con características de red social. Este modelo propuesto coincide y es aplicable a la solución que se busca construir para nuestro caso, ya que es fundamental la participación de las personas para contribuir a la mejora del conocimiento empresarial y su aprendizaje.

Adicionalmente, se puede rescatar que este modelo podría ayudar a la mejora en el marketing interno, en la gestión empresarial y en el aprendizaje organizacional. También promueve la utilización de nuevas herramientas tecnológicas, dispositivos móviles y nuevos estándares web.

Finalmente, es significativo mencionar que los cambios en el modo de aprendizaje en red han sido consecuencia de la incapacidad por parte de la educación formal para responder a las nuevas necesidades de las generaciones actuales y al uso de las nuevas tecnologías de la información [31], situación que se aplica también a las organizaciones.

### 2.5.3 Memoria organizacional y redes sociales

Diversos investigadores hacen hincapié en el aspecto social del conocimiento y aprendizaje [38][48]. Lave y Wenger [30] introducen las comunidades de práctica como vehículos ideales para aprovechar el conocimiento tácito y el aprendizaje. Dentro de estas comunidades los novatos pueden aprender de los expertos a participar, por lo que se mueven del aislamiento hacia la participación plena, lo que ha sido demostrado en investigaciones recientes que constatan la visión del concepto de aprendizaje como un proceso social [41]. En consecuencia, para tener alguna posibilidad de éxito, el enfoque de gestión del conocimiento y aprendizaje necesita reconocer el aspecto social y dar un fuerte énfasis en redes de conocimiento y construcción de comunidades para aprovechar de crear, mantener y compartir el conocimiento de forma colaborativa, a través de la participación, el diálogo, el debate, la observación y la imitación. Para construir este tipo de comunidades y redes en las organizaciones, hay que involucrar a todas las personas sin importar si desempeñan distintos roles pero que compartan intereses comunes [46].

Saidatul y Kamaruzaman [25], aseguran que las TI son un mecanismo sólido para gestionar la memoria organizacional. Éstos ayudan a retener, compartir y recuperar más fácilmente el conocimiento almacenado, lo que la hace permanente. El conocimiento es registrado, indexado, asegurado y organizado de tal manera que sea accesible.

Un problema presentado por Babcock [7], dice que las personas tienden a no compartir sus conocimientos valiosos por dos razones: creer que el conocimiento es poder y por falta de confianza entre los pares. Por lo tanto, un requisito fundamental para el intercambio de conocimiento es introducir una nueva cultura que permita que éste fluya y no castigue las iniciativas de colaboración. Bobcock [7] afirma que el requisito principal es la confianza, dado que las relaciones son las que la fomentan. Como solución recomienda la creación de oportunidades, donde las personas puedan conocer e interactuar en contextos formales e informales, que puedan desarrollar relaciones fácilmente aplicables a los entornos de redes sociales. Siemens [46] apoya esta tesis, afirmando que es necesario el contacto social para fomentar la confianza, en entornos seguros y protegidos.

En un estudio [8], se presenta que las relaciones son importantes para la obtención de información y que la interacción social es un vehículo para la creación del conocimiento, dado que hace más eficiente la búsqueda de la información por medio de ciertas características: el saber (a sabiendas de los que otra persona sabe), el valor (lo que otra persona sabe en relación con uno) y el acceso (tener acceso oportuno al pensamiento de otra persona).

Por otro lado, el capital intelectual dentro de una organización, es el activo intelectual que se maneja en la memoria organizacional, el cual se distribuye por medio de la red, convertido en conocimiento [10]. Existen diversas herramientas de gestión del capital intelectual enfocadas en la interacción social dentro de las organizaciones, ya sea a través de metodologías, estrategias de conocimiento, intercambio, de comunicación y gestión corporativa o de evaluación [10], involucrando la gestión del conocimiento y utilizando de manera parcial una interfaz de red social para lograr la activa participación de los colaboradores dentro de la organización.

De esta manera la Web 2.0 y las plataformas sociales cumplen con los requisitos de gestión del conocimiento, donde se puede crear contenido compartido y reutilizarlo con el fin de mejorar la memoria organizacional, de modo de aumentar las habilidades sociales y de colaboración humana. Apoyan las redes de conocimiento y la construcción de comunidades. Por ejemplo, las wikis proporcionan una oportunidad para la creación de contenido colaborativo e interacción social. Otro ejemplo son los blogs, que pueden generar una interesante discusión en sus comentarios, entre autores y lectores. Más ejemplos son los etiquetados sociales o tags, donde usuarios pueden compartir,



organizar, descubrir o buscar lo que otros han marcado y encontrar personas con los mismos intereses.

## 2.6 Contribución del capítulo

Para finalizar este capítulo, se puede concluir de este estado del arte, que los autores indican que el conocimiento se puede dividir en explícito, que es fácil de obtener, y en tácito, que es subjetivo y difícil de formalizar, y es éste último donde hay que enfocarse en tratar de obtenerlo. Para mantener una memoria organizacional, se debe intercambiar el conocimiento, debe ser socialmente construido, mantenido y dirigido, es por eso que hay que darle más importancia al lado humano en la gestión del conocimiento que debe girar en torno a tres componentes: repositorios, comunidades y conocedores. Por otro lado, el aprendizaje está directamente relacionado con el conocimiento, ya que es un procedimiento orientado a la acción, es una actividad social y debe ser un proceso cognitivo y constructivo. Se concluye también, que para el aprendizaje, las organizaciones deben poner énfasis sobre la base de compartir y colaborar, en vez de controlar y obligar a los usuarios a acceder a ambientes cerrados y que se unan a pequeñas comunidades. A partir de aquí, es donde se visualiza que las redes sociales pueden tener una aplicación importantísima en la gestión del conocimiento, ya que se basan en el intercambio, la colaboración humana y contenido distribuido. Se ve como una ventaja para poder manejar el conocimiento, y no la información como lo han hecho las iniciativas que han fracasado. Justamente se apunta al intercambio de ideas, puntos de vista que no están documentados y que son difíciles de documentar. Existen diversos métodos sociales y principios que pueden incrementar el uso en el tiempo con el fin de evitar el abandono de estas plataformas. Otra de las características que aporta el contacto social es la confianza, con el fin de que las personas aporten los conocimientos valiosos y donde puedan participar en contextos formales e informales.

## Análisis de Requisitos

### 3.1 Visión general del proyecto

Como este proyecto de tesis está basado en el caso especial de Komatsu, el análisis de requisitos se basará en las necesidades de la compañía, buscando solución a los problemas recurrentes que tiene, pero sin dejar de considerar que esta solución puede ser aplicable a otros casos. Para comenzar, se realiza una analogía entre los conceptos principales de este proyecto, con el fin de poder detectar correctamente una serie de requisitos fundamentales. Posteriormente se aplicará la metodología CommonKADS orientado a sistemas de gestión del conocimiento, para realizar un análisis de requisitos y obtener correctamente los requisitos esenciales de la plataforma.

### 3.2 Analogía entre redes sociales y sistemas de gestión del conocimiento

En el artículo “La Empresa Creadora de Conocimiento” de Ikujiro Nonaka [38], se indica que una de las mejores formas para convertir el conocimiento tácito en explícito, es utilizar el bagaje de lenguaje figurativo y simbolismos, con el fin de articular ideas e intuiciones. Uno especialmente importante es la metáfora, que es un método distintivo de percepción, es una forma en que las personas de diferentes contextos y con distintas experiencias puedan entender intuitivamente algo mediante el uso de la imaginación y los símbolos sin necesidad de análisis o generalización. El concepto utilizado para este proyecto, que perfectamente podría ser la metáfora, es el de “red social del conocimiento”. Aunque puede sonar contradictorio tener una red social dentro de la empresa (ya que generalmente éstas son utilizadas para el ocio) y unirla con la gestión del conocimiento, se espera que dé como resultado despertar el interés de las personas en estas redes, al mismo tiempo que se aprovecha la incorporación de un sistema que esté gestionando la información.

Según Nonaka, después de la metáfora, el paso que sigue es la analogía, un proceso estructurado de conciliar contradicciones y distinciones o aclarar las semejanzas y

diferencias entre las dos ideas de una frase. Las contradicciones implícitas en las metáforas se armonizan mediante la analogía. Es por eso que a continuación se presenta un cuadro con la analogía entre las redes sociales y los sistemas de gestión del conocimiento, basada en la propia experiencia del autor de esta Tesis, a modo de hipótesis, con el fin de buscar que las personas se interesen en utilizarlos.

| <b>¿Qué características o experiencias sobre las redes sociales gustan a las personas?</b>  | <b>¿Cómo se pueden utilizar estas características para aprovecharlas y crear un sistema de conocimiento atractivo e interesante?</b>  |
|---|---|
| Contar su vida, reflejar el "quién soy" (Identidad).  | Que las personas puedan crear sus propios perfiles y adecuarlos según sus gustos, experiencias, habilidades, intereses, y puedan hacer las publicaciones que estimen conveniente, tanto en el ámbito laboral como personal, sin salirse del contexto de la organización.  |
| Poder compartir el estado o decir cómo se siente la persona.  | Debido a que una característica atractiva es poder demostrar el "sentir" de las personas, el sistema podría permitir compartir el estado actual de las personas, con el fin de hacer un sistema más "humanizado".   |
| Hacer amigos, encontrar personas afines, poder conectarse con quienes "son como yo". Poder conocer a los demás y sus gustos.  | Es esencial que las personas puedan crear sus propios "mundos", así como conformar grupos para tareas en específico. También para encontrar personas que tengan mismos intereses o habilidades, de modo de poder resolver problemas en conjunto.  |
| Sentirse conectado al instante. Permitir que las personas sientan cercanía de una forma casi inmediata.   | En la organización, esto representa sentirse cercano al lugar de trabajo, generando "identidad corporativa personal", logrando que los empleados se sientan con la "camiseta puesta", conectados y cercanos.  |
| Lograr que las personas expresen gustos, afinidades, opiniones que en ocasiones no se expresan en interacciones personales, dado que, una red social los hace sentir menos presionados, más | Esto en la empresa, sirve para recabar información acerca de los empleados, sus gustos, necesidades, quejas, opiniones y sugerencias de una forma más dinámica, con más libertad de expresión, podría decirse más sincera, y que además posee el plus de que la información quedará registrada, y no se perderá fácilmente. |

|  |  |
|--|--|
| libres en su nivel de expresión.   |  |
| Compartir información de variados tipo con los demás.                          | En la gestión del conocimiento es imprescindible compartir documentos, información y conocimientos, a través de repositorios, grupos, blogs o cualquier método disponible dentro del sistema.  |
| Personalizar el perfil a gusto.  | Una manera de atraer a las personas, es la posibilidad de personalizar el perfil, de modo de ser lo más representativo en la plataforma (persona), y no ser un usuario virtual como en las intranets corporativas.   |
| Publicar fotos y videos.   | Un método básico de atraer a las personas es la posibilidad de publicar fotos o videos de reuniones, asambleas, viajes de trabajo, fiestas de la compañía, o cualquier tipo de evento.   |
| Poder expresarse de forma libre.   | Si bien las personas tendrían que tener un comportamiento adecuado, podrían estar de igual a igual operarios, técnicos, cargos medios y cargos altos, y podrán participar o no de todas las iniciativas existentes, poseyendo libertad absoluta de participación en la plataforma. |
| Recibir noticias de su gusto y estar al día de toda la comunidad que lo rodea. | Poder suscribirse a distintos grupos o aplicaciones de modo de recibir noticias de interés, siempre con la posibilidad de poder dar retroalimentación. Adicionalmente existirán recomendaciones de temas que se irán generando de acuerdo a los perfiles de cada persona.          |
| Navegar con facilidad de uso y que sea una plataforma intuitiva.               | De la misma forma que las redes sociales que existen al día de hoy, el sistema tendría que poseer una interfaz amigable y ser intuitiva en su navegación.  |
| Ser agradable a la vista y cómodo para navegar en ella.                        | Permitir que se diseñe una interfaz acorde a la compañía, pero sin perder de vista un diseño agradable y comodidad en la navegación. Donde se ofrezca una interfaz moderna y actual, con al menos tecnología HTML5 y CSS3.   |
| Posibilidad de invitar personas a eventos / páginas / juegos / etc.            | Con el fin de aumentar la interacción, el sistema tendrá la posibilidad de invitar las personas a reuniones, ver documentos, comentar, aportar puntos de vista, crear conocimiento, utilizar aplicaciones, entre todas las   |

|  |  |
|--|--|
|  | posibilidades que ofrezca la plataforma.   |
| Poder hacer publicidad o vender algún producto.  | El sistema no se utilizará para vender productos, pero si podrá servir para hacer publicidad en temas de los que las personas tengan interés, por ejemplo iniciativas, puntos de vista, campañas, discusiones, necesidades de cualquier tipo u otras que soporte la plataforma.              |
| Tener una diversidad de usos dentro de la red social (chat, email, noticias, compras, ver galerías, comentar). | Una de las características que debiera ofrecer esta solución, sería la de integrar diversas funcionalidades, al igual que las redes sociales, tales como chat, email, noticias, encuestas, galerías, plataformas de discusión, de generación de conocimiento, dentro de la misma plataforma. |
| Ser parte de una comunidad que está de moda y es masiva.   | Una iniciativa es lograr que la plataforma tenga un uso masivo y se use tanto por lo atractivo como por la efectividad que logre dentro de la organización. Para lograrlo se deberá poner énfasis en un modelo especial que abarque estos requisitos.  |
| Jugar en línea o utilizar aplicaciones de ocio.  | Para incentivar el uso, podría existir la posibilidad de incorporar elementos de ocio, no distractores, como por ejemplo encuestas o concursos laborales. Otra forma de entretenimiento será un sistema estadístico de ranking y de premios para incentivar el aporte de conocimiento.       |
| Poder conectarse a través de smartphones o tablets.  | Una buena manera para que la plataforma sea atractiva, sería poder soportarla en distintos dispositivos móviles de manera cómoda y fluida.   |
| Poder participar en concursos.   | Los premios, son una de las mejores formas de incentivar el uso en la plataforma, por lo que se incentivaría el uso de éstos, mediante la publicación en los listados de las personas que más aportan, o que tienen mejor evaluación en los contenidos aportados.                            |
| Botón me gusta.  | Una forma particular de influir en los temas que se estén viendo, son los botones de me gusta y no me gusta, con el fin de mejorar los ranking dentro de todos los contenidos ofrecidos.   |

**Tabla 3.1: Tabla de analogía entre redes sociales y SGC.**

Finalmente, el autor del artículo indica que después de la metáfora y la analogía, el último paso en el proceso de creación de conocimiento, es el de la elaboración de un modelo, debido a que se puede concebir en forma mucho más inmediata. Este modelo será parte del desarrollo del presente trabajo de tesis.

### 3.3 Aplicación de metodología

Siguiendo con la metodología aplicada para el proyecto, CommonKADS, se desarrollaron las plantillas OM-1, OM-2, OM-3, OM-4 y OM-5 para poder determinar los requisitos de la compañía y encontrar las aplicaciones que tendrá la solución desarrollada.

#### 3.3.1 Plantilla OM-1

La siguiente tabla contiene el desarrollo de la plantilla OM-1 de CommonKADS:

| Modelo Organizacional     | OM-1: Problemas y oportunidades   |
|---------------------------|---|
| Problemas y oportunidades | <ul style="list-style-type: none"> <li>• Existen 6 empresas dentro del holding que funcionan independiente una de otra, por lo que la comunicación y el manejo de conocimiento no es transversal.</li> <li>• Los trabajadores se encuentran distribuidos en distintas faenas, oficinas o se encuentran en traslados frecuentes, por lo que la relación entre ellos es poco estable.</li> <li>• Existe una pérdida de conocimiento importante debido a la alta rotación del personal capacitado generado por la competencia en el mercado minero.</li> <li>• La compañía necesita generar una base de conocimiento común para todas las empresas del holding con el fin de transferir de forma rápida y efectiva el conocimiento adquirido.</li> <li>• Se requiere que se tenga mayor control y visibilidad sobre las tareas que competen a las distintas empresas.</li> </ul> |

|                                       |   |
|---------------------------------------|---|
|                                       | <ul style="list-style-type: none"> <li>• La compañía requiere que las personas tengan mayor interacción para hablar de temas en específico.</li> <li>• Existe la necesidad de innovación permanente, en un mercado altamente competitivo.</li> <li>• Existe la necesidad de identificar problemas, estrategias y soluciones.</li> <li>• Existe gran cantidad de información estructurada y no estructurada que poseen los trabajadores, la que no se encuentra ordenada.</li> <li>• Los sistemas que existen actualmente son manejados por pequeños grupos de usuarios y son poco incentivados.</li> </ul>  |
| <p><b>Contexto Organizacional</b></p> | <ul style="list-style-type: none"> <li>• Visión: Ser la primera opción para los clientes de minería y construcción en Latinoamérica.</li> <li>• Misión: Ser la organización responsable de conocer y entender las necesidades de los clientes de minería y construcción en Latinoamérica, asegurando que reciban las soluciones necesarias a través de una red de distribución de excelencia.</li> <li>• Contexto: Fundada en 1921, Komatsu Limited, compañía de origen japonés y de enorme prestigio en la fabricación de equipos para minería, construcción y forestal, es uno de los líderes mundiales en este mercado.</li> </ul> <p>Luego de estar presente en Chile por más de 40 años bajo el alero de empresas distribuidoras, en 1999 Komatsu se establece en nuestro país, creando el Holding Komatsu, incorporando experiencia, tecnología de vanguardia y volumen organizacional.</p> <p>En un escenario donde la actividad minera se constituye como una de las principales actividades económicas del país, Komatsu se ha</p> |

|                          |  |
|--------------------------|--|
|                          | <p>consolidado como uno de los principales socios estratégicos en proveer de equipos, repuestos y servicios a la pequeña, mediana y gran minería chilena.</p> <p>Komatsu ha logrado una importante presencia de marca en las grandes mineras chilenas, transformándose además en la empresa con mayor experiencia en contratos de servicios de mantención a flotas de equipos mineros, en particular en nuevos proyectos.</p> <ul style="list-style-type: none"> <li>• Pilares Estratégicos:             <ul style="list-style-type: none"> <li>○ Entender las necesidades de nuestros clientes</li> <li>○ Proveer soluciones flexibles</li> <li>○ Eficiencia integral de nuestros procesos</li> <li>○ Sustentabilidad (crecimiento)</li> <li>○ Cuidado y desarrollo de nuestras personas</li> </ul> </li> <li>• Valores:             <ul style="list-style-type: none"> <li>○ Responsabilidad</li> <li>○ Compromiso</li> <li>○ Respeto</li> <li>○ Seguridad</li> <li>○ Ética</li> </ul> </li> </ul> |
| <p><b>Soluciones</b></p> | <ul style="list-style-type: none"> <li>• Construcción de una plataforma para la gestión del conocimiento basada en arquitectura colaborativa y con interfaz de red social, que cuente con las siguientes características:             <ul style="list-style-type: none"> <li>○ Motor de indexación que permita extraer estructura semántica (<i>taggeos</i> automáticos, uso de ontologías, técnicas de recomendación) e incluya un buscador semántico.</li> <li>○ Incluya un sistema de notificación y de estadísticas.</li> <li>○ Permita la gestión de documentos.</li> </ul> </li> </ul>   |



|  |  |
|--|--|
|  | <ul style="list-style-type: none"> <li>o Contenga diversas aplicaciones integradas dentro de la misma plataforma, tales como de wikis, foros, blogs, chats, correo, calendarios, etc.</li> </ul> |
|--|--|

**Tabla 3.2: Tabla de Plantilla OM-1 de CommonKADS.**

### 3.3.2 Plantilla OM-2

La siguiente tabla contiene el desarrollo de la plantilla OM-2 de CommonKADS:

| Modelo Organizacional           | OM-2: Aspectos a considerar  |
|---------------------------------|--|
| <p><b>Estructura</b></p>        | <p>En la figura 3.1 se representa la estructura organizacional del Holding.</p> <div style="text-align: center;"> <pre> graph TD     A[Komatsu Cummins Chile] --&gt; B[Komatsu Chile]     A --&gt; C[Komatsu Holding South América]     A --&gt; D[Komatsu Arriendo]     A --&gt; E[Distribuidora Cummins Chile]     A --&gt; F[Reman Center]             </pre> </div> <p><b>Figura 3.1: Estructura organizacional del Holding.</b></p> |
| <p><b>Procesos Críticos</b></p> | <ul style="list-style-type: none"> <li>• Generación de reportes e informes</li> <li>• Búsqueda de documentación y archivos técnicos</li> <li>• Discusión y solución de problemas técnicos</li> <li>• Asignación y seguimiento de tareas específicas a ingenieros</li> <li>• Actividades de aseguramiento de calidad</li> <li>• Generación de nuevas ideas</li> </ul>   |
| <p><b>Personal</b></p>          | <ul style="list-style-type: none"> <li>• Gerentes</li> <li>• Expertos</li> <li>• Ingenieros de apoyo y analistas (internos)</li> <li>• Ingenieros de soporte, especialistas de faena y técnicos (externos)</li> </ul>  |

|                            |  |
|----------------------------|--|
|                            | <ul style="list-style-type: none"> <li>• Contactos de distribuidores</li> <li>• Contactos de fábrica</li> </ul>  |
| <b>Recursos</b>            | <ul style="list-style-type: none"> <li>• Intranet compañía</li> <li>• Sistemas de fábrica</li> <li>• Sistemas de distribuidores</li> <li>• Sistema de garantías</li> <li>• Biblioteca virtual</li> <li>• CRM</li> <li>• Sistema de helpdesk</li> <li>• Sistema de comunicación interno</li> </ul>  |
| <b>Conocimiento</b>        | <ul style="list-style-type: none"> <li>• Conocimientos de fábrica</li> <li>• Conocimientos de gestión de garantías</li> <li>• Conocimientos que poseen distribuidores</li> <li>• Informes mensuales y anuales</li> <li>• Diversos sitios de la compañía con información</li> <li>• Conocimiento tácito de los expertos</li> <li>• Bases de datos corporativas</li> </ul>   |
| <b>Cultura y potencial</b> | <ul style="list-style-type: none"> <li>• Forma de comunicación muy informal, a través de chats o correo electrónico, a través de reuniones o llamadas telefónicas.</li> <li>• Información muy particular o muy técnica es trabajada a través de algunos sistemas de información.</li> <li>• Reportes son generados para obtener conclusiones a grandes rasgos.</li> <li>• Mucho conocimiento es poseído por los expertos o técnicos, pero no es transferido a otras personas, debido a que no se encuentra documentado.</li> </ul> |

Tabla 3.3: Tabla de Plantilla OM-2 de CommonKADS.

### 3.3.3 Plantilla OM-3

La siguiente tabla contiene el desarrollo de la plantilla OM-3 de CommonKADS:

| Modelo Organizacional |   | OM-3: Descomposición del proceso  |                 |   |             |             |
|-----------------------|---|---|-----------------|---|-------------|-------------|
| #                     | Tarea   | Realizada por   | ¿Dónde?         | Activo de K   | K intensivo | Importancia |
| 1                     | Generación de reportes e informes             | Gerentes, expertos, ingenieros de apoyo                                   | Todo el holding | Conocimientos de fábrica, conocimientos de gestión de garantías, informes mensuales y anuales, diversos sitios de la compañía con información, conocimiento tácito de los expertos, bases de datos corporativas | Sí          | Alta        |
| 2                     | Búsqueda de documentación y archivos técnicos | Gerentes, expertos, ingenieros de apoyo, ingenieros de soporte, contactos | Todo el holding | Conocimientos de fábrica, conocimientos de gestión de garantías, informes mensuales y anuales, diversos sitios de la compañía con información, bases de datos corporativas                                      | No          | Alta        |
| 3                     | Discusión y solución de problemas técnicos    | Expertos, ingenieros de apoyo, ingenieros de soporte, contactos           | Todo el holding | Conocimientos de fábrica, conocimientos de gestión de garantías, conocimientos que poseen distribuidores, informes mensuales y anuales, diversos sitios de la compañía con información, conocimiento            | Sí          | Muy alta    |

|   |  |   |                 |  |    |       |
|---|--|---|-----------------|--|----|-------|
|   |  |   |                 | tácito de los expertos, bases de datos corporativas  |    |       |
| 4 | <b>Asignación y seguimiento de tareas específicas a ingenieros</b> | Gerentes, ingenieros de apoyo   | Todo el holding | Conocimientos de fábrica, conocimientos de gestión de garantías, informes mensuales y anuales, diversos sitios de la compañía con información, conocimiento tácito de los expertos, bases de datos corporativas  | No | Media |
| 5 | <b>Actividades de aseguramiento de calidad</b>                     | Expertos, ingenieros de apoyo, ingenieros de soporte                      | Todo el holding | Conocimientos de gestión de garantías, conocimientos que poseen distribuidores, diversos sitios de la compañía con información, conocimiento tácito de los expertos, bases de datos corporativas   | No | Media |
| 6 | <b>Generar nuevas ideas</b>  | Gerentes, expertos, ingenieros de apoyo, ingenieros de soporte, contactos | Todo el holding | Conocimientos de fábrica, conocimientos de gestión de garantías, conocimientos que poseen distribuidores, informes mensuales y anuales, diversos sitios de la compañía con información, conocimiento tácito de los expertos, bases de datos corporativas | Sí | Alta  |

Tabla 3.4: Tabla de Plantilla OM-3 de CommonKADS.

### 3.3.4 Plantilla OM-4

La siguiente tabla contiene el desarrollo de la plantilla OM-4 de CommonKADS:

| Modelo Organizacional                         |  | OM-4: Activos del conocimiento   |                  |                  |                   |                    |
|---|--|--|------------------|------------------|-------------------|--------------------|
| Activo del conocimiento                       | Lo tiene   | Usado en   | ¿Forma correcta? | ¿Lugar correcto? | ¿Tiempo correcto? | ¿Calidad correcta? |
| <b>Conocimientos de fábrica</b>               | Expertos, ingenieros de apoyo, contactos de fábrica        | Generación de reportes e informes, búsqueda de documentación y archivos técnicos, discusión y solución de problemas técnicos, generación de nuevas ideas | NO               | NO               | NO                | NO                 |
| <b>Conocimientos de gestión de garantías</b>  | Expertos, ingenieros de apoyo, contactos de distribuidores | Generación de reportes e informes, búsqueda de documentación y archivos técnicos, discusión y solución de problemas técnicos, generación de nuevas ideas | NO               | NO               | SI                | SI                 |
| <b>Conocimiento que poseen distribuidores</b> | Contactos de distribuidores                                | Generación de reportes e informes, búsqueda de documentación y archivos técnicos, discusión y solución de problemas técnicos, generación de              | NO               | NO               | NO                | NO                 |

|   |   |  |    |    |    |    |
|---|---|--|----|----|----|----|
|   |   | nuevas ideas   |    |    |    |    |
| <b>Informes mensuales y anuales</b>                           | Gerentes, expertos, ingenieros de apoyo                                   | Generación de reportes e informes, búsqueda de documentación y archivos técnicos, discusión y solución de problemas técnicos, asignación y seguimiento de tareas específicas a ingenieros, actividades de aseguramiento de calidad, generación de nuevas ideas | NO | SI | NO | NO |
| <b>Diversos sitios de la compañía con información valiosa</b> | Gerentes, expertos, ingenieros de apoyo, ingenieros de soporte, contactos | Búsqueda de documentación y archivos técnicos, discusión y solución de problemas técnicos, actividades de aseguramiento de calidad, generación de nuevas ideas   | NO | SI | NO | NO |
| <b>Conocimiento tácito de los expertos</b>                    | Expertos  | Generación de reportes e informes, discusión y solución de problemas técnicos, asignación y seguimiento de tareas específicas a ingenieros, actividades de aseguramiento de calidad, generación de nuevas ideas  | NO | NO | SI | SI |

|                                    |                               |   |    |    |    |    |
|------------------------------------|-------------------------------|---|----|----|----|----|
| <b>Bases de datos corporativas</b> | Expertos, ingenieros de apoyo | Generación de reportes e informes, asignación y seguimiento de tareas específicas a ingenieros, actividades de aseguramiento de calidad, generación de nuevas ideas | SI | NO | SI | SI |
|------------------------------------|-------------------------------|---|----|----|----|----|

**Tabla 3.5: Tabla de Plantilla OM-4 de CommonKADS.**

### 3.3.5 Plantilla OM-5

La siguiente tabla contiene el desarrollo de la plantilla OM-5 de CommonKADS:

|                               |  |
|-------------------------------|--|
| <b>Modelo Organizacional</b>  | <b>OM-5: Análisis de viabilidad</b>  |
| <b>Viabilidad empresarial</b> | <ul style="list-style-type: none"> <li>• Beneficios: Permitir capturar, organizar y almacenar explícitamente el conocimiento tácito de las personas, convirtiéndolo en competencias corporativas y ventajas competitivas, ahorrando en costos y recursos.</li> <li>• Valor agregado: Se espera acelerar el traspaso de la información que tiene valor, facilitando el intercambio y flujo de ideas. Crear valor para los clientes internos y clientes externos que dependen de la compañía. Promueve las actividades de mejora continua. Formar futuras generaciones. Centralizar y disponer la información técnica esencial en una sola plataforma.</li> <li>• Coste esperado: Se consideran los costos asociados a la mantención y alojamiento de la plataforma desarrollada, tanto como los costos relacionados con la capacitación de las personas.</li> </ul> |

|                                |   |
|--------------------------------|---|
|                                | <ul style="list-style-type: none"> <li>• Cambios en la organización: Los departamentos tendrán que adecuar algunos procedimientos de gestión para poder participar de esta plataforma.</li> </ul>   |
| <b>Viabilidad técnica</b>      | <ul style="list-style-type: none"> <li>• La implantación de esta plataforma cuenta con viabilidad técnica debido a que será desarrollada durante el presente trabajo de tesis, y no posee requisitos especiales para su uso.</li> </ul>   |
| <b>Viabilidad del proyecto</b> | <ul style="list-style-type: none"> <li>• La organización no será responsable del proyecto, ya que se desarrollará en el presente trabajo de tesis.</li> <li>• Para que el proyecto sea viable se requiere de los siguientes puntos:             <ul style="list-style-type: none"> <li>○ Será necesario el compromiso de participación de parte de la organización, con al menos 10 personas participando activamente como mínimo.</li> <li>○ Será necesaria la difusión de la plataforma dentro de la compañía con el fin de promover su uso, una vez finalizada la construcción.</li> </ul> </li> </ul> |
| <b>Acciones propuestas</b>     | <ul style="list-style-type: none"> <li>• Desarrollar el presente trabajo de tesis, construyendo una plataforma de gestión del conocimiento con características de redes sociales y sistemas colaborativos.</li> <li>• Seguir la planificación definida en el presente trabajo.</li> <li>• Obtener el compromiso de una cantidad de entre 10 y 20 personas.</li> <li>• Hacer la evaluación final de la plataforma para iniciar su uso en toda la compañía.</li> </ul>  |

Tabla 3.6: Tabla de Plantilla OM-5 de CommonKADS.



### 3.4 Requisitos funcionales

Una vez desarrolladas las plantillas de la metodología CommonKADS, se pueden desprender los requisitos funcionales, los que se detallan a continuación:

- a) **Repositorio común entre el holding de empresas:** con el objetivo de permitir generar una base de conocimientos y poder agrupar la información estructurada.
- b) **Interacción entre personas:** permitir que las personas se puedan relacionar en torno de algún tema de interés específico donde pueda aprender o resolver algún problema.
- c) **Herramientas para manejar el conocimiento:** con el fin de transferir de forma rápida y efectiva el conocimiento no estructurado adquirido entre las personas y poder incrementarlo.
- d) **Estructuración de contenido:** permitir que sea posible estructurar todo el contenido en forma de algún tipo de mapa del conocimiento, mantenido por las mismas personas de la compañía.
- e) **Integración con otras aplicaciones:** permitir que la plataforma sea compatible o integrable con más aplicaciones internas o externas, de tipo workflow, blogs, chats, correo, calendarios, etc.
- f) **Motor de indexación:** que permita extraer estructura semántica del contenido generado dentro de los módulos, tales como etiquetas o información relevante que pueda ser estructurada automáticamente.
- g) **Buscador semántico:** es imprescindible contar con un buscador completo que contemple una búsqueda a través de etiquetas y no sólo texto, además de búsqueda por categorización.
- h) **Sistema de recomendación:** para poder incentivar el uso de la plataforma y generar más interacción, es necesario algún tipo de sistema que recomiende información relevante para la persona.

- i) **Módulo de generación de ideas:** es importante contar con una funcionalidad que permita generar nuevas ideas e incentive a las personas a participar más en generación de conocimiento.
- j) **Sistema estadístico:** es necesario contar con un sistema que muestre las diversas estadísticas de las personas y el sistema en sí, con el fin de poder detectar las debilidades y fortalezas de uso.
- k) **Repositorios de documentos:** con el fin de almacenar diferentes tipos de archivos, se debe contar con repositorios.
- l) **Plataforma de discusión (grupos):** con el fin de poder compartir el conocimiento y aportar distintos puntos de vista.
- m) **Preguntas y respuestas:** módulo que permite hacer una pregunta, y que diversas personas puedan dar sus respuestas y poder seleccionar la mejor respuesta.
- n) **Plataforma wiki:** módulo que funcione casi como un diccionario, con el conocimiento tácito generado en la plataforma y poder traspararlo para dejarlo como explícito.

### 3.5 Requisitos no funcionales

Se identificaron los siguientes requisitos de calidad que la plataforma debe cumplir:

- a. **Confidencialidad:** La información y contenido manejado en la plataforma debe estar protegido y debe tener acceso sólo a las personas o grupos a quienes se les ha dado.
- b. **Integridad:** La información y contenido manejado en la plataforma será objeto de cuidadosa protección contra el mal uso de ésta.
- c. **Disponibilidad:** A los usuarios autorizados se les debe garantizar el acceso a la información y funcionalidades disponibles en la plataforma, con el fin de mantener la seguridad y los datos en el momento deseado.

- d. **Desacoplado:** La plataforma debe permitir que los componentes sean autónomos entre sí, de tal manera que puedan funcionar independientes, sin dejar de funcionar correctamente otros módulos de la plataforma.
- e. **Interoperable:** La plataforma debe utilizar estándares conocidos para permitir la operatividad con otros componentes, módulos o aplicaciones, que se deseen integrar.
- f. **Mantenible:** La plataforma debe proporcionar una gestión, operación y administración centralizada para todos los servicios desplegados sobre su infraestructura.
- g. **Robusto:** La plataforma debe ser lo suficientemente robusta para soportar todo el tráfico de contenido, la cantidad de usuarios conectados al mismo tiempo, y los procesos concurrentes, evitando caídas o no disponibilidad.

### 3.6 Alcances y restricciones del proyecto

Los requisitos funcionales y no funcionales de la plataforma están definidos, pero se han establecido algunos alcances y restricciones al proyecto, ya que en este trabajo de tesis no se ahonda en todos los puntos mencionados anteriormente, debido a que es un tema que abarca otras aplicaciones que se salen del objetivo de este trabajo.

- El desarrollo de este proyecto está enfocado en cumplir con los requisitos y necesidades únicamente del holding Komatsu.
- Solución será desarrollada solamente por el alumno tesista.
- La plataforma será construida utilizando librerías, aplicaciones y frameworks de código libre que se encuentren disponibles en internet.
- La solución se basa en el modelo de espiral del conocimiento de Nonaka y Takeuchi.
- La evaluación final será aplicada a un universo de entre 10 y 20 usuarios representativos.
- Se valida que la solución propuesta es viable, generando un modelo de contexto de la metodología CommonKADS: 1. OM: estudio de alcance y factibilidad.

- Las siguientes son algunas características y módulos que debe contener la solución en la etapa de marcha blanca:
  - Solución construida sobre una plataforma web
  - Motor de indexación y buscador semántico
  - Sistema de taxonomías (etiquetas apropiadas)
  - Sistema de notificación (pizarra o muro)
  - Sistema de estadísticas
  - Gestión de documentos electrónicos (documentos, fotos, videos, otros)
  - Mapas de conocimiento
  - Wikis, grupos (foros) y chat
  - Repositorio o integración con la nube
- El sistema debe ser construido de manera que pueda integrar a futuro los siguientes módulos:
  - Sistema de workflow
  - Herramienta de apoyo a la innovación: brainstorming, mapas conceptuales, mapas mentales
  - Integración con clientes de la compañía
  - Integración con redes sociales externas
  - Construcción de API para integrar nuevos módulos personalizados
  - Encuestas
  - Blogs
  - Herramienta de inteligencia competitiva
  - Integración con calendario y tareas
  - Integración con correo
  - Integración con un sistema de e-learning
- No se considera un módulo de gestión de contenido independiente, ya que el contenido se genera mediante las diversas aplicaciones que ofrece la misma plataforma.

### 3.7 Contribución del capítulo

En este capítulo se analizaron y especificaron los principales requisitos funcionales y no funcionales de la plataforma de gestión del conocimiento mediante la aplicación de la metodología CommonKADS que se utiliza para la analizar la implementación de este tipo de sistemas, de manera de obtener correctamente los requisitos esenciales. También se realizó una analogía general entre las redes sociales y los sistemas de gestión del conocimiento, con el fin de poder conocer y aprovechar las ventajas de las redes sociales y poder aplicar de mejor forma en este proyecto.

En el siguiente capítulo se detallará el proceso de construcción de la arquitectura de la plataforma de GC y los componentes necesarios para cumplir con los requisitos definidos en este capítulo.

## Diseño de la Plataforma

### 4.1 Características de la arquitectura del sistema

Para cumplir con los requisitos funcionales y no funcionales especificados en el capítulo anterior, se diseñó una solución basándose en la metodología propuesta. Esta solución cumple con estándares de código abierto e integra los componentes utilizados tanto en servidores físicos como en la nube.

### 4.2 Arquitectura de la plataforma

A continuación se explica detalladamente la arquitectura física de la plataforma, y la arquitectura contenida en cada una de ellas.

#### 4.2.1 Arquitectura física

La arquitectura física se compone de tres servidores: un servidor de aplicaciones, un servidor de bases de datos y un servidor repositorio de datos, como se grafica en la figura 4.1. El servidor de aplicaciones es el único que debe tener acceso directo a los dos restantes y debe ser accesible desde la red. Será el servidor al que los usuarios tendrán acceso. Todos los servidores pueden estar o no en la misma red.

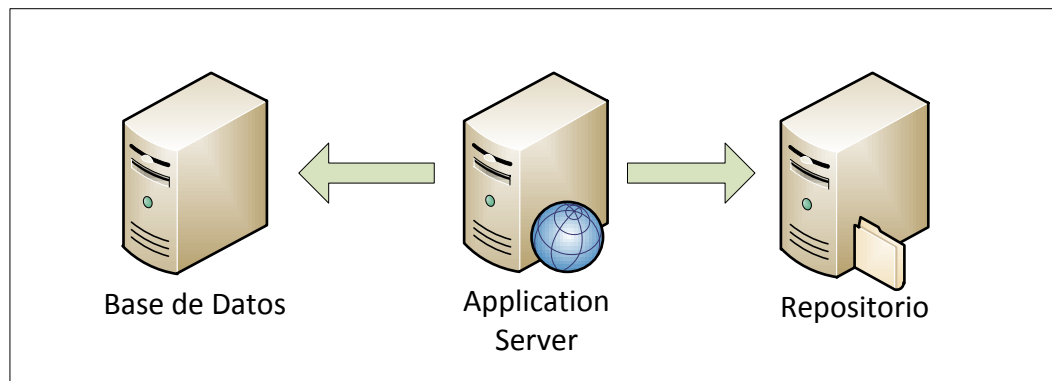


Figura 4.1: Arquitectura física de la plataforma.

### 4.2.2 Servidor de aplicaciones

Con el fin de cumplir con los requerimientos de sistema, el servidor de aplicaciones debe cumplir con tecnología de código abierto, el cual tendrá un sistema operativo Linux, un servicio de aplicaciones Apache y lenguaje de programación PHP5.

Se utilizará un framework basado en PHP: Fat-Free, el cual es un framework ligero que posee funcionalidades suficientes para cumplir con los requisitos funcionales, ofrece beneficios tales como: eficiencia, facilidad para la codificación, librerías adicionales y cumple con el modelo de 3 capas MVC.

La plataforma tendrá cuatro componentes principales: primero un núcleo, que será la base del sistema, utilizando tecnologías como PHP5 con la API de Fat-Free, HTML5, CSS3 y jQuery como las características más importantes; segundo, un sistema de gestión de colas de mensajes que procesará los mensajes de dos formas posibles: uno sin demora de tiempo y otro con demora de tiempo, los que serán enviados desde el núcleo y las aplicaciones; tercero, están las cuatro aplicaciones base para la gestión del conocimiento: Wiki, Grupos, Preguntas y Respuestas y Gestión Documental. El cuarto componente será un sistema de WebServices para estandarizar y comunicar todas las solicitudes entre el núcleo y las aplicaciones. En la figura 4.2 se explica gráficamente la estructura del servidor de aplicaciones, con los componentes numerados.

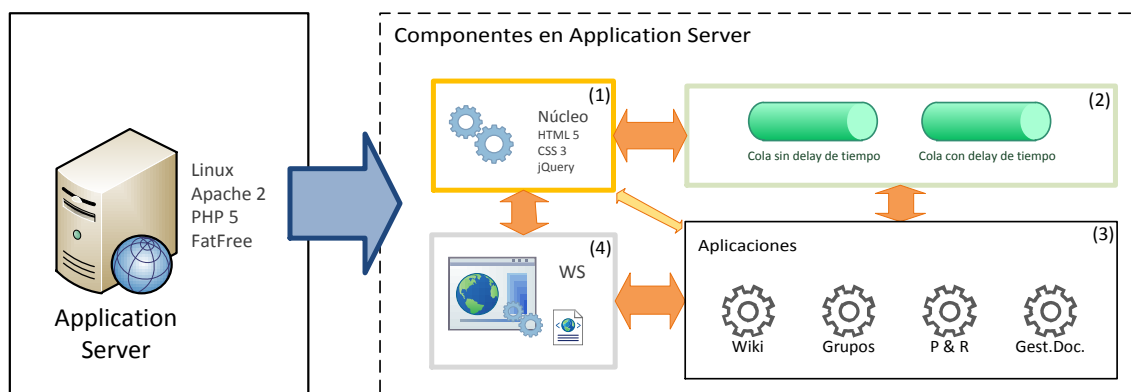


Figura 4.2: Componentes del servidor de aplicaciones.

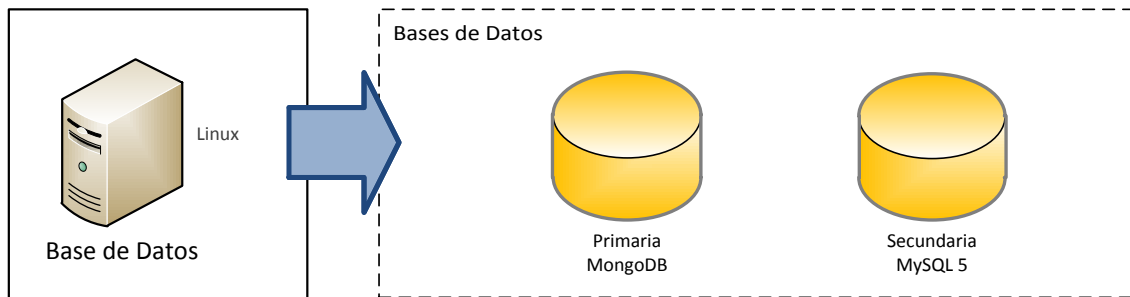
### 4.2.3 Servidor de bases de datos

El servidor de bases de datos albergará dos bases de datos: una principal que será MongoDB y una secundaria que será MySQL. Ambas bases de código abierto.

La principal, MongoDB, es una base de datos No-SQL, la cual está orientada a documentos, y trabaja bajo formato JSON, con un esquema dinámico, con el fin de que las aplicaciones sean más fáciles y rápidas. Esta base de datos tiene como objetivo

registrar los datos de la plataforma, tanto del núcleo, las aplicaciones principales de gestión del conocimiento, y todas las actividades y estadísticas.

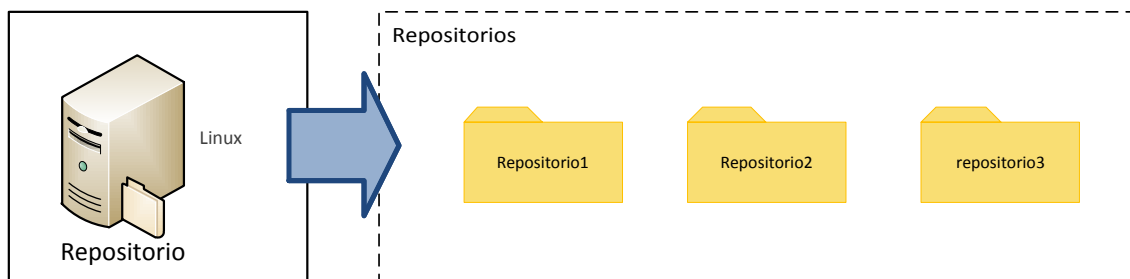
La segunda base de datos MySQL tiene como objetivo solamente albergar las aplicaciones adicionales que no sean compatibles con MongoDB y que se integren a la plataforma de gestión del conocimiento, de manera de compatibilizarlas. En la siguiente figura 4.3 se muestran estos dos servidores.



**Figura 4.3: Servidor de bases de datos.**

#### 4.2.4 Servidor de repositorios

El servidor de repositorios tendrá como objetivo administrar todos los repositorios requeridos en la plataforma de gestión del conocimiento, para contener los archivos que sean subidos e integrados desde el módulo de gestión documental, como se grafica en la figura 4.4.



**Figura 4.4: Servidor de repositorios.**

#### 4.2.5 Conexión con la nube

La plataforma tendrá la capacidad de conexión con la nube, por ejemplo, acceso a documentos remotos, mensajería instantánea, integración con correo electrónico y tareas o videos.



### 4.3 Componentes de la plataforma

La plataforma se diseñó utilizando diversos componentes para cumplir con los requerimientos de sistema, los que conforman la arquitectura principal del servidor de aplicaciones, como se muestra en la figura 4.5.

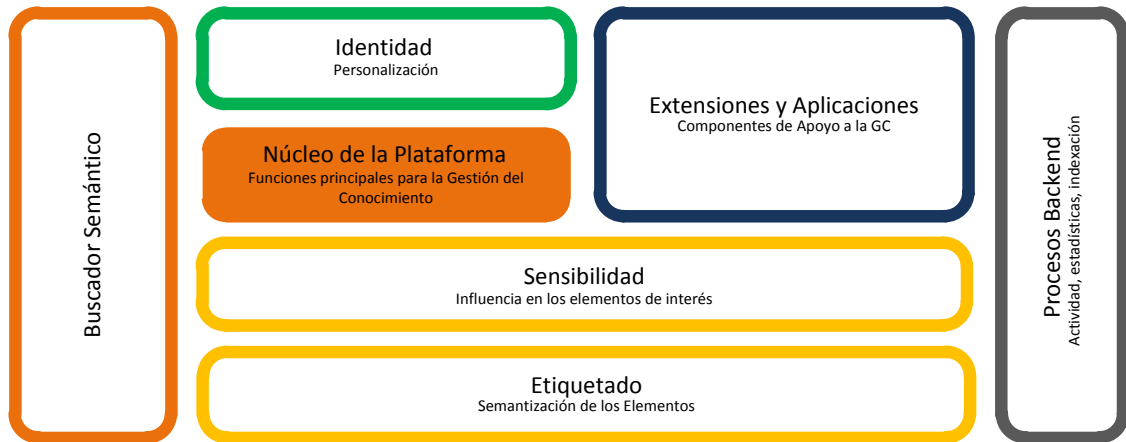


Figura 4.5: Componentes de la plataforma.

#### 4.3.1 Núcleo de la plataforma

El núcleo de la plataforma es el componente principal que se definió en la etapa de diseño, posee los módulos más importantes para la gestión del conocimiento, que se concluyen del resultado de la aplicación de la metodología aplicada para el proyecto. Las cuatro aplicaciones seleccionadas son: Grupos, Preguntas y Respuestas, Gestión Documental y Wiki. Estas aplicaciones comprenden la esencia de la gestión del conocimiento. La figura 4.6 muestra gráficamente los componentes principales.



**Figura 4.6: Componentes del núcleo de la plataforma.**

### 4.3.2 Componente de identidad

La componente de identidad toma un rol importante dentro de este proyecto, ya que es la personalización del usuario en una plataforma virtual, donde se hace la diferencia entre un usuario común y una persona, con el fin de que éste pueda personificarse de la mejor manera, como se muestra en la figura 4.7.

Dentro de las alternativas que tiene la persona, está la personalización del perfil y sus preferencias, también está la personalización de sus habilidades, sus gustos y sus áreas de trabajo.



**Figura 4.7: Componente de identidad.**

### 4.3.3 Componente de sensibilidad

La componente de sensibilidad se refiere a la componente que detecta la influencia de los elementos de interés que tiene la persona, de acuerdo a su forma de navegación y utilización de las diferentes aplicaciones de la plataforma, con el fin de tomar un registro de cada acción efectuada y guardarlo dentro de las preferencias de la persona.

Entre las sensibilidades diseñadas, están las visitas o aperturas de un contenido específico, el etiquetado de algún contenido, los comentarios ingresados, las búsquedas efectuadas, el contenido compartido, la categorización de los elementos, la aprobación o desaprobación de algún contenido y las recomendaciones realizadas. Estas sensibilidades se muestran en la figura 4.8.

Todas estas acciones quedan en un historial, con el que se podrá obtener un perfil de preferencias dinámico de cada persona, y ofrecer contenido recomendado según lo que indiquen estas preferencias.

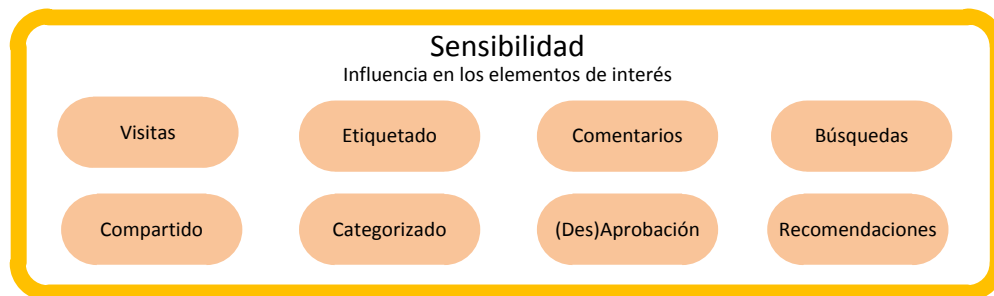


Figura 4.8: Componente de sensibilidad.

### 4.3.4 Componente de etiquetado

La plataforma se basará en un sistema de semantización básico, que se irá construyendo dinámicamente según el contenido aportado por las personas, como son los casos de etiquetado simple en cada contenido, y el de categorización de éstos.

Como se grafica en la figura 4.9, ambas alternativas se podrán utilizar sobre todos los contenidos de la plataforma, y específicamente en el caso de la categorización, se podrá generar un mapa de conocimiento creado colaborativamente por el aporte de las personas.

Esta categorización estará disponible en el buscador semántico de manera de poder filtrar según las categorías seleccionadas.



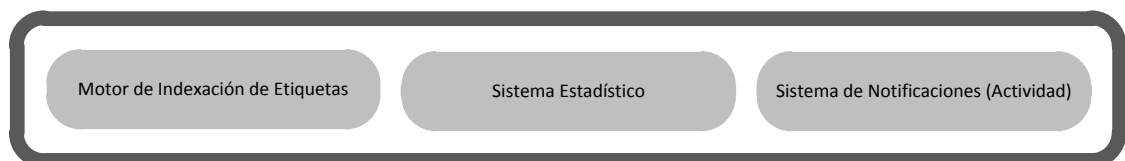
**Figura 4.9: Componente de etiquetado.**

### 4.3.5 Buscador semántico

El buscador semántico es un componente transversal de la plataforma, servirá para hacer búsquedas semánticas, utilizando las etiquetas y categorización definida por los usuarios, la cual arrojará resultado de todas las aplicaciones integradas en la plataforma.

### 4.3.6 Procesos de back-end

Los procesos de back-end son el segundo componente transversal de la plataforma. Son procesos que no son visibles por el usuario, pero que constantemente están realizando cambios según las acciones que ejecutan los usuarios. Como se detalla en la figura 4.10, el primer proceso es el motor de indexación de etiquetas, el que consiste en recolectar información semántica de las aplicaciones que se encuentran utilizando la base de datos secundaria, para efectuar cambios en la base de datos primaria. El segundo proceso es el sistema estadístico, que se encarga de trabajar con los datos semánticos para generar estadísticas por usuario. El tercer proceso es el de sistema de notificaciones o actividad, que permite recolectar e informar todas las actividades recientes de cada persona.



**Figura 4.10: Procesos de back-end.**

### 4.3.7 Extensiones y aplicaciones

Como apoyo a la gestión del conocimiento, existe una serie de aplicaciones adicionales a las cuatro aplicaciones principales del núcleo. Estas aplicaciones o extensiones sirven para dar valor agregado a la plataforma e incentivar el uso de ésta, haciéndola más atractiva al hacerla más similar a una red social.

Como extensión de una red social están las aplicaciones de repositorio de imágenes y videos, chat y geolocalización.

Como aplicaciones adicionales también están los blogs, encuestas, calendario y tareas, integración con correo electrónico, herramientas de brainstorming y mapas conceptuales, workflow e integración con la nube.



**Figura 4.11: Extensiones y aplicaciones.**

## 4.4 Gestor de colas de mensajes

Dentro de la plataforma se diseñó un sistema para la gestión de colas de mensajes, que cumple con la capacidad de procesar mensajes desde y hacia distintos componentes del sistema, a medida que se vaya requiriendo, esto con el fin de no saturar la plataforma con solicitudes excesivas en tiempo real.

Existen dos posibilidades, una de envíos de mensajes sin demora de tiempo y otra con demora de tiempo.

Para el caso del procesamiento de mensajes sin demora de tiempo, se consideran los procedimientos o acciones que deben ser procesados de inmediato pero a medida que sea posible, con el fin de no saturar los servidores, debido a la alta cantidad de eventos. Esta modalidad se aplica a los siguientes procesos:

- Proceso de Sensibilización
- Proceso de Actualización de Wiki
- Proceso de Actividad

En el caso del procesamiento de mensajes con demora de tiempo, éstos no requieren que se ejecuten de inmediato, ya que no es crítico que se mantenga actualizado el contenido en el instante. Esta modalidad también tiene como fin darle prioridad a los procesos sin demora de tiempo. Los procesos que se consideran para esta modalidad son:

- Proceso de Actualización Secundaria a Primaria
- Proceso de Recomendación
- Proceso de Generación de Estadísticas

En la figura 4.12 se muestran ambos tipos de procesamiento de mensajes.

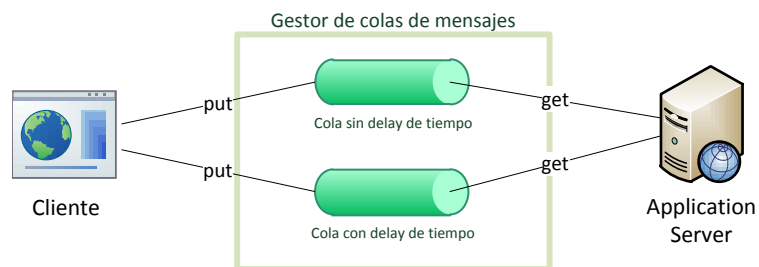


Figura 4.12: Gestor de colas de mensaje.

## 4.5 Componentes del front-end de la plataforma

Los componentes del front-end son aquellas funciones en las que el usuario puede acceder y operar a través de la plataforma. Para describir los componentes, se utiliza el estándar de Casos de Uso, descrita en el lenguaje UML<sup>1</sup>. A continuación se indican los componentes de front-end.

<sup>1</sup> Unified Modeling Language™ (UML®), <http://www.uml.org/>

### 4.5.1 Buscador semántico

Como se indicó en los puntos anteriores, el buscador semántico es un componente transversal que opera en toda la plataforma. El primer caso de uso es la búsqueda de texto libre, para desplegar un resultado. Sobre esa búsqueda se pueden seleccionar los resultados deseados, escogiendo el tipo de búsqueda, por ejemplo por personas, grupos, entradas, documentos, etc. Otro caso de uso es el de selección por tipo de filtro, donde se puede ir eliminando las etiquetas para desplegar un resultado más exacto o restableciendo los filtros para volver al resultado original.

En la figura 4.13 se muestra en detalle los casos de uso para el buscador semántico.

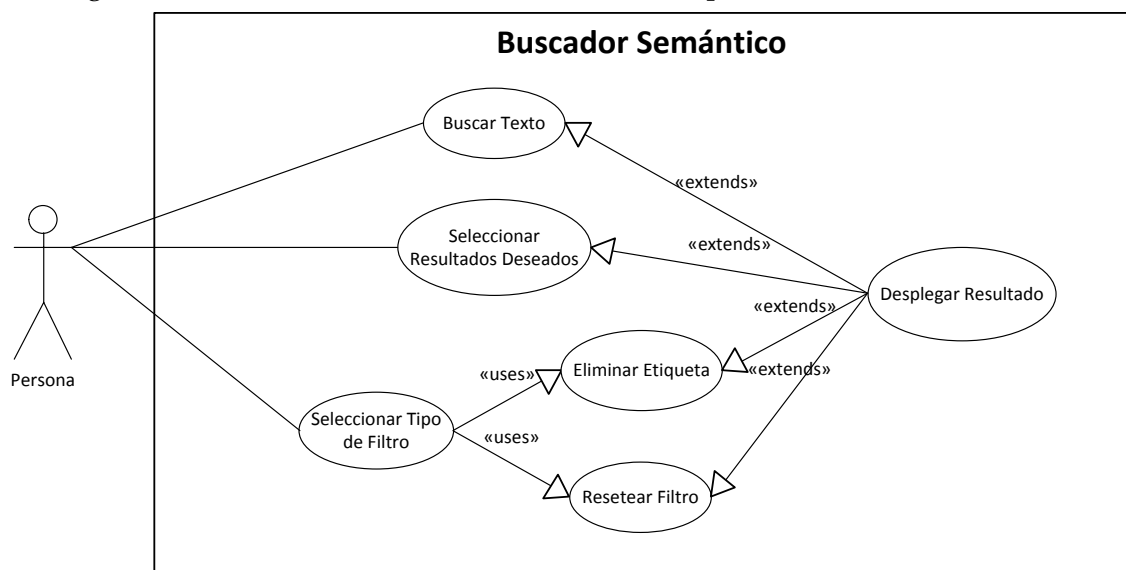


Figura 4.13: Caso de Uso para Buscador semántico.

En cuanto al wireframe (WF) o estructura visual de la interfaz web, se esquematizó de acuerdo a la figura 4.14: en la sección superior se encuentra el encabezado de la plataforma, luego abajo, un cuadro con el espacio para escribir el texto libre de búsqueda y a la derecha de éste la sección para escoger el tipo de resultado. A continuación abajo se encuentra el filtro de etiquetas y finalmente el espacio de resultados.

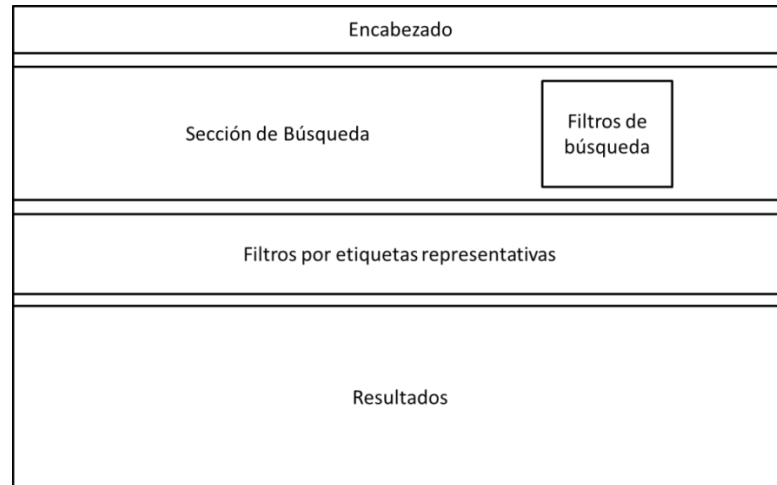


Figura 4.14: Wireframe del buscador semántico.

### 4.5.2 Identidad

En cuanto a los casos de uso del componente de identidad, la persona podrá visualizar su perfil, y además podrá modificar la información básica de su perfil, como fotos, presentación, datos de contacto y gustos, podrá modificar sus preferencias, su experiencia, habilidades técnicas y conocimientos, también sus áreas de trabajo, tal como se muestra en la figura 4.15.

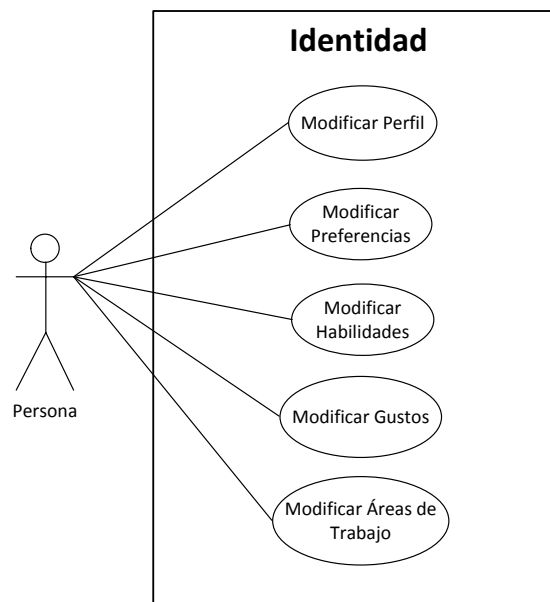
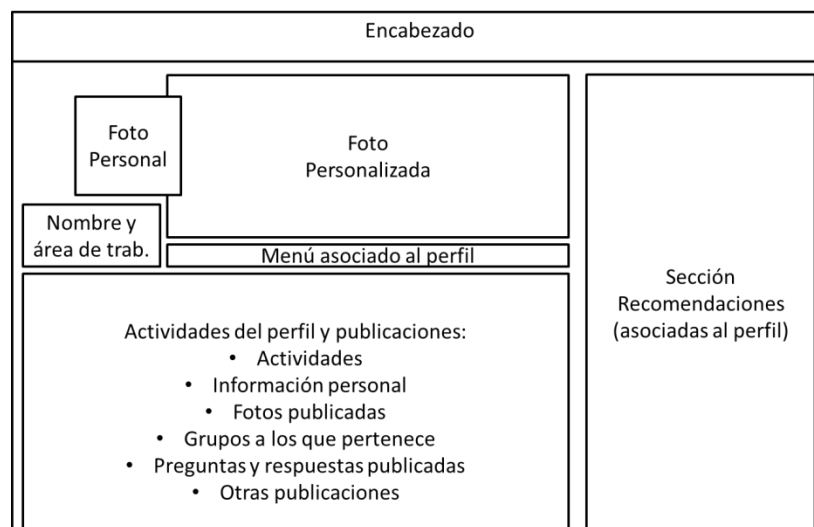


Figura 4.15: Caso de Uso para la Identidad.



El WF de la visualización del perfil del usuario, se estructuró de acuerdo a la figura 4.16: en la sección superior se encuentra el encabezado de la plataforma, luego abajo viene una foto de perfil junto con otra foto personalizada. Abajo de la foto de perfil se encuentra su nombre y el área de trabajo a la que pertenece. Debajo de la foto personalizada se encuentra el menú asociado al perfil, con diferentes accesos a información relacionada con el perfil, como información básica, contactos, galería de fotos, preguntas realizadas, y otras. Debajo de todo lo anterior se encuentra el contenido asociado al menú seleccionado. Al costado derecho se encontrará una sección informativa con las recomendaciones de contenido asociadas a las preferencias del usuario.



**Figura 4.16: Wireframe del perfil.**

### 4.5.3 Sensibilidad

La sensibilidad es un componente que detecta los intereses del usuario, de acuerdo a la navegación y acciones que realiza sobre los distintos contenidos. Estas acciones se traducen en diferentes casos de uso, como son el de visitar un contenido específico, puede ser etiquetado o categorizado. También el contenido puede ser compartido, recomendado, o se puede comentar sobre ello. Según las preferencias de la persona, esto también puede ser aprobado o desaprobado. Por último la sensibilidad también actúa sobre la búsqueda de contenido.

En todos estos casos de sensibilidad, las acciones implican que se realicen cambios sobre las preferencias de la persona y se actualicen en su historial.

En la figura 4.17 se grafican los casos de uso para el componente de sensibilidad.

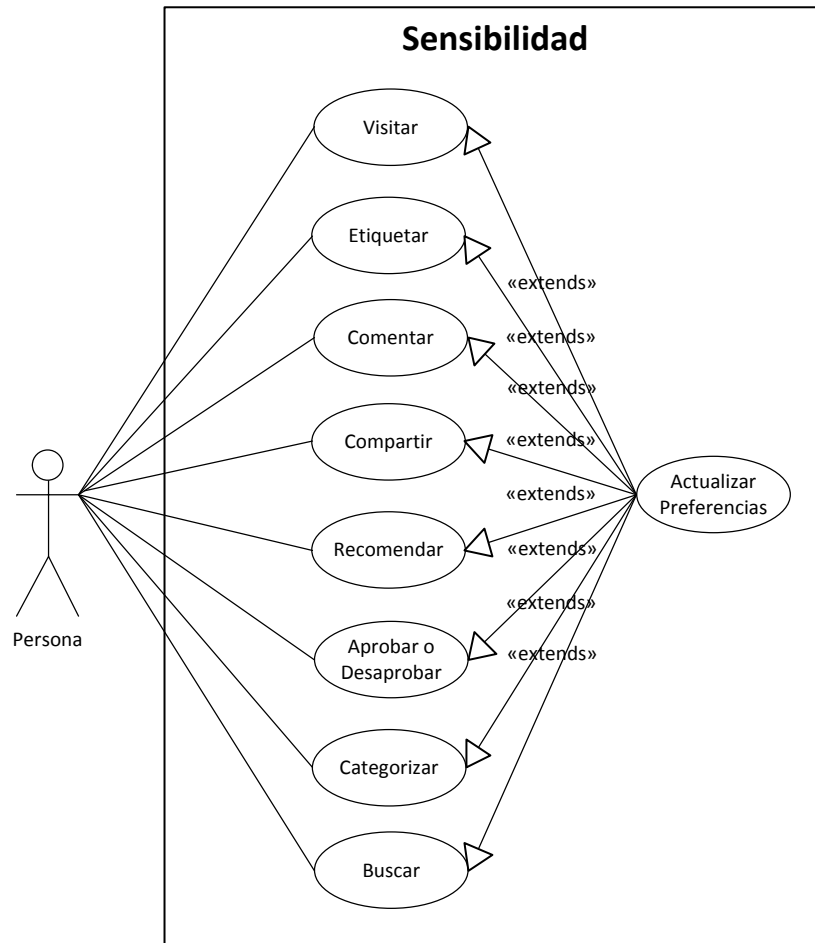


Figura 4.17: Caso de Uso para la Sensibilidad.

#### 4.5.4 Grupos

Los grupos son uno de los principales componentes para la gestión del conocimiento. Es una aplicación que consiste en la posibilidad de agrupar personas con el fin de discutir sobre algún tema específico. Dentro de los casos de uso del componente, está la creación de un grupo, desde donde se desprende la modificación de los datos del grupo y la posibilidad de cambiar los permisos del grupo, para que sea un grupo abierto o cerrado a la comunidad. Otro caso de uso es la invitación de personas al grupo y su respectiva asignación de roles dentro del grupo. Los casos de uso restantes son la creación de temas y la publicación dentro del tema creado, con la posibilidad de poder moderar, de quienes tienen permisos en éstos.

En la figura 4.18 se grafican los casos de uso para el componente de grupos.

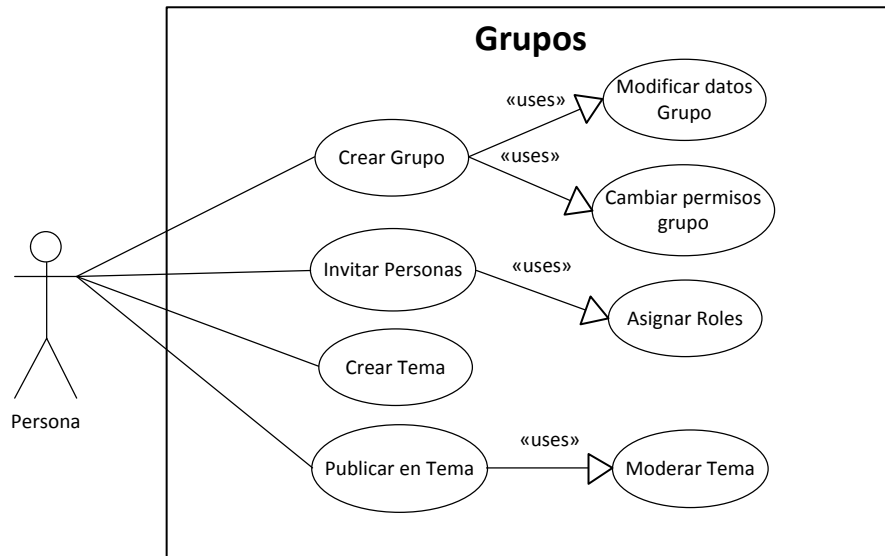


Figura 4.18: Caso de Uso para Grupos.

#### 4.5.5 Preguntas y respuestas

Las preguntas y respuestas son otro de los principales componentes para la gestión del conocimiento. Es una aplicación donde las personas tienen la posibilidad de hacer preguntas abiertas a la comunidad, con el fin de obtener la mejor respuesta y aclarar dudas, evaluando cada respuesta. Los casos de uso son la creación o publicación de una pregunta, con la posibilidad de modificarla y cambiar los permisos, las que pueden ser abiertas o cerradas. Otro caso de uso es publicar una respuesta a una pregunta dada. El último caso de uso es poder asignar un puntaje a cada respuesta. En la figura 4.19 se muestran los casos de uso para el componente de preguntas y respuestas.

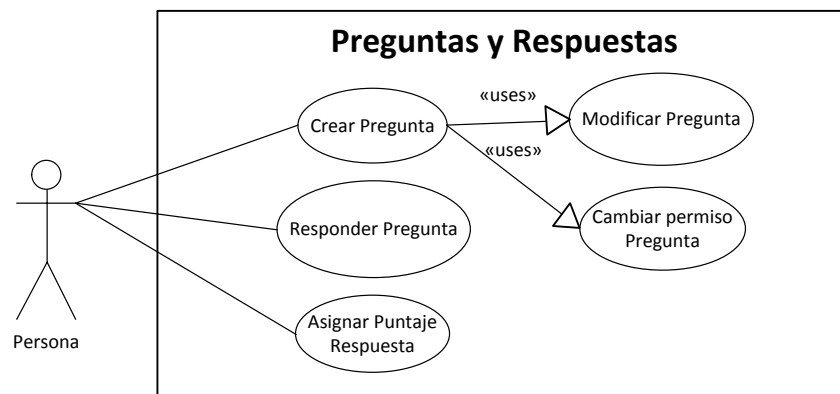


Figura 4.19: Caso de Uso para Preguntas y Respuestas.

### 4.5.6 Gestión documental

Otro de los componentes principales de la gestión del conocimiento es la gestión documental. Ésta es una aplicación para gestionar los archivos y todo tipo de documentos que sean importantes para considerar como parte del conocimiento por la comunidad. Los casos de uso son la publicación de un documento, con la posibilidad de actualizar el documento con una nueva versión, donde se pueden cambiar los permisos y/o se pueden asignar a un grupo específico. También está el caso de uso de pre-visualizar el documento, sin necesidad de tener que descargarlo y el último caso es el de descargar el archivo en el disco duro del usuario.

En la figura 4.20 se muestran los casos de uso para el componente de gestión documental.

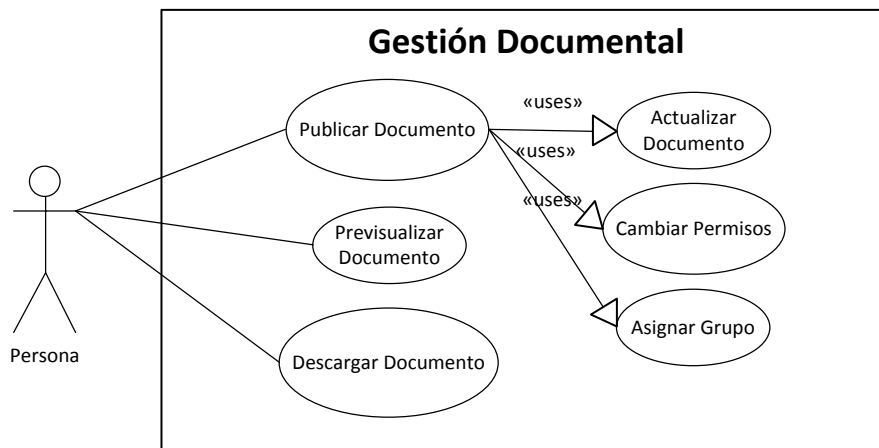


Figura 4.20: Caso de Uso para Gestión Documental.

### 4.5.7 Wiki

El último de los componentes principales como gestión del conocimiento es la aplicación de Wiki. Esta aplicación tiene como objetivo crear conocimiento, es una herramienta colaborativa, donde todos los miembros de la comunidad pueden hacer aportes o mejoras en definiciones ya existentes o pueden crear nuevas definiciones wiki. Los casos de uso para esta aplicación son: publicar una nueva wiki, editar la wiki creada, desde donde se desprende la posibilidad de cambiar la definición, se pueden asignar nuevos términos relacionados, cambiar las referencias y actualizar los enlaces externos. También está el caso de uso de marcar una wiki como una etiqueta, y poder ver el historial de cambios de cada wiki, donde se muestra el usuario y la información de la versión modificada.

En la figura 4.21 se detallan los casos de uso para el componente de wiki.

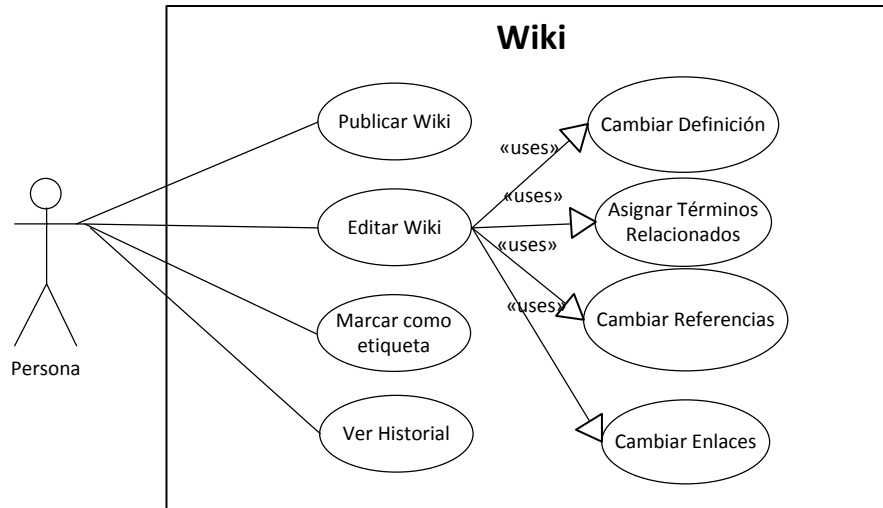
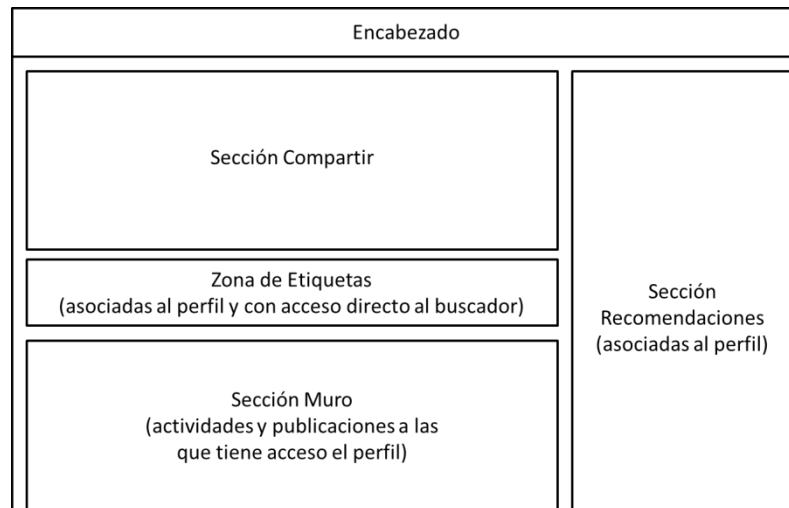


Figura 4.21: Caso de Uso para Wiki.

#### 4.5.8 Otros componentes

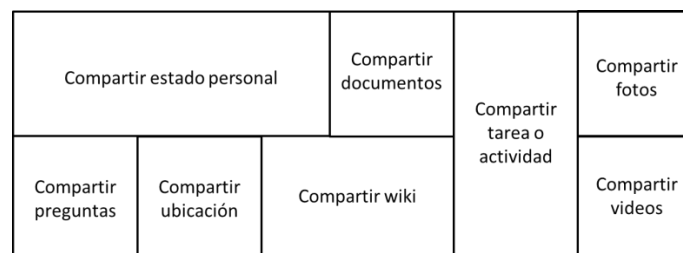
Además de los componentes principales para la gestión del conocimiento, existirán otras aplicaciones y secciones dentro de la plataforma. Todas estas secciones seguirán la misma estructura en la que se destaca la página principal o muro de la plataforma. El WF para esta sección se compondrá primero del encabezado en la parte superior, luego vendrá un cuadro con una sección para poder compartir diferentes tipos de datos o conocimiento. Más abajo habrá una zona de etiquetas, en las que aparecerán las etiquetas más importantes asociadas al perfil y enlazadas al buscador. Más abajo se ubicará la zona de actividades y publicaciones de toda la comunidad, las que irán ordenadas por las más recientes. Finalmente, al igual que en el perfil, al costado derecho se encontrará una sección informativa con las recomendaciones de contenido asociadas a las preferencias del usuario.

En la figura 4.22 se detalla el WF para la página principal.



**Figura 4.22: Wireframe de página principal.**

La sección compartir de la página principal, como se muestra en la figura 4.23, es una de las más importantes ya que es la que impulsa el compartir información o conocimiento, por eso debe ubicarse en primera plana de la página principal. Esta sección se divide en los elementos más importantes para compartir, como son: el estado particular de la persona, para incentivar la participación de las personas y emular un funcionamiento similar al de las redes sociales; compartir documentos para incentivar el crecimiento de conocimiento explícito; compartir preguntas, para aclarar dudas y aumentar la participación de la comunidad; compartir ubicación, es otra de las funcionalidades de redes sociales, que busca poder localizar a las personas cuando se encuentran en distintas localidades o en traslado; compartir wiki que busca incrementar el conocimiento a través de las definiciones de conceptos y las relaciones que tienen con otros; compartir tareas o actividades para incentivar la participación en comunidad; y finalmente compartir fotos y videos, para hacer participar a las personas de la misma forma que una red social, en actividades relacionadas con el trabajo.



**Figura 4.23: Wireframe de sección compartir.**

## 4.6 Procesos del back-end de la plataforma

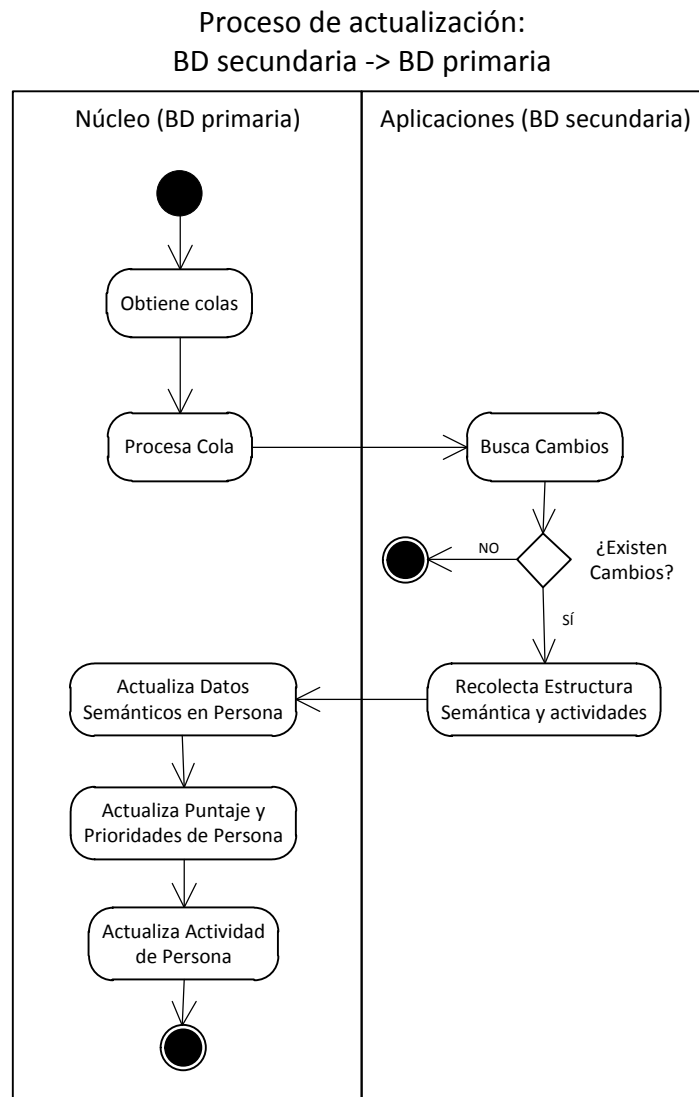
Los procesos del back-end son aquellos procedimientos que no son visibles para el usuario y funcionan por debajo del sistema. Para describir estos procesos, se utiliza el estándar de Diagrama de Actividades, descrita en el lenguaje UML. A continuación se indican los procesos del back-end.

### 4.6.1 Motor de indexación de etiquetas

El motor de indexación de etiquetas es el proceso que se encarga de actualizar la base de datos secundaria a la primaria. Este proceso se encarga de recolectar toda la información semántica y las actividades de las aplicaciones que se encuentran funcionando con la base de datos secundaria con el objetivo de ingresarlos en la base de datos primaria.

Este proceso trabaja con el gestor de colas de mensajes, en modalidad con demora de tiempo, ya que no está siempre trabajando. El procedimiento comienza cuando el gestor de colas activa el proceso, donde la primera tarea es buscar cambios en la base de datos secundaria, si no encuentran cambios termina el proceso, en cambio si encuentran cambios en los datos de las aplicaciones, recolecta toda la información relevante en cuanto a la estructura semántica y actividades relacionadas con el usuario. Una vez obtenida esta información realiza tres procesos: actualiza los datos semánticos de la persona, actualiza el puntaje y las prioridades de la persona, y finalmente actualiza las últimas actividades de la persona.

En la figura 4.24 se muestran los procesos involucrados para el motor de indexación de etiquetas.

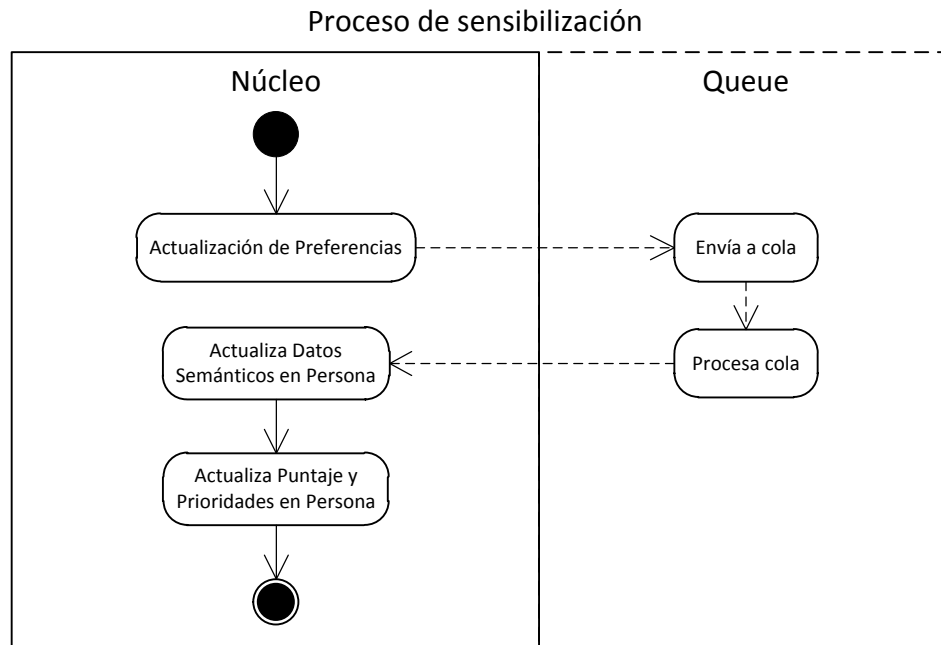


**Figura 4.24: Diagrama de proceso de actualización.**

#### 4.6.2 Proceso de sensibilización

El proceso de sensibilización es un proceso que trabaja bajo modalidad de gestor de colas sin demora de tiempo y, es el que se encarga de actualizar los datos semánticos de la persona y actualizar las prioridades de la persona cuando ésta actualiza sus preferencias desde el componente front-end de identidad, tal como se muestra en la figura 4.25.





**Figura 4.25: Diagrama de proceso de sensibilización.**

### 4.6.3 Proceso de actualización de wikis

El proceso de actualización de wikis es otro proceso que trabaja con el gestor de colas de mensajes en modalidad sin demora de tiempo. Se inicia cuando la persona crea una nueva etiqueta, luego el núcleo lo marca como nuevo contenido wiki posible, y es éste quien lo crea como nuevo contenido wiki. Finalmente la persona es quien edita el contenido wiki de la nueva entrada creada, desde el componente front-end de wiki. En la figura 4.26 se muestran los procesos relacionados con la actualización de wikis.

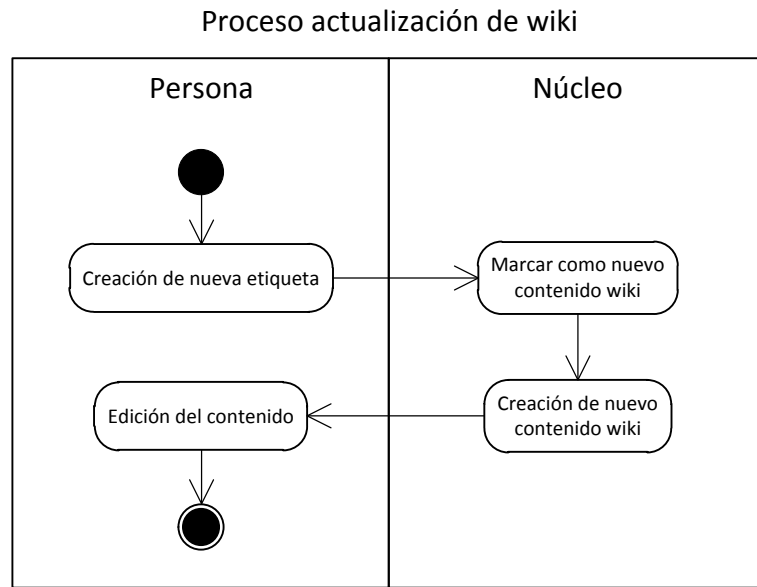


Figura 4.26: Diagrama de proceso de actualización de wiki.

#### 4.6.4 Proceso de recomendaciones

El proceso de recomendaciones es otro que trabaja bajo la modalidad con demora de tiempo en el gestor de colas de mensajes. Este proceso se activa cuando la persona visualiza la sección de recomendaciones. Desde ese punto es activada en el núcleo la obtención de prioridades que tiene la persona. Si no existen resultados, es decir, la persona no ha tenido actividades o es nueva, entonces obtiene los elementos según la prioridad global de toda la plataforma, en cambio si obtiene resultados, busca los elementos según las prioridades que tiene la persona. Luego de tener los resultados, éstos se preparan y finalmente se despliegan en la sección de recomendaciones. En la figura 4.27 se detallan los procedimientos del proceso de recomendación.

Proceso de recomendación

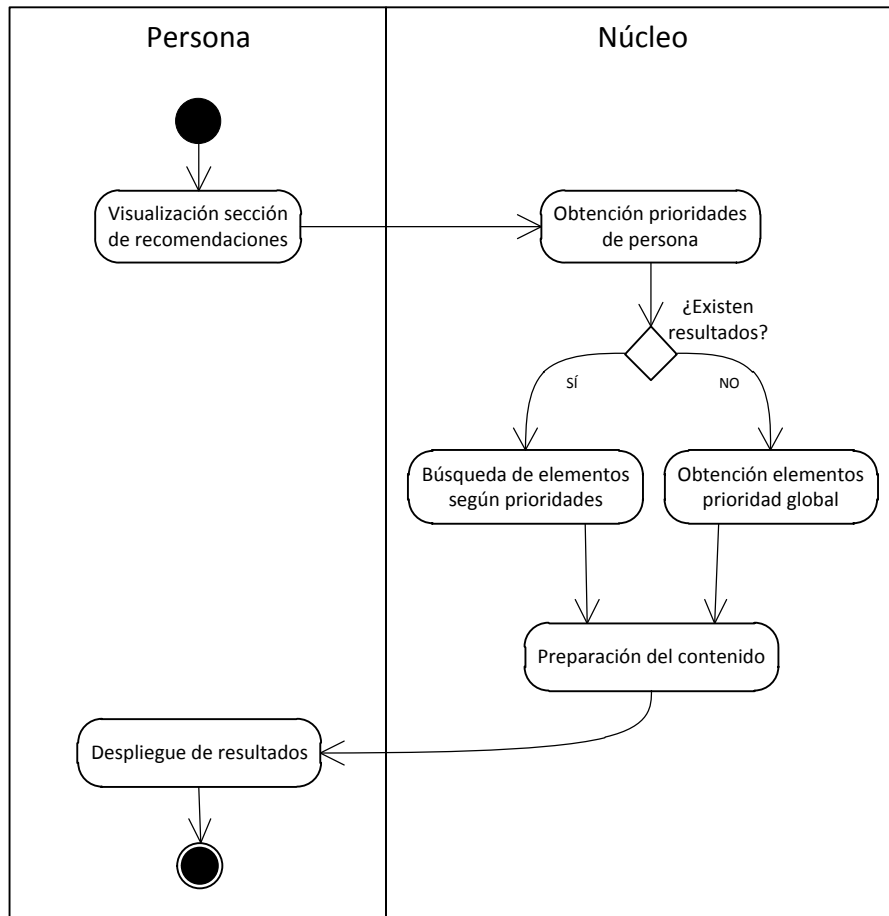
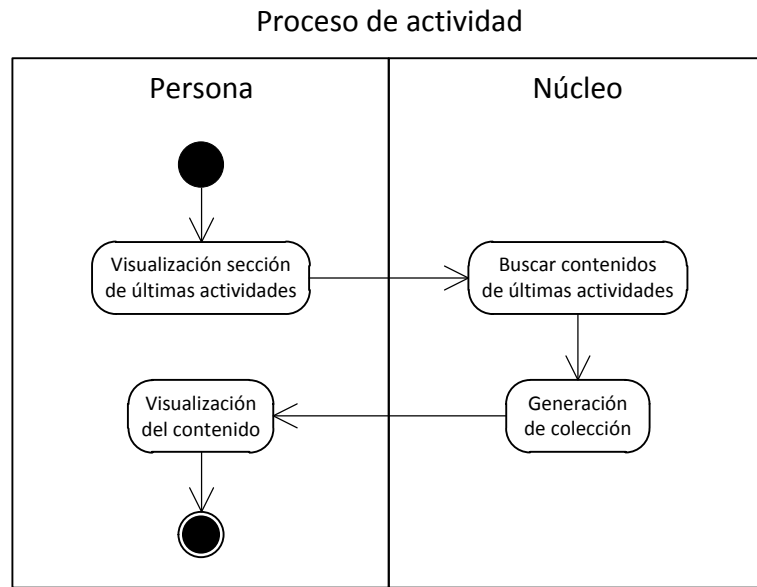


Figura 4.27: Diagrama de proceso de recomendación.

#### 4.6.5 Proceso de actividades

El proceso de actividades es un proceso que trabaja bajo la modalidad sin demora de tiempo a través del gestor de colas de mensajes. Este proceso consiste en la visualización de las últimas actividades de toda la comunidad o de una persona en particular. Se inicia cuando la persona entra a la página principal de la plataforma o cuando entra a un perfil específico de otra persona. Luego de visualizar, el núcleo busca los contenidos de las últimas actividades que estén relacionadas. Después de obtenerlas, genera una colección de actividades, que será el contenido que finalmente se despliegue para que la persona lo visualice. El proceso de actividad se detalla gráficamente en la figura 4.28.



**Figura 4.28: Diagrama de proceso de actividad.**

#### 4.6.6 Proceso de estadísticas

El proceso de estadísticas es el último proceso que trabaja bajo la modalidad con demora de tiempo del gestor de colas de mensajes. Este proceso se inicia cuando la persona realiza alguna acción, desde el componente de sensibilidad del front-end, donde a través del núcleo se ingresan los datos semánticos.

Por otro lado, cuando la persona visualiza la sección de estadísticas, el núcleo realiza una búsqueda de los datos estadísticos ya ingresados anteriormente. Finalmente se genera una colección con los datos registrados y se visualiza el contenido. Este proceso se muestra gráficamente en la figura 4.29.

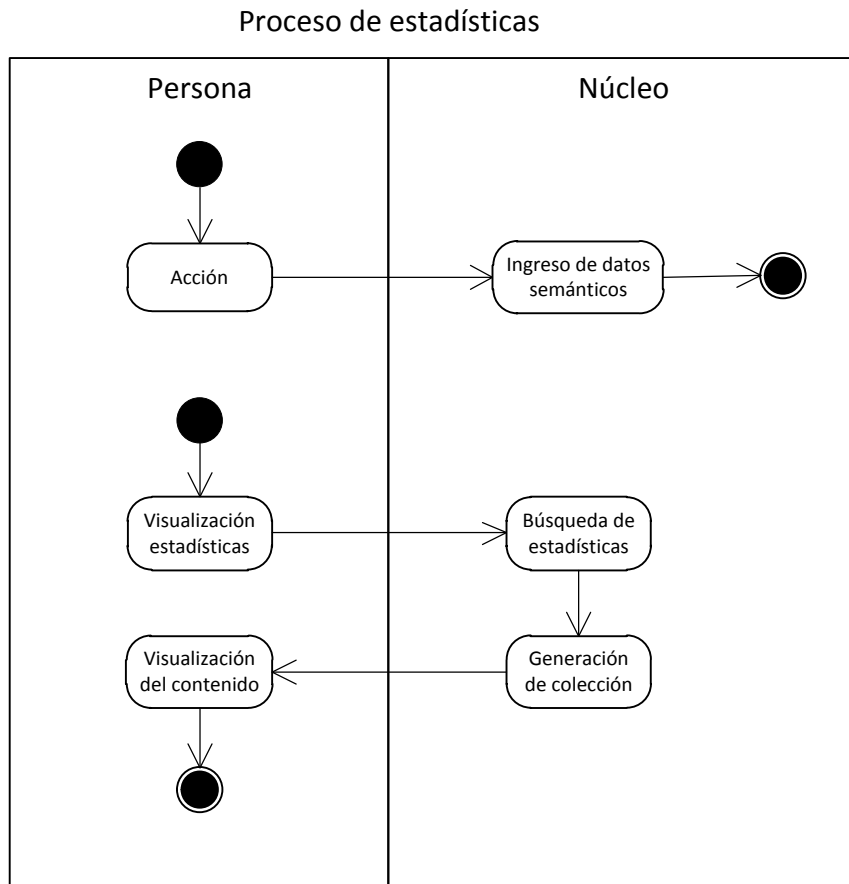


Figura 4.29: Diagrama de proceso de estadísticas.

## 4.7 Diseño gráfico

El diseño gráfico de esta plataforma es importante discutirlo, debido a que este proyecto está muy relacionado con la usabilidad de una red social. Es por esto que es necesario estudiar aspectos para que sea atractivo para los usuarios y sea similar a las redes sociales más usadas. También al ser una plataforma corporativa, debe mantener algunas reglas de la compañía en donde se está implementando este proyecto.



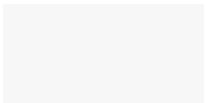
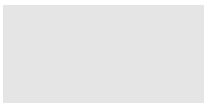

### 4.7.1 Logo y colores corporativos

Para diseñar gráficamente la plataforma, se estudia el logo corporativo de la compañía, que se muestra en la figura 4.30, y sus respectivas reglas en cuanto al uso de colores, para no salirse del contexto gráfico y mantener la línea corporativa.



**Figura 4.30: Logo corporativo.**

Los colores corporativos se desprenden del manual de uso del logotipo<sup>2</sup> de la compañía, desde el que se rescata el color principal, que es el azul, con la codificación CMYK 100% 90% 0% 0%. Como segundo color principal se rescata el amarillo con la codificación CMYK 0% 33% 100% 0%. Finalmente se escogen tres colores adicionales que cumplen con la exigencia de manual de uso, que son una variedad de grises, desde uno suave hasta uno oscuro, tal como se muestra en la siguiente tabla:

| Color       | CMYK                                | Código HTML | Muestra   |
|-------------|-------------------------------------|-------------|---|
| Azul        | C: 100%<br>M: 90%<br>Y: 0%<br>K: 0% | # 111083    |   |
| Amarillo    | C: 0%<br>M: 33%<br>Y: 100%<br>K: 0% | #FFAA00     |  |
| Gris suave  | C: 0%<br>M: 0%<br>Y: 0%<br>K: 3%    | #F7F7F7     |  |
| Gris medio  | C: 0%<br>M: 0%<br>Y: 0%<br>K: 10%   | #E5E5E5     |  |
| Gris oscuro | C: 0%<br>M: 0%<br>Y: 0%<br>K: 66%   | #575757     |  |

**Tabla 4.1: Tabla de colores utilizados en la plataforma.**

<sup>2</sup> Manual de uso del logotipo de Komatsu  
[http://www.komatsu.com/CompanyInfo/brand/logotype/download/komatsu\\_logotype\\_manual.pdf](http://www.komatsu.com/CompanyInfo/brand/logotype/download/komatsu_logotype_manual.pdf)

#### 4.7.1.1 Resultado logo de la plataforma

Como resultado del diseño gráfico de un logo para la plataforma, se obtuvo un logo basado del logo de la compañía, fusionado con la palabra comunidad en inglés “Community”, donde se mantiene “KOM” del logo de Komatsu en su color azul original y “munity” que viene de la palabra “Community” en el color amarillo corporativo. Adicionalmente se añade la palabra “My” dentro de un ícono de chat, para dar la idea que se está en “mi comunidad Komatsu”, haciendo la idea de una red social. Este resultado dio como nombre de la plataforma “MyKOMmunity”, el que se muestra gráficamente en la figura 4.31.



Figura 4.31: Logo propuesto para plataforma.

#### 4.7.2 Estructura encabezado

Para completar la idea de que se está en una red social, se estudiaron las redes sociales más usadas y conocidas, como son las de Facebook, LinkedIn, Twitter, Google, Outlook y Foursquare, donde se dedujo que todas siguen el mismo esquema: al extremo izquierdo va el logo pequeño, en el centro el buscador y a la derecha el perfil con los íconos de notificaciones y configuración. Es por eso que se diseñó un WF siguiendo esta misma estructura, como se muestra en la figura 4.32.

En la siguiente tabla se muestran los diferentes encabezados de las redes más populares:

| Encabezados de redes más populares |  |
|------------------------------------|--|
| Facebook:                          |  |
| LinkedIn:                          |  |
| Twitter:                           |  |



**Tabla 4.2: Tabla de encabezados de redes más populares.**

|           |          |      |
|-----------|----------|------|
| Logo-Home | Buscador | Menú |
|-----------|----------|------|

**Figura 4.32: Wireframe del encabezado.**

#### 4.7.2.1 Resultado encabezado de la plataforma

Como resultado del encabezado de la plataforma, se siguió el mismo patrón utilizado en las redes sociales más usadas, con el logo de “MyKOMmunity” a la izquierda, el buscador en el centro y los íconos de perfil y configuración a la derecha, utilizando los colores definidos anteriormente. En la figura 4.33 se muestra gráficamente el resultado obtenido.



**Figura 4.33: Diseño de encabezado propuesto para la plataforma.**

Los íconos utilizados en este diseño son: Muro con acceso directo a la página principal; E-Mail que indica las notificaciones de correos nuevos; Chat con acceso a la aplicación de chat y con las notificaciones de mensajes nuevos; Alerta con las notificaciones varias de todas las aplicaciones; Perfil, con acceso directo al perfil de la persona que está conectadas; Configuración, para acceder a la edición del perfil y configuraciones varias; Aplicaciones, para acceder al menú de todas las aplicaciones disponibles. En la figura 4.34, se explican cada uno de los íconos del encabezado.





**Figura 4.34: Explicación de diseño de encabezado propuesto.**

### 4.7.3 Estructura del sitio web

La estructura del sitio web seguirá el diseño del encabezado, los colores corporativos y siguiendo los WF diseñados en los puntos anteriores.

Es en el caso del WF de la página principal y de la sección compartir, el diseño gráfico que se realizó, donde se destaca que las opciones para compartir, intercambian los colores azul y amarillo, con un fondo gris suave. La sección de etiquetas sigue con un fondo gris medio y el texto con el gris oscuro.

Las cajas de actividades poseen un fondo gris suave, donde los textos son negros, se muestra la foto de perfil de la persona que efectuó esa actividad, con el nombre y el tiempo que ha pasado a un costado. En la parte inferior de la actividad se muestran las acciones que pueden efectuarse, como son las de hacer un comentario, poder aprobar o desaprobar la publicación, poder etiquetar la publicación o poder compartirla con más personas. Estas últimas tres acciones siguen en color amarillo y azul con un fondo gris medio.

En la columna derecha aparece el cuadro de recomendaciones, que sigue la misma estructura de las actividades de la parte central, pero en forma de columna, pero con la diferencia que no existen las opciones de poder comentar, aprobar, etiquetar o compartir, sólo se puede acceder a la publicación directamente.

En la figura 4.35 se muestra el resultado gráfico y diseño propuesto para la plataforma. Este mismo diseño gráfico y las mismas secuencias deben seguir todas las secciones de la plataforma, tanto para los resultados de búsquedas, perfiles o aplicaciones que estén disponibles dentro de la plataforma.

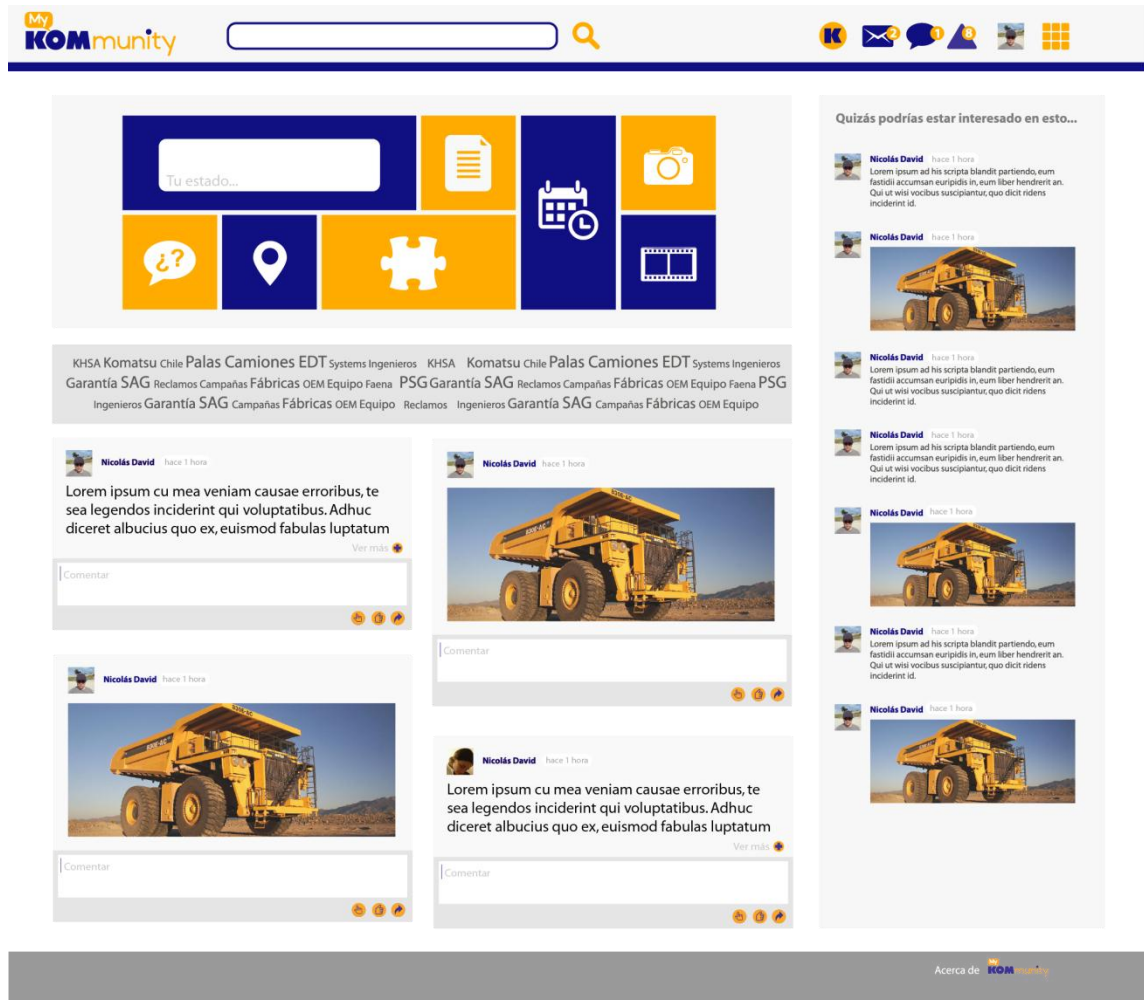


Figura 4.35: Diseño final propuesto para la plataforma.

## 4.8 Contribución del capítulo

En este capítulo se presentaron los principales patrones de diseño de componentes y procesos involucrados en la concepción de la plataforma, cumpliendo con los requisitos definidos en el capítulo anterior. Se definió una arquitectura de tres servidores, de aplicaciones, base de datos y repositorios. Además se definió la utilización de una base de datos primaria no estándar, llamada NoSQL, que contribuye en que la plataforma sea más rápida en el manejo de datos y facilita su construcción.

Adicionalmente se presenta la propuesta de diseño gráfico para la plataforma, que sigue las mismas tendencias de redes sociales actuales.

## Implementación de la Plataforma

### 5.1 Plan de proyecto

Para dar cumplimiento a los objetivos del proyecto, se implementó la solución que mejor se ajustó, según el resultado del análisis realizado, utilizando las características definidas en el capítulo anterior y resolviendo los requisitos funcionales y no funcionales identificados. El proceso de despliegue del sistema y las tareas necesarias para hacerlo se distribuyeron en las siguientes fases:

- a. Selección de una metodología de desarrollo de software.
- b. Revisión de los procesos de negocio.
- c. Revisión bibliográfica y del estado del arte de la documentación de sistemas e interfaces existentes y de las tecnologías requeridas.
- d. Especificación y análisis de los requisitos de la solución.
- e. Diseño de la solución que permita alcanzar el objetivo general.
- f. Desarrollo e implementación de la solución.
- g. Ejecución de pruebas integrales del sistema.
- h. Puesta en producción del sistema.

Se decidió separar el proyecto de implementación de la plataforma de gestión del conocimiento en las fases enunciadas anteriormente con los propósitos de:

- Determinar en cada fase los requisitos a cumplir.
- Determinar en cada fase los artefactos que serán traspasados a la siguiente fase.
- Retroalimentar la fase anterior con los resultados de la fase siguiente, de modo de realizar las correcciones necesarias para asegurar la validez, consistencia y coherencia del entregable en la fase siguiente.
- Verificar el cumplimiento de los criterios de aceptación y la determinación de cómo continuar el proceso en su fase siguiente.
- Establecer los recursos requeridos en cada fase, tanto de hardware, software, infraestructura y de recursos humanos.

- Establecer los tiempos en que se desarrollará el proceso de implementación y determinar los especialistas requeridos en cada fase.

En este capítulo se explican punto a punto cada una de estas fases.

## 5.2 Metodología de desarrollo de software

Con el fin de cumplir exitosamente el desarrollo de esta plataforma, se buscó una metodología de software que fuera apropiada por las características de este proyecto. Debido al poco tiempo disponible que se tenía para desarrollarlo y porque se necesitaba ir avanzando por etapas, se escogió una metodología incremental de desarrollo para trabajar, la que da la posibilidad no tener que esperar hasta una versión final completa para poder lanzar versiones de la plataforma, con la que se priorizan las funcionalidades más importantes a un inicio, y al final las restantes, dejando operativo en cada etapa la plataforma. Dentro de los beneficios de esta metodología está en que los primeros incrementos sirven como prototipos que ayudan en la tarea de detectar posteriores requisitos, también sirve para minimizar los riesgos de fallar con el proyecto final. Debido a que éste es un proyecto que se trabajó directamente con las personas que lo utilizarían, no había problemas con la utilización de esta metodología. En la figura 5.1 se muestra gráficamente la aplicación de la metodología de desarrollo de software utilizada.

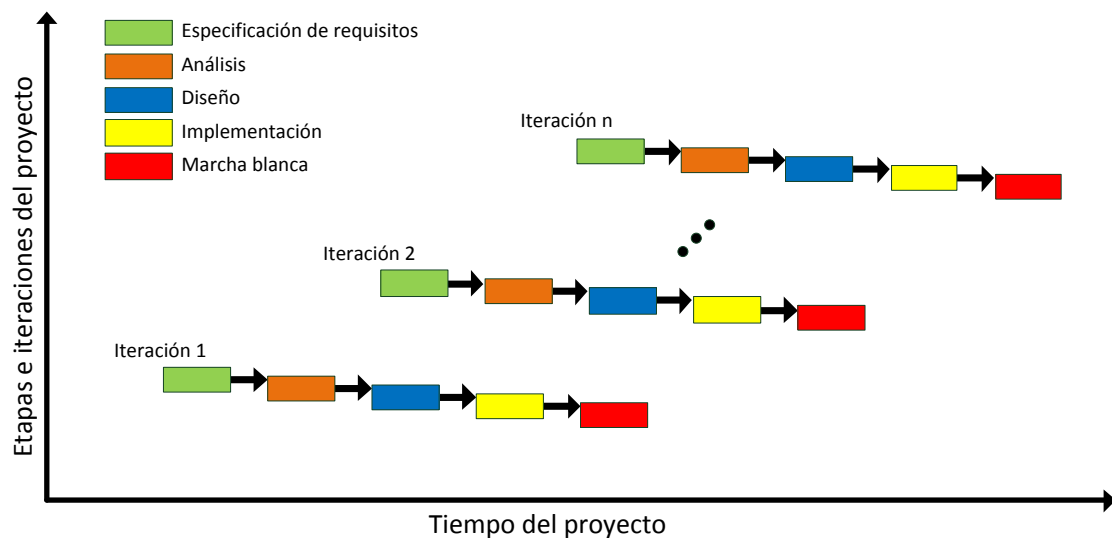


Figura 5.1: Metodología de desarrollo de software incremental.

### 5.3 Revisión de procesos de negocio

En esta fase, se revisaron los procesos de negocio y como resultado de este análisis se identificaron y especificaron las necesidades corporativas que tiene la compañía, tales como: transformar el conocimiento de las personas en competencias corporativas y ventajas competitivas, centralizar la información técnica esencial, capturar, organizar y almacenar el conocimiento de los trabajadores, etc. que demuestran y justifican la construcción de una de una plataforma especializada que las resuelva. Desde un punto de vista técnico los entregables de esta fase fueron los siguientes:

- Descripción a un alto nivel de los requisitos de la plataforma utilizando los requerimientos de negocio detectados.
- Análisis de brecha de software e identificación de los componentes que puedan ser necesarios para cubrir los requerimientos de negocio.
- Diseño conceptual de la solución de la plataforma.

### 5.4 Revisión de alternativas tecnológicas

En esta fase se realizó una investigación y prospección de estándares y de alternativas tecnológicas de código abierto disponibles en el mercado que sean los más apropiados para solucionar las necesidades de negocio detectadas en la fase anterior y que permitan desarrollar una plataforma de software completa. Esta investigación incluyó: revisiones bibliográficas, búsquedas en plataformas de gestión de proyectos open source, revisión de códigos fuentes, testeo de ejemplos y compatibilidad con otros componentes. El principal entregable de esta etapa fue la definición, a través de un proceso de evaluación, de un listado de frameworks, APIs, librerías y tecnologías disponibles como potenciales proveedores de la solución técnica demandada.

### 5.5 Especificación de requisitos

Con posterioridad al proceso de exploración tecnológica, se evaluó técnicamente de manera objetiva, cuidadosa y eficaz cada una de las alternativas seleccionadas, con el fin de verificar que cumplan con las especificaciones de requisitos y capacidad de soportar la construcción de la plataforma de gestión del conocimiento. Una vez finalizada esta evaluación, se seleccionaron los elementos que mejor cumplieron con los requisitos técnicos especificados.

## 5.6 Diseño de la plataforma

En esta etapa, se desarrolló un plan de acción para determinar los componentes requeridos para la construcción de la plataforma de GC y cumplir con los objetivos del proyecto. Los objetivos de esta fase fueron los siguientes:

- Diseño detallado de los procesos de negocio y componentes.
- Diseño detallado de la arquitectura funcional y técnica de la solución necesaria para soportar los procesos de negocio junto con su dimensionamiento correspondiente.

Para alcanzar los objetivos de esta fase se realizaron las actividades definidas a continuación, basadas en el diseño realizado en el capítulo 4 anterior.

### 5.6.1 Diseño arquitectónico de los servidores

En esta tarea se realizó el análisis y diseño detallado de la arquitectura física del sistema según fue explicado en el Capítulo 4. Adicionalmente, se confeccionaron y validaron las especificaciones de los servidores de bases de datos, de aplicaciones y de repositorios.

#### 5.6.1.1 Especificación servidor de bases de datos

**a. Servidor de base de datos primario:**

- **Motor de base de datos:** MongoDB versión 2.0.4.
- **Estructura de lenguaje:** JSON.
- **Cliente:** phpMoAdmin (PHP) versión 1.0.
- **Esquema de base de datos:** km
- **Espacio requerido en disco:** 218 MB.

**b. Servidor de base de datos secundario:**

- **Motor de base de datos:** MySQL versión 5.5.
- **Estructura de lenguaje:** SQL.
- **Cliente:** phpMyAdmin (PHP) versión 3.4.
- **Esquema de base de datos:** km
- **Espacio requerido en disco:** 200 MB.

#### 5.6.1.2 Especificación servidor de aplicaciones

- **Sistema operativo:** Linux Ubuntu versión 12.
- **Servidor de aplicaciones:** Apache versión 2.2.
- **Lenguaje base soportado:** PHP versión 5.3.

- **Framework:** Fat-Free PHP Framework versión 3.2.2.
- **Espacio requerido en disco:** 2 GB.

#### 5.6.1.3 Especificación servidor de repositorios

- **Sistema operativo:** Linux Ubuntu versión 12.
- **Servidor de aplicaciones:** Apache versión 2.2.
- **Lenguaje base soportado:** PHP
- **Espacio requerido en disco:** 20 GB.

### 5.6.2 Diseño base de datos

La base de datos primaria posee un motor MongoDB, el cual es un tipo de base de datos documental de estructura de datos NoSQL, por lo que el diseño de la base de datos se compone de colecciones que se guardan en formato BSON.

En el caso de la base de datos secundaria posee un motor de base de datos MySQL, el que sigue una estructura clásica de datos de SQL. Esta base de datos el libre de uso para las distintas aplicaciones que integradas a la plataforma no soporten una base de datos NoSQL como la primaria, es por eso que en esta etapa no se hará una especificación de los objetos para la base secundaria.

Para la base de datos primaria se definió un esquema llamado “km”, que representa a la gestión del conocimiento por sus siglas en inglés KM.

#### 5.6.2.1 Especificación base de datos primaria

##### a. Colecciones del esquema de base de datos “km”:

- **activities:** Registros de actividades que se realizan en la plataforma, indicando el tipo de actividad, la hora, el usuario que lo efectúa y si existe un destinatario, finalmente la actividad.
- **entries:** En esta colección se guardan los registros de tipo entrada, con son el caso de las wikis o de muro, donde se detalla el usuario, la fecha, tipo de entrada, texto de entrada, y un posible destinatario.
- **galleries:** Aquí se registran las fotos, imágenes y galerías de imágenes por cada usuario, registrando el archivo, fecha y tipo.
- **groups:** En esta colección se guardan todas las personas que están relacionadas con otras, tanto como contactos o como grupos de personas, que pueden ser privados o públicos, se registran los miembros, nombre del grupo, tipo de grupo y persona a la que pertenecen.
- **logs:** Aquí se registra todo tipo de respaldo como información del sistema, indicando el tipo de operación, usuario y fecha.

- **messages:** Se registran los mensajes internos entre las personas, indicando el emisor y el receptor, el mensaje y la fecha.
- **sensibilities:** En esta colección se indican todo el tipo de sensibilidades que ocurren en la plataforma, las que pueden ser acciones sobre el botón me gusta, compartir, etiquetados sobre elementos o comentarios, registrando el elemento, la persona, el tipo de operación, y toda información respectiva.
- **users:** Se registran los usuarios, con toda la información correspondiente a su perfil, sus contactos, última conexión, contraseña y opción para recuperar su contraseña.

#### 5.6.2.2 Especificación base de datos secundaria

##### a. Tablas del esquema de base de datos “km”:

- Ninguna requerida en la fase inicial de la plataforma.

### 5.6.3 Diseño MVC

Debido a que los patrones son una parte importante en el desarrollo de aplicaciones, ya que proporcionan soluciones probadas para sistemas comunes, se utilizó un patrón de diseño MVC o modelo-vista-controlador, tal como se muestra en la figura 5.2, soportado por el framework seleccionado Fat-Free. En este caso, ofrece una metodología en el desarrollo de la plataforma, que interactúa con el usuario para la manipulación de la información.

El objetivo de este patrón es separar la capa lógica de negocios de la interfaz gráfica, de manera que cambios en la misma, no afecten la lógica de negocios. El modelo representa a la información que tanto el usuario como la aplicación pueden manejar. La vista representa a todos los elementos que componen la UI, como son los cuadros de texto, botones, grillas, páginas, etc. En cambio el controlador, maneja la interacción y comunicación entre el modelo y las acciones que vienen desde la capa de vista, accionadas por el usuario.

Este patrón de diseño, se ha extendido de mejor forma a las aplicaciones web, de manera que el usuario navega hacia la URL de la plataforma, el controlador recibe la acción ejecutada por el usuario, y éste se comunica con el modelo para indicar la acción a tomar. El modelo regresa los datos requeridos y el controlador pasa estos datos a la vista. La vista genera el código HTML que se despliegan a través del navegador web, donde el usuario está actuando.



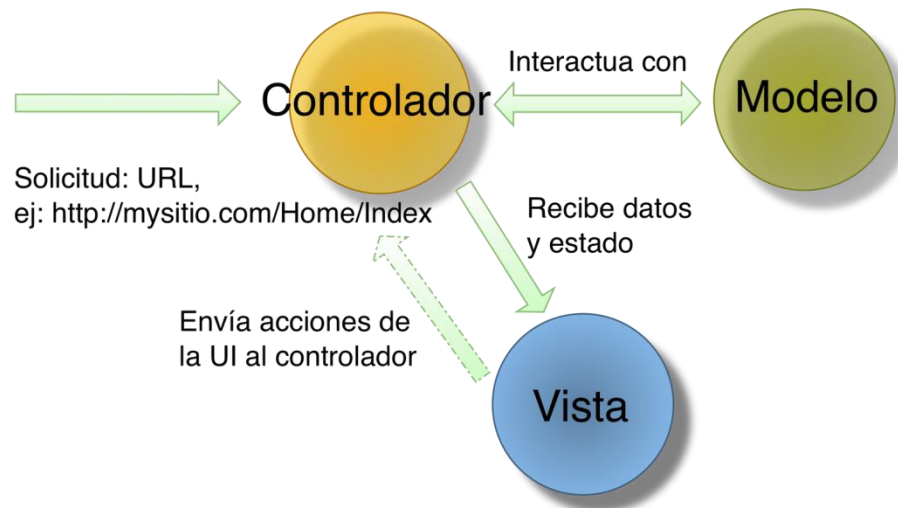


Figura 5.2: Patrón de diseño modelo-vista-controlador.

A continuación se indican las especificaciones de componentes y procesos que se basan en el diseño de arquitectura MVC del framework Fat-Free de PHP.

#### 5.6.3.1 Especificación componente de identidad

- **profile\_edit:** Función ubicada en la capa controlador que permite modificar el perfil, con opción de enviar los siguientes parámetros:
  - **preferences:** Datos asociados a las preferencias de la persona.
  - **skills:** Datos asociados a las habilidades de la persona.
  - **likes:** Datos asociados a los gustos de la persona.
  - **workarea:** Datos asociados a las áreas de trabajo de la persona.

#### 5.6.3.2 Especificación componente de sensibilidad

- **sensibility\_save:** Función ubicada en la capa controlador que permite guardar una sensibilidad, con opción de enviar los siguientes parámetros:
  - **visit:** Datos asociados a la visualización del elemento por la persona.
  - **comment:** Datos asociados a los comentarios que efectúa la persona.
  - **like:** Datos asociados a aprobación o desaprobación que realiza la persona.
  - **tag:** Datos asociados a la etiqueta que genera la persona en algún elemento.
  - **search:** Datos asociados a una búsqueda realizada por la persona.
  - **categorize:** Datos asociados a una categorización realizada por la persona.

- **share:** Función ubicada en la capa controlador que permite guardar la sensibilidad de compartir.

#### 5.6.3.3 Especificación componente de etiquetado

- **tag\_new:** Función ubicada en la capa controlador que permite crear una nueva etiqueta sobre algún elemento.
- **categorize\_new:** Función ubicada en la capa controlador que permite realizar una categorización sobre algún elemento.

#### 5.6.3.4 Especificación componente de búsqueda

- **search:** Función ubicada en la capa controlador que permite realizar una búsqueda de diferentes elementos, que permite los siguientes parámetros:
  - **query:** Datos asociados a la cadena de búsqueda realizada por la persona.
  - **resultType:** Selección que representa el tipo de resultado deseado.
  - **filterType:** Selección que representa el tipo de filtro deseado.
  - **removeTags:** Opción que permite eliminar ciertas etiquetas con el fin de mostrar un resultado más acertado.

#### 5.6.3.5 Especificación componente de grupos

- **group\_show:** Función ubicada en la capa controlador que permite visualizar el contenido de un listado de grupos o un grupo en específico.
- **group\_new:** Función ubicada en la capa controlador que permite la creación de un nuevo grupo.
- **group\_edit:** Función ubicada en la capa controlador que permite la modificación de un grupo existente, con los siguientes parámetros:
  - **data:** Datos que representan la información general del grupo.
  - **permissions:** Opciones que permiten cambiar los permisos del grupo.
  - **invite:** Opción que permite invitar a personas a participar del grupo.
  - **roles:** Opción que permite cambiar los roles de las personas que pertenecen al grupo.
- **group\_publish:** Función que permite publicar un tema en un grupo específico.
- **group\_moderate:** Función que permite moderar un tema específico de un grupo existente.

#### 5.6.3.6 Especificación componente de preguntas y respuestas

- **qa\_save:** Función ubicada en la capa controlador que permite publicar una nueva pregunta o respuesta, donde se permiten los siguientes parámetros:
  - **type:** Opción que permite indicar si es una pregunta o una respuesta.
- **qa\_edit:** Función ubicada en la capa controlador que permite modificar una pregunta o respuesta existente, donde se permiten los siguientes parámetros:
  - **permission:** Opción que permite cambiar los permisos de una pregunta, para indicar si es pública o privada.
  - **score:** Opción que permite indicar el puntaje o nota a una respuesta.

#### 5.6.3.7 Especificación componente de gestión documental

- **document\_save:** Función ubicada en la capa controlador que permite publicar un nuevo documento.
- **document\_edit:** Función ubicada en la capa controlador que permite modificar un documento existente, donde se permiten los siguientes parámetros:
  - **permission:** Opción que permite cambiar los permisos de un documento, para indicar si es público, privado o si pertenece a algún grupo específico.
- **document\_get:** Función ubicada en la capa controlador que permite extraer un documento existente, donde se permiten los siguientes parámetros:
  - **type:** Opción que permite indicar si el documento se quiere descargar o visualizar en pantalla.

#### 5.6.3.8 Especificación componente de wiki

- **wiki\_save:** Función ubicada en la capa controlador que permite publicar una nueva entrada wiki.
- **wiki\_edit:** Función ubicada en la capa controlador que permite modificar una entrada wiki existente, donde se permiten los siguientes parámetros:
  - **entry:** Datos que representan la definición general de la entrada wiki.
  - **terms:** Opción que permite indicar los términos relacionados.
  - **references:** Opción que permite indicar las referencias de la definición.
  - **links:** Opción que permite indicar los enlaces de la definición wiki.
  - **tag:** Opción que permite marcar la entrada wiki como etiqueta.
- **wiki\_history:** Función ubicada en la capa controlador que permite ver el historial de cambios de la entrada wiki.

#### *5.6.3.9 Especificación proceso de actualización base de datos*

- **queue\_get:** Función ubicada en la capa controlador que permite obtener la cola de mensajes de actualización desde la base de datos secundaria hacia la primaria.
- **queue\_process:** Función ubicada en la capa controlador que procesa un mensaje desde la cola de mensajes.
- **search\_changes:** Función ubicada en la capa controlador que se encarga de buscar cambios en la base de datos secundaria.
- **semantic\_data\_collect:** Función ubicada en la capa controlador que se encarga de recolectar todos los cambios producidos en la base de datos secundaria.
- **update\_changes:** Función ubicada en la capa controlador que se encarga de realizar los cambios en la base de datos primaria.
- **update\_priorities:** Función ubicada en la capa controlador que se encarga de actualizar los puntajes y prioridades de las personas que realizaron estos cambios.
- **update\_activities:** Función ubicada en la capa controlador que se encarga de actualizar las actividades de las personas que realizaron estos cambios.

#### *5.6.3.10 Especificación proceso de sensibilización*

- **update\_preferences\_sens:** Función ubicada en la capa controlador que se encarga de actualizar las preferencias gatilladas por las personas.
- **queue\_put:** Función ubicada en la capa controlador que permite enviar un mensaje a la cola de mensajes con la actualización de preferencias.
- **queue\_process:** Función ubicada en la capa controlador que procesa un mensaje desde la cola de mensajes.
- **update\_semantic\_data\_sens:** Función ubicada en la capa controlador que se encarga de actualizar los datos semánticos de la persona que actualiza las preferencias.
- **update\_priorities\_sens:** Función ubicada en la capa controlador que se encarga de actualizar los puntajes y prioridades de las personas que realizaron estos cambios.

#### *5.6.3.11 Especificación proceso de actualización de wikis*

- **tag\_new:** Función ubicada en la capa controlador que permite crear una nueva etiqueta sobre algún elemento.
- **select\_tag\_to\_wiki:** Función ubicada en la capa controlador que permite marcar una etiqueta existente como una nueva entrada wiki.

- **tag\_to\_wiki:** Función ubicada en la capa controlador que convierte una etiqueta marcada como una nueva entrada wiki.
- **wiki\_edit:** Función ubicada en la capa controlador que permite modificar una entrada wiki existente, donde se permiten los siguientes parámetros:
  - **entry:** Datos que representan la definición general de la entrada wiki.
  - **terms:** Opción que permite indicar los términos relacionados.
  - **references:** Opción que permite indicar las referencias de la definición.
  - **links:** Opción que permite indicar los enlaces de la definición wiki.
  - **tag:** Opción que permite marcar la entrada wiki como etiqueta.

#### *5.6.3.12 Especificación proceso de recomendación*

- **recommendations\_view:** Función ubicada en la capa controlador que permite visualizar la sección de recomendaciones.
- **get\_priorities:** Función ubicada en la capa controlador que permite extraer las prioridades de la persona que está visualizando la sección de recomendaciones.
- **search\_elements:** Función ubicada en la capa controlador que permite buscar todos los elementos recomendados según las prioridades, donde tiene como parámetro:
  - **tipo:** Valor que indica si los elementos están según las prioridades de la persona o según la prioridad global.
- **display\_results:** Función ubicada en la capa controlador que despliega los resultados obtenidos de la búsqueda de elementos para la visualización de recomendaciones.

#### *5.6.3.13 Especificación proceso de actividades*

- **activities\_view:** Función ubicada en la capa controlador que permite visualizar la sección de últimas actividades.
- **search\_elements:** Función ubicada en la capa controlador que permite buscar las últimas actividades.
- **display\_results:** Función ubicada en la capa controlador que despliega los resultados obtenidos de la búsqueda de últimos elementos.

#### *5.6.3.14 Especificación proceso de estadísticas*

- **statistics\_view:** Función ubicada en la capa controlador que permite visualizar la sección de estadísticas.

- **search\_statistics:** Función ubicada en la capa controlador que permite obtener las estadísticas.
- **display\_results:** Función ubicada en la capa controlador que despliega los resultados obtenidos de la búsqueda de estadísticas.

#### 5.6.4 Diseño gestor de colas de mensaje

Para cumplir con el requisito de procesar mensajes desde y hacia distintos componentes de la plataforma, se especifican los diferentes tipos de gestor de colas de mensajes.

##### 5.6.4.1 Especificación de colas de mensaje

###### a. Cola con demora de tiempo:

- **DelayedQueue:** Cola de mensajes que se encarga de procesar los mensajes de manera retardada, definida con un tiempo prudente para no saturar el servidor de peticiones.

###### b. Cola sin demora de tiempo:

- **NotDelayedQueue:** Cola de mensajes que se encarga de procesar los mensajes de manera inmediata, donde los procedimientos a ejecutar no son pesados para el servidor, de manera de no saturarlo.

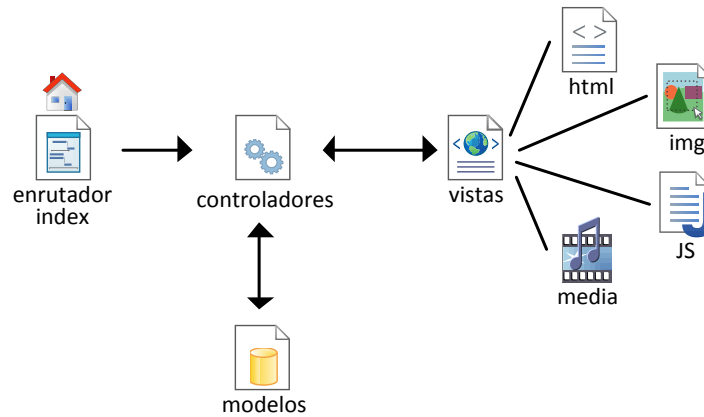
### 5.7 Implementación de la plataforma

En esta fase se realizó la implementación de la plataforma, se realizó la codificación del diseño realizado en la etapa anterior en base a la utilización del framework MVC que se escogió. A continuación se detalla cómo se utilizó el framework MVC y la codificación de la plataforma.

#### 5.7.1 Utilización del Framework

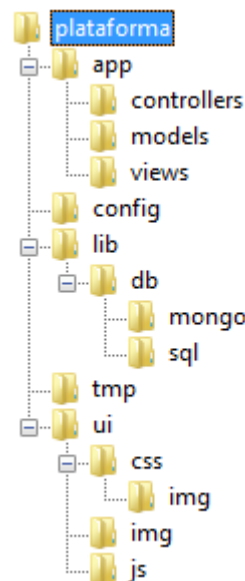
El framework que se escogió es Fat-Free, el que es un micro-framework liviano, de utilización simple, para construir sistemas web dinámicos y robustos. Funciona de una manera que hay que definir una estructura de carpetas, donde se encontrarán los archivos que componen las tres capas de modelo, vista y controlados, junto con otros archivos de configuración y de GUI, como se muestra gráficamente en la figura 5.3. Se debe cumplir con las características definidas en la especificación de servidor de aplicaciones para que funcione correctamente. El micro-framework trabaja de forma que debe haber un enrutador, que mapea las rutas de la URL, llamando las funciones

del controlador, que se comunican con la capa de modelo para obtener los datos e indica cuál es el elemento que debe visualizar de la capa de vista.



**Figura 5.3: Utilización del framework.**

Para cumplir con las características del framework, se definió una estructura de carpetas como se puede ver en la figura 5.4 y detallado en el Anexo A.



**Figura 5.4: Estructura de carpetas del proyecto.**

## 5.7.2 Codificación de la plataforma

Basados en la estructura de las carpetas del proyecto, se codificaron los archivos en lenguaje PHP orientado a objetos, utilizando el framework Fat-Free, en todas sus capas y utilizando librerías adicionales tanto de PHP como de JavaScript, como se detalla a continuación:

### 5.7.2.1 Enrutador

El enrutamiento de la plataforma se encuentra directamente en el archivo *index.php* de la carpeta principal de la plataforma. En el contenido del archivo se puede apreciar que importa la clase base del framework, setea el nivel de error, importa la configuración de la plataforma, define el tipo de codificación UTF-8 y finalmente define el enrutamiento junto con la función es llamada del controlador. El detalle de este enrutador se puede ver en el Anexo B.

### 5.7.2.2 Controladores

Se definen cuatro controladores que son la base de la plataforma, donde el principal controlador es *KM.php*, luego están los controladores *Chat.php*, *Documents.php* y finalmente *Images.php*, como se explican en la siguiente tabla:

| Controlador   | Descripción   |
|---------------|---|
| KM.php        | Controlador principal de la plataforma que agrupa todas las funciones esenciales del sistema. |
| Chat.php      | Controlador que se utiliza para la comunicación vía chat a través de la plataforma.           |
| Documents.php | Controlador que maneja la carga y descarga de archivos.                                       |
| Images.php    | Controlador que administra el manejo y gestión de imágenes.                                   |

**Tabla 5.1: Controladores de la plataforma.**

En el contenido del controlador *KM.php*, que está detallado en el Anexo C, se puede apreciar que al inicio importa una librería para envío de emails, luego se definen las variables globales, a continuación se define el constructor del controlador, que utiliza el patrón de diseño *singleton* para mantener una única instancia de la clase. Siguen dentro del constructor, las dos conexiones a la base de datos primaria y secundaria, luego se



define el tipo de archivo MIME, seguido de la validación de la sesión del usuario y finalmente el layout que va a corresponder según si viene de AJAX, tiene sesión o no tiene sesión. Por último siguen todas las funciones controladoras que se definieron en el enrutador, y como ejemplo ilustrativo, está la función *about()*, donde se define el tipo de la página en que se está, y validando que el usuario esté autenticado, indica la vista que debe ser visualizada, en este caso *about.html*, definiendo algunas variables o en caso contrario *about2.html*. El contenido del controlador *Chat.php*, detallado en el Anexo D, muestra al igual que el controlador anterior, que define las variables globales y define al constructor junto con las funciones requeridas desde el enrutador. En el contenido del controlador *Documents.php*, detallado en el Anexo E, se ve que sigue el mismo patrón del controlador anterior. Por último, el controlador *Images.php*, detallado en el Anexo F, también sigue el mismo patrón de los controladores anteriores.

### 5.7.2.3 Modelos

En la capa de modelo, se mapean sólo las colecciones de la base de datos primaria de MongoDB, que se especificaron anteriormente, ya que son las requeridas para esta etapa inicial de la plataforma, las que son: *Activities.php*, *Entries.php*, *Galleries.php*, *Groups.php*, *Logs.php*, *Messages.php*, *Sensibilities.php* y *Users.php*. Estos archivos están detallados en el Anexo G. Todos los modelos mapeados siguen el mismo patrón, utilizando las mismas funciones del tipo CRUD para operar sobre las colecciones de MongoDB.

### 5.7.2.4 Vistas

En esta capa, se generaron las vistas que se indicaron en la los cuatro controladores definidos anteriormente, cumpliendo así con las funcionalidades básicas de la plataforma. El contenido de cada uno de estos archivos es HTML puro, con algunos tags especiales con el fin de incluir contenido dinámico obtenido desde los controladores. En la siguiente tabla se explican todas las vistas utilizadas en la plataforma:

| Vista       | Descripción  |
|-------------|--|
| about.html  | Vista que muestra información acerca del proyecto con sesión de usuario. |
| about2.html | Vista que muestra información acerca del proyecto sin sesión de usuario. |

|                   |   |
|-------------------|---|
| activate.html     | Vista que sirve para activar una nueva cuenta.              |
| blank.html        | Vista que no tiene contenido.                               |
| chat.html         | Vista para visualizar el chat de la plataforma.             |
| dashboard.html    | Vista del muro o la página principal de la plataforma.      |
| invitations.html  | Vista para visualizar la sección de invitaciones.           |
| js-dashboard.html | Vista para incluir código JS de la página principal.        |
| js-profile.html   | Vista para incluir código JS de la página del perfil.       |
| js-search.html    | Vista para incluir código JS de la página de búsqueda.      |
| js-wiki.html      | Vista para incluir código JS de la página de wiki.          |
| layout.html       | Vista del diseño utilizado en la plataforma con sesión.     |
| layout2.html      | Vista del diseño utilizado en la plataforma sin sesión.     |
| locations.html    | Vista para visualizar la sección de ubicaciones.            |
| login.html        | Vista del ingreso a la plataforma.                          |
| lostPassword.html | Vista para recuperar la contraseña.                         |
| policies.html     | Vista de la sección de políticas de uso.                    |
| profile.html      | Vista de perfil de la plataforma.                           |
| profile_edit.html | Vista para editar el perfil.                                |
| question.html     | Vista para visualizar la sección de preguntas y respuestas. |
| search.html       | Vista del buscador de la plataforma.                        |
| settings.html     | Vista de la configuración de la plataforma.                 |
| wiki.html         | Vista de la aplicación de wiki.                             |
| wiki_detail.html  | Vista para ver el detalle de una definición wiki.           |
| wiki_edit.html    | Vista para editar una entrada wiki.                         |
| wiki_history.html | Vista para ver el historial de modificaciones de una wiki.  |

**Tabla 5.2: Tabla con el detalle de las vistas de la plataforma.**

A continuación, en la figura 5.5, se muestra el mapa de navegación de la plataforma para todos los perfiles:

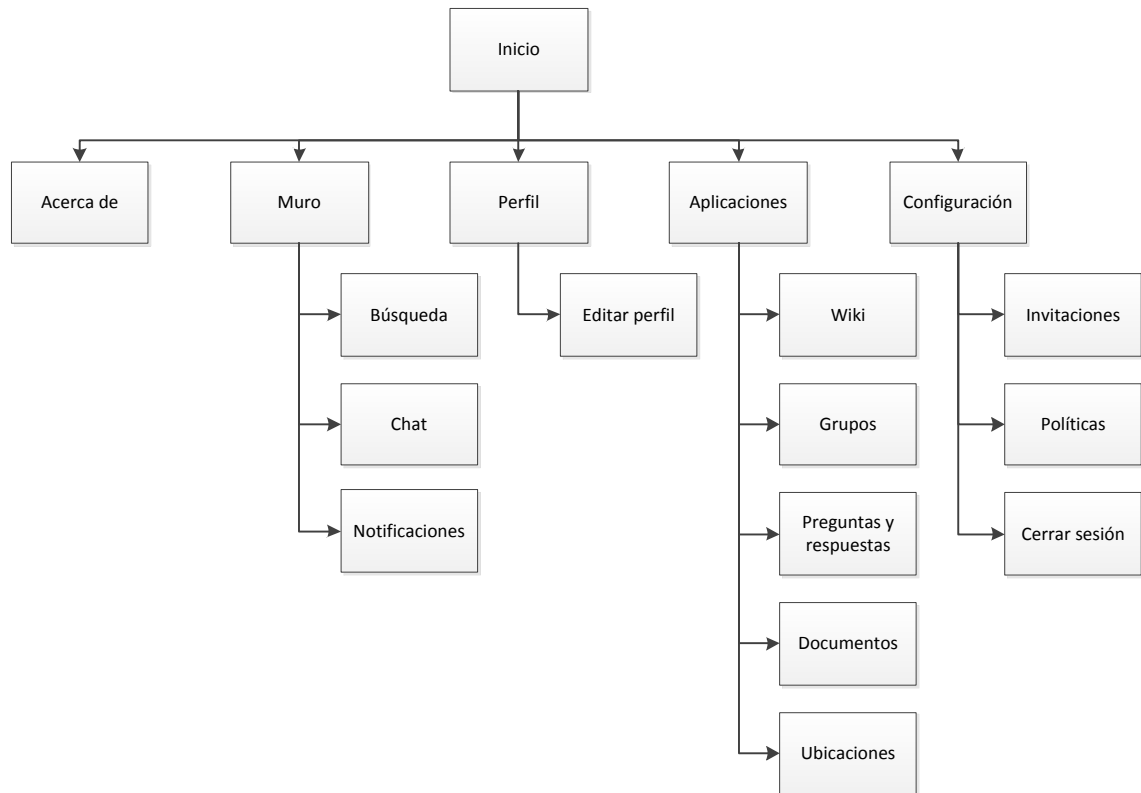


Figura 5.5: Mapa de navegación de la plataforma.

### 5.7.3 Base de datos

Para poder utilizar la base de datos MongoDB, primero se creó la base de datos *km* desde la consola del motor, utilizando el comando *use km*. Una vez creada la base de datos, se comenzó a crear las colecciones. Éstas por no seguir la estructura clásica de SQL, se crean sin campos ni tipos de datos, sólo deben contener el nombre que utilizará, como se muestra el script ejecutado en la siguiente tabla:

| Script para la creación de colecciones de la base de datos |
|--|
| <code>db.createCollection("activities");</code>            |
| <code>db.createCollection("entries");</code>               |
| <code>db.createCollection("galleries");</code>             |
| <code>db.createCollection("groups");</code>                |
| <code>db.createCollection("logs");</code>                  |
| <code>db.createCollection("messages");</code>              |

```
db.createCollection("sensibilities");
db.createCollection("users");
```

**Tabla 5.3: Tabla con el script de creación de colecciones.**

### 5.7.4 Implementación de la interfaz gráfica

En la implementación de la interfaz gráfica se trabajó con varias librerías adicionales para poder lograr una experiencia de usuario (UX) similar a la que se tiene en las redes sociales más comunes. Principalmente se trabajó con librerías JavaScript jQuery y AJAX, como así también hojas de estilo CSS versión 3.

En la siguiente tabla se explican en detalle las librerías que se trabajaron y hojas de estilo:

| Librería                   | Descripción   |
|----------------------------|---|
| dashboard.js               | Llamados de JavaScript que se utiliza en el muro o página principal de la plataforma.                             |
| jquery-1.10.2.min.js       | Framework JavaScript multipropósito que permite simplificar la codificación, utilizado por múltiples librerías.   |
| jquery.isotope.min.js      | Librería JavaScript que sirve para ordenar elementos dinámicamente en forma de línea de tiempo, requiere jQuery.  |
| jquery.tagcanvas.min.js    | Librería JavaScript que genera una nube de etiquetas dinámicas en 3D, requiere jQuery.                            |
| jquery.tagcloud.js         | Librería JavaScript que genera una nube de etiquetas dinámicas en forma de texto, requiere jQuery.                |
| jquery.timeago.js          | Librería JavaScript que sirve para mostrar hace cuánto tiempo sucedió un evento, requiere jQuery.                 |
| jquery.timer.js            | Librería JavaScript que sirve para definir tiempos de ejecución sobre algún evento, requiere jQuery.              |
| jquery-ui-1.10.4.custom.js | Librería JavaScript personalizada que contiene algunas funcionalidades de UI y efectos visuales, requiere jQuery. |
| jsDatePick.min.1.3.js      | Librería JavaScript que sirve para seleccionar una fecha dentro de un calendario desplegado.                      |
| lightbox-2.6.min.js        | Librería JavaScript que se utiliza para visualizar imágenes en pantalla completa dentro de la plataforma.         |

|                             |   |
|-----------------------------|---|
| modernizr.custom.js         | Librería JavaScript que sirve para detectar capacidades del navegador y permite trabajar con browsers que son más antiguos. |
| zaria.js                    | Librería JavaScript que sirve como editor de texto enriquecido a través de un formulario web.                               |
| Htmlparser.js               | Librería JavaScript para parsear código HTML, requerida por la librería zaria.js.   |
| style.css                   | Hoja de estilo principal que incluye todos los diseños de la plataforma.  |
| dashboard.css               | Hoja de estilo con características especiales para el muro o página principal de la plataforma.                             |
| jquery-ui-1.10.4.custom.css | Hoja de estilo con elementos necesarios de la librería jQuery UI incluida.  |
| lightbox.css                | Hoja de estilo para diseñar los elementos de la visualización de imágenes.  |

**Tabla 5.4:** Tabla con el detalle de librerías y hojas de estilo.

## 5.8 Despliegue de los servidores

Este proceso incluyó la obtener los servidores, la preparación de los mismos, instalación de software y servicios, compra de un dominio de internet, configuración de servidores DNS, de email para envío de correos vía SMTP, tal como se especificó en los requisitos de hardware. En esta etapa se configuró el hardware y se configuraron los componentes de software de la plataforma de GC. Los objetivos de esta fase fueron los siguientes:

- Despliegue de la infraestructura de hardware, componentes de software y habilitación de dominio de internet vía DNS.
- Paquetes de software para cubrir los requerimientos.

A continuación se describen las actividades ejecutadas para alcanzar los objetivos definidos para esta etapa.

### 5.8.1 Obtención de dominio internet

Para cumplir con el propósito de este proyecto se obtuvo un dominio de internet desde NIC Chile, manteniendo el mismo nombre que se definió en la etapa de diseño:

- mykommunity.cl

### 5.8.2 Instalación de servidor

En esta etapa, debido a que el sistema está en una fase de pruebas, se utilizó un sólo servidor físico, para instalar el servidor de aplicaciones, de base de datos y de documentos. Se encargó a una empresa externa un servidor pre-configurado, donde solamente posea el sistema operativo instalado y con un acceso SSH hacia el servidor a través de una IP pública. Una vez conseguido este servidor se procedió con la instalación y configuración. La configuración inicial fue la siguiente:

- **Sistema operativo:** Linux Ubuntu versión 12.
- **IP's:** 1 IP pública.
- **SSH:** Acceso con usuario root.
- **Servicios instalados:** No.

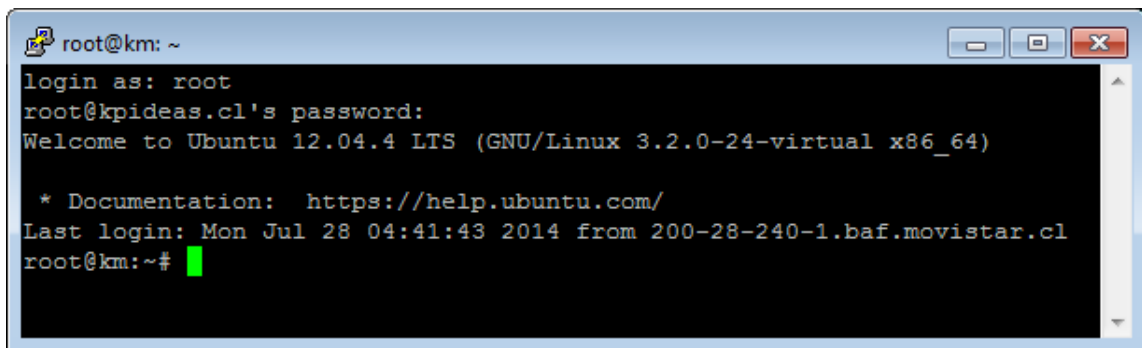


Figura 5.6: Acceso vía SSH a la consola del servidor.

### 5.8.3 Instalación y configuración de paquetes en servidor

En esta etapa se instalaron todos los paquetes y servicios requeridos, luego se hizo una configuración general de todos los servicios, como se detalla en la siguiente tabla:

| Paquete instalado | Descripción  |
|-------------------|--|
| bind9             | Paquete del servidor DNS con el fin de habilitar el nombre de dominio. |
| apache2           | Paquete con el servidor web Apache 2.                                  |
| apache2-doc       | Paquete con la documentación del servidor web.                         |
| apache2-utils     | Paquete con funcionalidades esenciales del servidor web.               |
| php5              | Paquete que posee el intérprete de código PHP 5.                       |

|                     |  |
|---------------------|--|
| libapache2-mod-php5 | Paquete que contiene la librería para interpretar código PHP en el servidor web. |
| php-pear            | Paquete con funcionalidades esenciales del intérprete PHP.                       |
| php5-dev            | Paquete de PHP necesario para compilar otros módulos.                            |
| php5-gd             | Paquete de PHP necesario para trabajar con gráficos.                             |
| vsftpd              | Paquete del servidor FTP, necesario para la transferencia de archivos.           |
| Mongodb             | Paquete que posee el servidor de base de datos MongoDB.                          |
| php5-mongo          | Paquete que posee la librería para que PHP reconozca la base de datos MongoDB.   |
| mysql-server        | Paquete que posee el servidor de base de datos MySQL.                            |
| php5-mysql          | Paquete que posee la librería para que PHP reconozca la base de datos MySQL.     |
| Phpmyadmin          | Paquete que incluye un cliente web de visualización de base de datos MySQL.      |

**Tabla 5.5: Descripción de paquetes instalados en servidor.**

A continuación se detallan las configuraciones que se debieron realizar para cada uno de los servidores instalados.

### 5.8.3.1 Configuración servidor DNS

Fue requerido configurar el servidor DNS, para esto se editaron los archivos */etc/bind/named.conf.local*, */var/cache/bind/db.mykommunity.cl* y */var/cache/bind/db.190.114.254*, como se detalla en el Anexo H.

Una vez realizados estos cambios, se ingresó como servidor de nombres DNS primario el host recién configurado en NIC Chile para redirigir el nombre de dominio mykommunity.cl al servidor web:

- DNS primario: km.mykommunity.cl

### 5.8.3.2 Configuración servidor Web

Para la configuración del servidor web, se tomaron los valores por defecto tanto para el archivo `/etc/apache2/apache2.conf` de Apache, como para el archivo `/etc/php5/apache2/php.ini` de PHP. Para habilitar el sitio web se creó el archivo de configuración `/etc/apache2/sites-enabled/mykommunity.cl` con el contenido que se detalla en el Anexo I.

### 5.8.3.3 Configuración servidor FTP

La configuración del servidor FTP se realizó editando el archivo `/etc/vsftpd.conf`, con algunos parámetros adicionales a los originales, como se muestra en la siguiente tabla:

| Contenido de la configuración del servidor FTP  |
|---|
| <pre>anonymous_enable=NO local_enable=YES write_enable=YES chroot_local_user=YES chroot_list_enable=YES</pre> |

Tabla 5.6: Detalle de la configuración del servidor FTP.

### 5.8.3.4 Configuración servidor Base de Datos MongoDB

En la configuración de la base de datos MongoDB, se mantuvo la configuración inicial, como se detalla a continuación el archivo `/etc/mongodb.conf`:

| Contenido de la configuración del servidor MongoDB  |
|---|
| <pre>dbpath=/var/lib/mongodb logpath=/var/log/mongodb/mongodb.log logappend=true bind_ip = 127.0.0.1 port = 27017</pre> |

Tabla 5.7: Detalle de la configuración del servidor MongoDB.

### 5.8.3.5 Configuración servidor Base de Datos MySQL

En el caso de la base de datos MySQL, también se mantuvo la configuración inicial desde el archivo `/etc/mysql/my.cnf`, como se puede ver en el Anexo J.



## 5.9 Despliegue de la plataforma

Una vez desplegado el servidor de aplicaciones y configurado todos sus servicios, se subieron los archivos del sitio utilizando un usuario habilitado, a la carpeta definida en el servidor de aplicaciones, que en este caso es */var/www/mykommunity.cl*. En el cuadro del Anexo K se muestran las carpetas creadas y los archivos que fueron subidos a través del FTP.

### 5.9.1 Configuración de la plataforma

Para configurar la plataforma, se debió editar el archivo *setup.cfg*, asignando los parámetros requeridos. Una vez editado este archivo con los datos necesarios, se subió por FTP a la carpeta *config/* de la plataforma. El detalle de los parámetros se detalla en el Anexo L.

También es requisito editar el archivo *.htaccess* ubicado en la carpeta principal de la plataforma, con el fin de permitir que el enrutamiento del framework Fat-Free desde *index.php* para que funcione correctamente, también detallado en el Anexo L.

## 5.10 Pruebas de la plataforma

En esta etapa se confeccionó un plan de pruebas donde se detallaron los casos de prueba con sus respectivos resultados esperados. En esta fase fueron probadas las funcionalidades de la plataforma de GC desde los siguientes puntos de vista:

- Pruebas de aseguramiento de la calidad.
- Pruebas funcionales.
- Pruebas de sistema.
- Pruebas de aceptación.

### 5.10.1 Pruebas de aseguramiento de calidad

En esta etapa se ejecutaron pruebas de QA necesarias para detectar errores y resolver fallas antes de la puesta en producción del sistema, a través del método de pruebas unitarias. Los problemas encontrados fueron recopilados para ser resueltos antes de la aceptación final, de manera de asegurar la calidad del software para que no tenga errores en los datos, lógica o algoritmos.

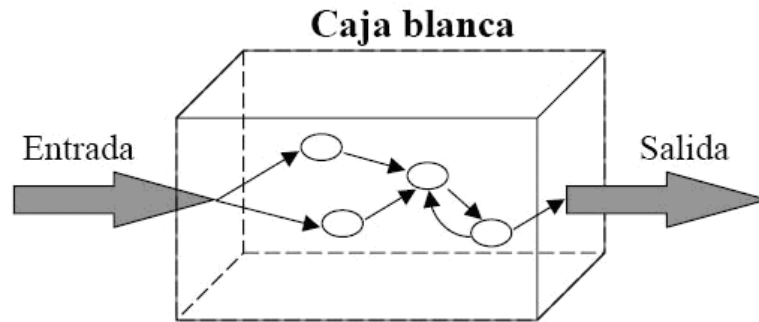


Figura 5.7: Método de la caja blanca para pruebas unitarias.

### 5.10.2 Pruebas funcionales y de sistema

Con el fin de detectar errores en la implementación y fallas en el cubrimiento de los requerimientos, antes de entregárselo a los usuarios, se definió la etapa de pruebas funcionales y de sistema, con el fin de asegurar que exista una apropiada aceptación de datos y que exista un adecuado procesamiento, recuperación e implementación de las reglas de negocio.

En la siguiente tabla se describen las principales pruebas funcionales y de sistema junto con los resultados obtenidos para validar el funcionamiento de la plataforma desplegada:

| N°    | Prueba                                     | Resultado esperado   |
|-------|--|--|
| PF-01 | Ingreso al sistema y cambio de contraseña. | Usuario debe poder ingresar al sistema y poder visualizar la página principal y su perfil personal. También debe poder cambiar su contraseña desde la configuración. |
| PF-02 | Búsqueda de contenido.                     | El usuario debe ser capaz de poder realizar una búsqueda personalizada, obteniendo resultados acordes a la búsqueda y posibilidad de entrar al contenido.            |
| PF-03 | Publicar y editar entrada wiki.            | El usuario puede crear una entrada wiki, editarla y ver el historial de cambios.   |
| PF-04 | Publicar pregunta y respuesta.             | La persona puede ser capaz de realizar una pregunta en la comunidad y poder publicar una respuesta sobre las preguntas existentes.                                   |
| PF-05 | Publicación de documentos.                 | Desde el muro y el perfil, el usuario debe ser capaz de publicar un documento y visualizarlo en su perfil y página principal.  |

|       |                          |   |
|-------|--------------------------|---|
| PF-06 | Visualización de grupos. | El usuario debe poder visualizar los grupos de personas.                        |
| PF-07 | Uso de Chat.             | El usuario puede utilizar la mensajería instantánea a través del ícono de Chat. |

**Tabla 5.8: Pruebas funcionales.**

### 5.10.3 Pruebas de aceptación final

Las pruebas de aceptación fueron diseñadas para dar la aceptación de la plataforma antes de pasar a la marcha blanca. En cuanto a la ejecución de las pruebas de aceptación, se escogieron dos de las personas que trabajarán en la marcha blanca, del área de sistemas para realizar esta tarea, con el fin de que detecten fallas en la implementación del sistema. Los objetivos de esta fase fueron los siguientes:

- Planificación y preparación de las condiciones de prueba y de los resultados esperados.
- Ejecución de pruebas para asegurar que la plataforma cubre los requerimientos funcionales.

Para cumplir con estos objetivos se llevaron a cabo las siguientes actividades:

| N°    | Prueba                  | Resultado esperado  |
|-------|-------------------------|---|
| PA-01 | Ingreso al sistema.     | Usuario debe poder ingresar al sistema y poder visualizar la página principal y su perfil personal.                                   |
| PA-02 | Edición de perfil.      | La persona debe ser capaz de personalizar su propio perfil, editando los datos e imágenes.  |
| PA-03 | Publicación en el muro. | El usuario debe ser capaz de realizar una publicación en la página principal de la plataforma y que quede registrado en el historial. |
| PA-04 | Búsqueda de contenido.  | El usuario debe ser capaz de poder realizar una búsqueda personalizada, obteniendo resultados acordes a su búsqueda.                  |
| PA-05 | Uso de aplicaciones.    | La persona debe poder seleccionar y usar una aplicación desde el menú.  |
| PF-06 | Invitación a personas.  | El usuario debe poder invitar a más personas de la compañía a través del menú Invitación.   |

**Tabla 5.9: Pruebas de aceptación final.**

Las pruebas de aceptación finales constituyeron la etapa final para la implementación de la plataforma de GC. Después de resolver todos los problemas encontrados durante el período de prueba, la prueba de aceptación final se realizó para dar una evaluación final de la plataforma y las aplicaciones desarrolladas, antes de comenzar con la puesta en producción del sistema.

## 5.11 Puesta en producción del sistema

Debido a que este proyecto de tesis llega hasta la etapa de marcha blanca, es el mismo ambiente de pruebas el que se utiliza como puesta en producción, entregándoles acceso a los usuarios libremente.

### 5.11.1 Marcha blanca

En esta etapa de puesta en producción, se definió un período de marcha blanca de un mes, para que los usuarios pudieran utilizar la plataforma y evaluar el funcionamiento de ésta, para finalmente obtener retroalimentación que sirva para concluir que este proyecto cumplió o no con los objetivos planteados en el comienzo de esta tesis.

Algunos ejemplos del uso que se le dio a la plataforma, se muestran en la figura 5.8, figura 5.9 y figura 5.10.

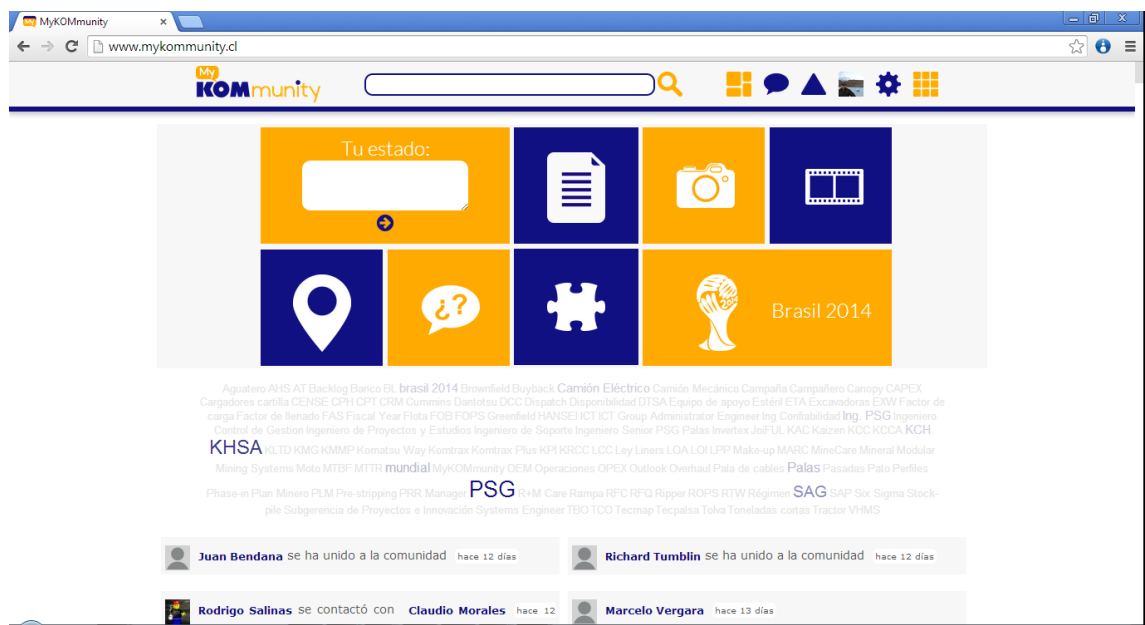


Figura 5.8: Actividad reflejada durante la marcha blanca.

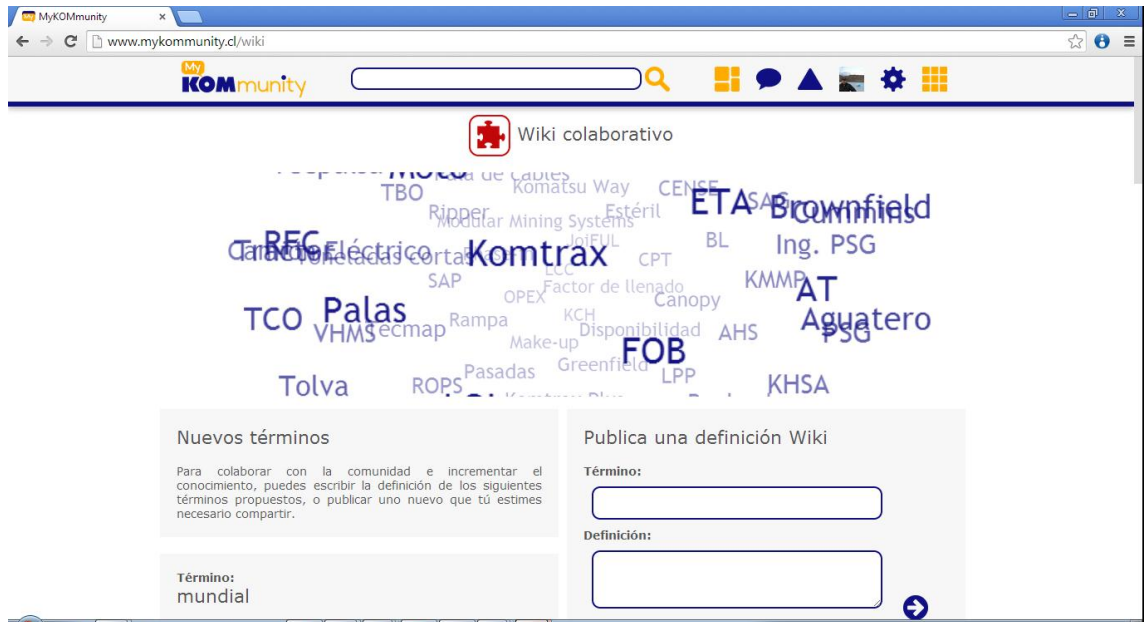


Figura 5.9: Términos definidos durante la marcha blanca.

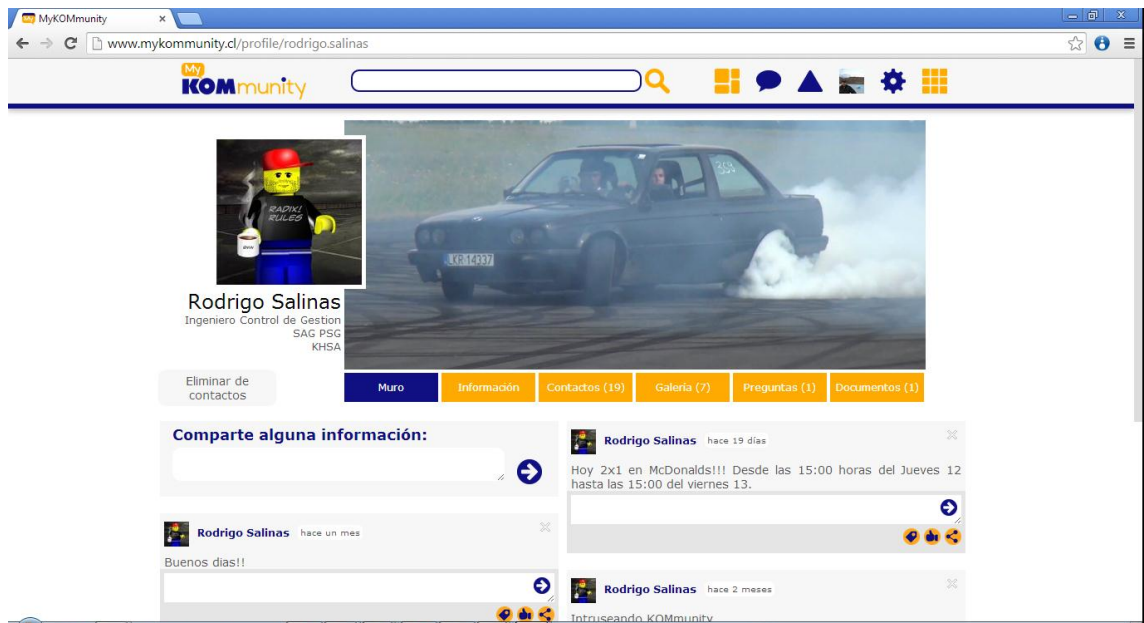


Figura 5.10: Ejemplo de perfil creado durante la marcha blanca.

### 5.11.2 Encuesta experiencia de usuario

Para medir los resultados de la utilización de esta plataforma, se realizó una encuesta de experiencia de usuario, a las 10 personas más representativas, dentro de los 74 usuarios registrados durante la marcha blanca, con el fin de poder obtener una apreciación desde todos los puntos de vista de la plataforma, desde el uso, la facilidad, el diseño, la funcionalidad y la utilidad, considerando los objetivos específicos definidos para esta Tesis.

A continuación se muestra la encuesta realizada con los promedios de cada pregunta, donde un 1 es la nota más baja y 7 la más alta, junto con el número de objetivo específico a la que se aplica:

| Pregunta  | Objetivo específico | Promedio nota |
|---|---------------------|---------------|
| Me fue sencillo utilizar MyKommunity.   | 5                   | 6.5           |
| Este sistema puede ser de gran ayuda para el trabajo que realizo.   | 1 y 2               | 6.2           |
| Las aplicaciones del sistema me fueron de utilidad (documentos, wiki, grupos, preguntas, fotos, etc).                                 | 3                   | 6.2           |
| MyKommunity me ayudó a compartir información de interés con mis demás compañeros.   | 4                   | 6.7           |
| MyKommunity me permite acelerar el tiempo cuando comparto documentos, fotos o información relevante.                                  | 2 y 3               | 6.5           |
| Considero que este sistema me facilita la cercanía y comunicación con mis compañeros, sobre todo los que están más lejanos.           | 4                   | 6.8           |
| Considero que este sistema puede servir como un repositorio de información que genere una memoria organizacional para la empresa.     | 1                   | 5.8           |
| El sistema me permite capturar, organizar y almacenar información relevante para mí.  | 1 y 3               | 6.2           |
| Además, me es útil para crear grupos facilitando el intercambio de información, el flujo de ideas, dudas, experiencias e inquietudes. | 1, 2 y 4            | 6.2           |
| Creo que el sistema podría ser un espacio para aprender colaborativamente.  | 2                   | 6.8           |
| La plataforma se comportó correctamente, sin errores ni caídas.   | 5                   | 5.7           |

|   |                |     |
|---|----------------|-----|
| Me dio confianza utilizar MyKommunity.    | 5              | 6.7 |
| Me gustó el diseño de la plataforma.      | 5              | 6.7 |
| MyKommunity cumplió con mis expectativas. | 1, 2, 3, 4 y 5 | 6.3 |

**Tabla 5.10: Encuesta de experiencia de usuario.**

En el siguiente capítulo se realiza un análisis de estos resultados, con el fin de verificar el cumplimiento de los objetivos específicos definidos anteriormente.

### 5.11.3 Estadísticas de uso en marcha blanca

Con la puesta en producción de la plataforma, se obtuvieron una serie de estadísticas de uso durante el período de marcha blanca, los cuales sirven para contribuir a un mejor análisis de resultados, desarrollado en el siguiente capítulo.

| Estadística de uso   | Medición  |
|--|---|
| Usuarios invitados a la plataforma.  | 122   |
| Usuarios que activaron su cuenta.  | 74  |
| Usuarios que completaron su perfil.  | 24  |
| Usuarios que perdieron y recuperaron su contraseña.                              | 18  |
| Usuarios que publicaron algún contenido.   | 54  |
| Número de publicaciones totales.   | 147   |
| Número de entradas wiki publicadas.  | 103   |
| Número de documentos subidos.  | 12  |
| Número de preguntas y respuestas publicadas.                                     | 10  |
| Número de sesiones activas durante todo el período de marcha blanca <sup>3</sup> | 769   |
| Número de páginas vistas *   | 5472  |
| Páginas vistas por sesión *  | 7.12  |
| Duración media de la sesión *  | 00:07:46  |
| Ciudades de origen de conexión *   | Santiago 95.5%,<br>Antofagasta 3.2%,<br>Concepción 0.5% |
| Navegadores más utilizados *   | Chrome 93.5%,<br>Firefox 4.3%                           |

<sup>3</sup> Las estadísticas marcadas con \* se obtuvieron a través del sistema Google Analytics instalado en la plataforma: <https://www.google.com/analytics/>

|                                      |  |
|--------------------------------------|--|
| Sistemas operativos más utilizados * | <b>Windows 93.4%,</b><br><b>Android 5.1%,</b><br><b>iOS 1.6%</b> |
|--------------------------------------|--|

**Tabla 5.11: Estadísticas de uso en marcha blanca.**

## 5.12 Contribución del capítulo

En este capítulo se presentó la implementación de la plataforma, utilizando la metodología propuesta en base a la planificación del proyecto, la que parte desde el diseño técnico, donde se realizó la elección del framework y librerías, pasando por la codificación de la plataforma, realizando el despliegue de los servidores y sistema, ejecutando las pruebas de aseguramiento de calidad para finalmente iniciar la puesta en marcha de la plataforma en la que los usuarios tuvieron un tiempo determinado para utilizar la plataforma y realizar una evaluación final.



## Conclusiones y trabajo a futuro

### 6.1 Análisis por capítulo

Para concluir y comentar en detalle el trabajo realizado en esta tesis, se realizará el análisis por capítulo en orden secuencial.

#### Capítulo 1:

- La memoria organizacional consiste en el almacenamiento y preservación del conocimiento dentro de las organizaciones con la finalidad de reutilizarlo e ir mejorando los procesos que la experiencia va dando. Sin embargo, las empresas no siempre manejan de manera correcta la memoria organizacional. Por otro lado, la gestión del conocimiento implica una evolución en la cultura organizacional, lo que significa un cambio de estructuras organizativas y transformación de procesos. Dentro de muchas organizaciones existen sistemas que apoyan la administración organizacional, sin embargo, éstos son a menudo muy estructurados, muchos de ellos con interfaces obsoletas, lo que da como resultado que las personas se sientan poco atraídas para interactuar con ellas.
- Cuando las organizaciones tienden a favorecer la creación y administración del conocimiento y así generar ventajas competitivas, logran mantener una memoria organizacional y permiten un mayor aprendizaje colaborativo. Esto se puede lograr de forma más eficiente gracias a la propuesta hecha en este proyecto, la cual se basa en el diseño y construcción de una plataforma de gestión del conocimiento con arquitectura de aprendizaje colaborativo bajo las características de interfaz de red social y alta usabilidad y disponibilidad 100% enfocada a las personas.
- La propuesta del presente documento está dirigida al holding de empresas de Komatsu Chile, la que presenta problemas de gestión del conocimiento, aprendizaje colaborativo y memoria organizacional debido a la situación geográfica que presentan muchos de sus colaboradores, la movilidad e incremento de personal constante, la falta de historial de las actividades y la ausencia de repositorios de información.

- La solución planteada pretende subsanar los problemas mencionados, permitiendo la facilitación de la extracción de la información, el intercambio y flujo de ideas, dudas, experiencias e inquietudes, que darán como resultado el aprendizaje colaborativo y la memoria organizacional gestionando la información relevante.

### Capítulo 2:

- Los conceptos de “Memoria Organizacional y Gestión del Conocimiento”, “Aprendizaje Colaborativo” y “Redes Sociales”, son importantes en el desarrollo de este proyecto, sin embargo, lo más destacado del capítulo 2 reside en el resultado que surge de la fusión entre los 3.
- En la fusión de la *memoria organizacional* y el *aprendizaje colaborativo*, se podría resumir que se aprende compartiendo, porque el conocimiento reside en las personas y son estas las que generan gestión del conocimiento y memoria organizacional en el largo plazo. Por otro lado, en la fusión del *aprendizaje colaborativo* y *las redes sociales*, las herramientas que hoy brindan las redes sociales como: clasificación, comentarios, reseñas, críticas y recomendaciones son el valor agregado que hoy se da a la inteligencia colectiva que actualmente es utilizada por grandes compañías, donde es de vital importancia la participación de las personas en una plataforma virtual de aprendizaje. En el fondo, se trata de generar sistemas atractivos para incentivar la participación de las personas a compartir sus conocimientos.
- Finalmente, en la *memoria organizacional* y *las redes sociales*, la Web 2.0 y las plataformas sociales cumplen con los requisitos de gestión del conocimiento, donde se puede crear contenido, compartirlo y reutilizarlo en los momentos que se necesite, es por esto que la red social ayuda a mejorar la memoria organizacional, aumentando las habilidades sociales y de colaboración humana.

### Capítulo 3:

- En este capítulo se analizaron y especificaron los principales requisitos funcionales y no funcionales de la plataforma de gestión del conocimiento mediante la aplicación de la metodología CommonKADS utilizada para analizar la implementación de este tipo de sistemas de GC, de tal forma que se puedan detectar los problemas y oportunidades, tener en cuenta los aspectos a considerar, realizar una descomposición de los procesos, detectar los activos del conocimiento y realizar un análisis de viabilidad de la construcción de este proyecto.

- También se realizó una analogía general entre las redes sociales y los sistemas de gestión del conocimiento, con el fin de poder conocer y aprovechar las ventajas de las redes sociales y poder aplicar de mejor forma sus características en este proyecto.
- Estos dos análisis permiten realizar un diseño y una construcción de la plataforma de una forma más específica y correcta, con el fin de lograr completar el objetivo general del proyecto de forma satisfactoria.

#### Capítulo 4:

- Para cumplir con los requisitos funcionales y no funcionales especificados, se diseñó la solución bajo una arquitectura web de cliente servidor, en donde se definieron tres servidores: uno de aplicaciones, otro de base de datos y uno de repositorios. En el servidor de aplicaciones se siguieron los patrones de diseño MVC para la construcción de la plataforma, que aportan una facilidad y orden en la construcción.
- Se definieron los componentes y procesos involucrados que cumplen con los requisitos funcionales y no funcionales definidos en el capítulo anterior. Además se definió la utilización de una base de datos primaria no estándar, llamada NoSQL, que contribuye en que la plataforma sea más rápida en el manejo de datos y facilita su construcción.
- Por último se presentó la propuesta de diseño gráfico, que sigue con las mismas tendencias gráficas que se utilizan actualmente en las redes sociales más comunes, para mejorar su uso. También se mantuvo un diseño corporativo en la plataforma.

#### Capítulo 5:

- Para dar cumplimiento a los objetivos del proyecto, se implementaron el framework y las librerías que mejor se ajustaron a los resultados de la investigación realizada, cumpliendo con las características de diseño y arquitectura definidas para la plataforma de gestión del conocimiento en el capítulo anterior y que también resuelven los requisitos funcionales y no funcionales identificados.
- Considerando la complejidad de la plataforma, ésta se dividió en componentes y procesos que se desarrollaron como módulos de la plataforma. Se siguió una metodología incremental en el proceso de desarrollo e implementación que incluyó las siguientes etapas: especificación de requisitos, diseño, codificación, despliegue, pruebas y finalmente puesta en producción, dichas etapas recibieron un seguimiento y fueron controladas en todo el proceso.

- En esta etapa fue importante dar acceso a los usuarios para que pudieran utilizar la plataforma. Se pudo observar cómo actuaban los usuarios en el sistema, se pudo evaluar, corregir errores en el momento y obtener retroalimentación de las personas que trabajaron en ella, siendo éste un punto importante para la llegar a la conclusión de esta tesis, con el fin de apreciar la utilidad que proporciona este proyecto.

## 6.2 Verificación del cumplimiento de objetivos

En esta tesis se desarrollaron todas las etapas propuestas en un principio y para realizar la verificación de los cumplimientos de estos objetivos, se realizó una encuesta con una muestra representativa de usuarios a los que se les dio acceso a la plataforma de gestión del conocimiento (ver tabla 5.10). Esta encuesta tenía como misión obtener una retroalimentación en cuanto al uso que se le dio a la herramienta y en cuanto a la efectividad que tuvo en el cumplimiento de los objetivos para los que fue creada.

La encuesta se basó en tres grupos de preguntas: uso de la plataforma; gestión del conocimiento; y comunicación en el trabajo, que abarcaron los temas más importantes de este trabajo de tesis. En las tres categorías de preguntas se obtuvo una evaluación, con una nota promedio final de 6.4 dentro de una escala de 1.0 a 7.0. El detalle de la evaluación, por cada objetivo específico se muestra en la tabla 6.1.

| Objetivo específico  | Promedio nota |
|--|---------------|
| 1) Gestionar el conocimiento de manera que sea posible crear una memoria organizacional.   | 6.1           |
| 2) Demostrar que las personas puedan participar de manera colaborativa dentro de la plataforma.                                      | 6.4           |
| 3) Determinar los módulos de la plataforma que permitan la creación de conocimiento y sean relevantes para los usuarios.             | 6.3           |
| 4) Considerar una implementación de la plataforma que mejore la comunicación en el trabajo.  | 6.5           |
| 5) Mejorar la usabilidad, navegación, experiencia de usuario y estabilidad en relación a otros sistemas informáticos de la compañía. | 6.4           |

**Tabla 6.1: Promedios por objetivos específicos.**

A través de esta encuesta y de la planificación realizada, se puede validar entonces que se cumple con el objetivo general de “diseñar y crear una plataforma que permita la

gestión del conocimiento basada en una arquitectura colaborativa y con características de red social”.

### 6.3 Contribución de la tesis

La contribución principal de este trabajo de tesis consistió en diseñar e implementar una plataforma de gestión del conocimiento con forma de red social y características de aprendizaje colaborativo que permitirá a la compañía mantener e incrementar su conocimiento de una manera fácil, mejorar la comunicación entre los colaboradores, desde y hacia la compañía, e incentivar la innovación a través de un sistema corporativo y con imagen propia.

Dentro de la encuesta realizada a las personas que utilizaron la plataforma en la marcha blanca, las preguntas mejor evaluadas fueron: “MyKommunity me ayudó a compartir información de interés con mis demás compañeros”; “Considero que este sistema me facilita la cercanía y comunicación con mis compañeros, sobre todo los que están más lejanos”; “Creo que el sistema podría ser un espacio para aprender colaborativamente”; “Me dio confianza utilizar MyKommunity”; “Me gustó el diseño de la plataforma”. De estas preguntas se puede concluir que el uso, la comunicación y la colaboración fueron los puntos más fuertes y reconocidos de esta plataforma.

### 6.4 Limitaciones

Actualmente la plataforma de gestión del conocimiento está desarrollada en una versión limitada, que se utilizó en la etapa de marcha blanca. Esta versión incluye las aplicaciones principales que son las de wiki, gestión documental, preguntas y respuestas y la de grupos. Adicionalmente incluye un buscador, edición de perfil, chat, muro de la plataforma, pero todas estas aplicaciones y módulos están limitados en funcionalidad, ya que sólo se cumplió con las definidas en los requisitos funcionales.

Algunas características del proyecto desarrollado, tales como servidores y sus capacidades o implementación del código, fueron diseñados y utilizados con fines de utilizarlos con usuarios limitados en la fase de marcha blanca, donde no está preparado para utilizarlo en toda la compañía. En caso de requerir más capacidad, esto no está considerado dentro del proyecto de tesis.

Otras funcionalidades de la plataforma, tales como estadísticas, prioridades, sistema gestor de colas de mensaje, que fueron definidas en la etapa de diseño del proyecto, no

fueron implementadas, debido a que requieren más horas de trabajo por el alto desarrollo que demandan.

Otro de las limitaciones de esta plataforma es la privacidad, no está analizada e implementada de forma madura el manejo de la privacidad, para que los usuarios puedan incorporar información delicada, que no necesariamente sea pública.

## 6.5 Trabajo a futuro

Entre las preguntas de la encuesta que se realizaron a las personas que utilizaron la plataforma en la marcha blanca, las preguntas con la evaluación más baja fueron: “Considero que sistema puede servir como un repositorio de información que genere una memoria organizacional para la empresa” y “La plataforma se comportó correctamente, sin errores ni caídas”. De estas preguntas se puede desprender, que como un trabajo futuro está el de buscar una mejor forma, cómo la plataforma puede aportar más en hacer que ésta sirva como un repositorio de memoria organizacional de la compañía, ahí es donde hay que poner más esfuerzo. También se deduce que hay que invertir más tiempo en la validación y aseguramiento de calidad de la plataforma, ya que se generaron algunos errores al momento de la marcha blanca debido al uso masivo, aunque fueron corregidos en el momento.

Los análisis de factibilidad para nuevos requerimientos, tales como aplicaciones, módulos, integraciones, nuevas funcionalidades, pueden ser evaluados a partir de este proyecto de tesis, debido a que es una plataforma abierta que fue diseñada para integrar más servicios y construida de manera que sea evolutiva en el tiempo.

Es importante poder desarrollar a futuro una mejora en la privacidad de la plataforma, de manera de cuidar decisiones importantes que puedan ser tomadas y datos o información delicada que manejen los usuarios dentro de la plataforma, a través de diferentes métodos, como el de mecanismos autónomos, propuesto en el artículo “A Model to Incorporate Privacy in Organizational Memory Systems” [26], o como el de definición de roles, propuesto en el artículo “Temporal Blurring: A Privacy Model for OMS Users” [3].

Por otro lado, en cuanto al tema estudiado, es un tema del que existe abundante documentación, y día a día aparecen más artículos e informes que hablan sobre la evolución de la gestión del conocimiento, aprendizaje colaborativo y redes sociales, por

lo que el tema está abierto para seguir incluyendo nuevos aportes que resuelvan más problemáticas.

# Bibliografía

- [1] Ackerman, M. S., & Halverson, C. (1998, November). Considering an organization's memory. In Proceedings of the 1998 ACM conference on Computer supported cooperative work (pp. 39-48). ACM.
- [2] Akamavi, N., & Kimble, C. (2005, March). Knowledge sharing and computer supported collaborative work: the role of organizational culture and trust. In Proceedings of the 10<sup>th</sup> Annual Conference of the United Kingdom Academy of Information Systems (UKAIS 2005) (p. 22).
- [3] Alarcón, R. A., Guerrero, L. A., & Pino, J. A. (2005). Temporal blurring: a privacy model for OMS users. In User Modeling 2005 (pp. 417-422). Springer Berlin Heidelberg.
- [4] Alexander, B. (2006). Web 2.0: A new wave of innovation for teaching and learning?. *Educause review*, 41(2), 32.
- [5] Allee, V. (1999). Knowledge or Learning. Leverage, <http://www.vernaallee.com/VA/KM-library.htm>.
- [6] Arambarri, J., Estay-Niculcar, C. A., & Villar, S. G. (2007). "La gestión del conocimiento en las organizaciones: la implantación de entornos colaborativos de trabajo". *Telos: Cuadernos de comunicación e innovación*, (72), 32-38.
- [7] Babcock, P. (2004). Five Reasons People Don't Share. *HR Magazine*, 49(5), 47-50.
- [8] Borgatti, S. P., & Cross, R. (2003). A relational view of information seeking and learning in social networks. *Management science*, 49(4), 432-445.
- [9] Brown, J. S., & Duguid, P. (1998). Organizing knowledge. *California management review*, 40(3), 91.
- [10] Carlos Alfonso Lee Blanco, "El capital intelectual y las redes de conocimiento", *INNOTEK-Gestion*, No 4 (2012).
- [11] Catalán, M. A. R., Pérez, R. G., García, O. B., & Sánchez, R. B. (2012). Las comunidades virtuales como potencial pedagógico para el aprendizaje colaborativo a través de las TIC. *Enseñanza & Teaching: Revista interuniversitaria de didáctica*, (30), 105-126.



- 
- [12] Chatti, M.A., Jarke, M., & Frosch-Wilke, D. (2007). The future of e-learning: a shift to knowledge networking and social software. *International journal of knowledge and learning*, 3(4), 404-420.
- [13] Chatti, M.A., Klamma, R., Jarke, M., Kamtsiou, V., Pappa, D., Kravcik, M. and Naeve, A. (2006a) 'Technology enhanced professional learning: process, challenges and requirements', Proceedings of the second International Conference on Web Information Systems and Technologies (WEBIST 2006), 11-13 April, Setubal, Portugal.
- [14] Cho, H., Gay, G., Davidson, B., & Ingraffea, A. (2007). Social networks, communication styles, and learning performance in a CSCL community. *Computers & Education*, 49(2), 309-329.
- [15] Conklin, J. (1996). Designing organizational memory: preserving intellectual assets in a knowledge economy. *Group Decision Support Systems*, 1, 362.
- [16] Cross, J. (2003). Informal Learning—the other 80%. Internet Time Group <http://www.internettime.com/Learning/articles/The%20Other%2080%25.doc>
- [17] Davenport, T. H., & Pruzak, L. (2000). *Working knowledge: How organizations manage what they know*. Harvard Business Press.
- [18] Delmonte, A. J., & Aronson, J. E. (2004). The relationship between social interaction and knowledge management system success. *Journal of Knowledge Management Practice*, 5.
- [19] Drucker, P.F. (1989) *The New Realities: In Government and Politics, in Economics and Business*, in *Society and World View*, 207 Page 14.
- [20] Garrigós, I., Mazón, J. N., Saquete, E., Puchol, M., & Moreda, P. (2010). La influencia de las redes sociales en el aprendizaje colaborativo. *Jornadas de Enseñanza Universitaria de la Informática (16es: 2010: Santiago de Compostela)*.
- [21] Gartner Group (1998) 'Gartner group perspective on knowledge management technologies', <http://www.knowledge-portal.com>
- [22] Ghahfarokhi, A. D., & Zakaria, M. S. (2009, August). Knowledge retention in knowledge management system: Review. In *Electrical Engineering and Informatics, 2009. ICEEI'09. International Conference on (Vol. 2, pp. 343-347)*. IEEE.

- 
- [23] Guerrero, L. A., & Pino, J. A. (2001). Understanding organizational memory. In Computer Science Society, 2001. SCCC'01. Proceedings. XXI International Conference of the Chilean (pp. 124-132). IEEE.
- [24] Gutierrez, F., Baloian, N., & Zurita, G. (2011). Boosting participation in virtual communities. In Collaboration and Technology (pp. 14-29). Springer Berlin Heidelberg.
- [25] Hamidi, S. R., & Jusoff, K. (2009). The characteristic and success factors of an organizational memory information system. *Computer and Information Science*, 2(1), P142.
- [26] Herskovic, V., Ochoa, S. F., & Pino, J. A. (2006, May). A model to incorporate privacy in organizational memory systems. In Computer Supported Cooperative Work in Design, 2006. CSCWD'06. 10th International Conference on (pp. 1-6). IEEE.
- [27] Hildreth, P. M., & Kimble, C. (2002). The duality of knowledge. *Information Research*, 8(1), 8-1.
- [28] Hodgins, H. W. (2000). Into the future a vision paper. In Commission on Technology and Adult Learning. Retrieved from <http://www.Learnativity.Com/download/MP7.PDF> on 6/13/2007.
- [29] Kimble, C., Hildreth, P., & Wright, P. (2001). Communities of practice: going virtual. *Knowledge management and business model innovation*, 220-234.
- [30] Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation*. Cambridge university press.
- [31] Lozano, A. (2005). Comunidades de aprendizaje en red: diseño de un proyecto de entorno colaborativo. *Revista Electrónica de Teoría de la Educación*, 5.
- [32] Lytras, M. D., Naeve, A., & Pouloudi, A. (2005). Knowledge management as a reference theory for e-learning: a conceptual and technological perspective. *International Journal of Distance Education Technologies (IJDET)*, 3(2), 1-12.
- [33] Malhotra, Y. (2005). Integrating knowledge management technologies in organizational business processes: getting real time enterprises to deliver real business performance. *Journal of knowledge management*, 9(1), 7-28.
- [34] Manuel Moreno Castañeda (2005), "Redes de conocimiento en la educación a distancia", *Apertura*. Año/vol.5 número 0001. Universidad de Guadalajara: México. Pp. 6 a 23. Tobón, S.

- 
- [35] McDermott, R. (2001) 'Knowing in communities: 10 critical success factors in building communities of practice', *Community Intelligence Labs*
- [36] Naeve, A. (2005). The human Semantic Web shifting from knowledge push to knowledge pull. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 1(3), 1-30.
- [37] New Media Consortium and EDUCAUSE Learning Initiative. (2007). The horizon report: 2007 edition. Retrieved May, 28, 2007.
- [38] Nonaka, I. (1995). *The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation*. Oxford university press.
- [39] O'Reilly, T. (2005). *What is web 2.0*.
- [40] Ozdemir, S. (2010). To err is human, buy to persist is diabolical: Loss of organizacional memory and e-learning project. *Computer & Education*, 55, p. 101-108, 2010.
- [41] Paavola, S., Lipponen, L., & Hakkarainen, K. (2002, January). Epistemological foundations for CSCL: A comparison of three models of innovative knowledge communities. In *Proceedings of the Conference on Computer Support for Collaborative Learning: Foundations for a CSCL Community* (pp. 24-32). International Society of the Learning Sciences.
- [42] Pollard, D. (2003). The future of knowledge management. How to save the world blog, Discussion paper.
- [43] Rosenberg, M. J. (2005). *Beyond e-learning: Approaches and technologies to enhance organizational knowledge, learning, and performance*. Wiley.com.
- [44] Schreiber, G. (Ed.). (2000). *Knowledge engineering and management: the CommonKADS methodology*. The MIT Press.
- [45] Silvestre Flores de los Santos (2013, January). Las comunidades de aprendizaje como mecanismo eficiente para la formación de formadores en la sociedad del conocimiento. Learning communities as an efficient mechanism for the training of trainers in the knowledge society. In *Congreso Universidad* (Vol. 1, No. 2, p. 10).
- [46] Siemens, G. (2006). *Knowing Knowledge*: Lulu.com.

- [47] Spiller, A., Cunha, C. (2007). Memória Organizacional e Reutilização do Conhecimento Técnico 113mu ma Empresa do Setor Electroelectrônico no Brasil. RBCN, São Paulo, 9, 25, 96-108.
- [48] Wenger, E. (1998). Communities of practice: Learning as a social system. *Systems thinker*, 9(5), 2-3.
- [49] Wheeler, S., YeoMANs, P., & WHEELER, D. (2008). The good, the bad and the wiki: Evaluating student-generated content for collaborative learning. *British journal of educational technology*, 39(6), 987-995.
- [50] Wilson, T. D. (2002). The nonsense of knowledge management. *Information research*, 8(1), 8-1.
- [51] Yates, R. B., & Pino, J. A. (2005). Towards formal evaluation of collaborative work. *Information Research*, 11(4), 12.

# Acrónimos

|         |   |
|---------|---|
| AJAX    | : Asynchronous JavaScript And XML                   |
| API     | : Application Programming Interface                 |
| BSON    | : Binary JSON                                       |
| CMYK    | : Cyan, Magenta, Yellow and Key                     |
| CRM     | : Customer Relationship Management                  |
| CRUD    | : Create, Read, Update and Delete                   |
| CSCCL   | : Computer-Supported Collaborative Learning         |
| CSS     | : Cascading Style Sheets                            |
| DBMS    | : Data Base Management System                       |
| DNS     | : Domain Name System                                |
| FTP     | : File Transfer Protocol                            |
| GC      | : Gestión del Conocimiento                          |
| GUI     | : Graphical User Interface                          |
| HTML    | : Hyper Text Markup Language                        |
| HTTP(S) | : Hypertext Transfer Protocol (Secure)              |
| ISO     | : International Organization for Standardization    |
| IP      | : Internet Protocol                                 |
| IMAP    | : Internet Message Access Protocol                  |
| KM      | : Knowledge Management                              |
| KMS     | : Knowledge Management System                       |
| MIME    | : Multipurpose Internet Mail Extensions             |
| MVC     | : Model-view-controller                             |
| NoSQL   | : Not Only SQL                                      |
| NIC     | : Network Information Center                        |
| IEEE    | : Institute of Electrical and Electronics Engineers |
| J2EE    | : Java 2 Platform Enterprise Edition                |
| JS      | : JavaScript  |
| JSON    | : JavaScript Object Notation                        |
| OM      | : Organizational Model                              |
| OOP     | : Object-oriented programming                       |
| QA      | : Quality Assurance                                 |
| PA      | : Prueba de aceptación                              |
| PF      | : Prueba funcional                                  |
| PHP     | : PHP Hypertext Pre-processor                       |
| RSS     | : Really Simple Syndication                         |
| SGC     | : Sistema de Gestión del Conocimiento               |
| SMTP    | : Simple Mail Transfer Protocol                     |
| SQL     | : Structured Query Language                         |

|      |   |
|------|---|
| SSH  | : Secure Shell                                    |
| TI   | : Tecnologías de la Información                   |
| TIC  | : Tecnologías de la Información y la Comunicación |
| UI   | : User Interface                                  |
| UML  | : Unified Modeling Language                       |
| URI  | : Uniform Resource Identifier                     |
| URL  | : Uniform Resource Locator                        |
| UTF8 | : 8-bit Unicode Transformation Format             |
| UX   | : User Experience                                 |
| WF   | : Wireframe                                       |
| WS   | : Web Services                                    |
| WWW  | : World Wide Web                                  |
| XML  | : eXtensible Markup Language                      |
| 3D   | : Three-dimensional                               |

# Anexos

## Anexo A

| Carpeta       | Descripción   |
|---------------|---|
| Plataforma    | Carpeta base donde se ubica el proyecto.                                |
| _app          | Carpeta donde se ubican las tres capas del modelo MVC.                  |
| __controllers | Carpeta con los archivos de la capa controlador.                        |
| __models      | Carpeta con los archivos de la capa modelo.                             |
| __views       | Carpeta con los archivos de la capa vista.                              |
| _config       | Carpeta que contiene la configuración del framework.                    |
| _lib          | Carpeta que contiene los archivos base del framework y otras librerías. |
| __db          | Carpeta donde se ubican los archivos de base de datos del framework.    |
| ___mongo      | Carpeta con los archivos de MongoDB del framework.                      |
| ___sql        | Carpeta con los archivos de MySQL del framework.                        |
| _tmp          | Carpeta que gestiona el framework para administrar el caché.            |
| _ui           | Carpeta con archivos y librerías externas de la capa de vista.          |
| __css         | Carpeta que contiene los archivos CSS del proyecto.                     |
| ___img        | Carpeta con imágenes requeridas por los CSS.                            |
| __img         | Carpeta con las imágenes del proyecto.                                  |
| __js          | Carpeta con archivos JavaScript requeridos.                             |

Tabla 7.1: Tabla con la especificación de la estructura de carpetas del proyecto.

## Anexo B

| Contenido del archivo index.php                       |
|---|
| <pre>&lt;?php \$main = require('lib/base.php');</pre> |

```

$main->set('ONERROR',
    function($main) {
        echo '<p>'.$main->get('ERROR.code').'</p>';
        echo '<p>'.$main->get('ERROR.status').'</p>';
        echo '<p>'.$main->get('ERROR.title').'</p>';
        echo '<p>'.$main->get('ERROR.text').'</p>';
    }
);
$main->set('DEBUG',3);
$main->config('config/setup.cfg');
$main->set('ENCODING','UTF-8');

$main->route('GET /','KM->dashboard');
$main->route('GET /dashboard','KM->dashboard');
$main->route('GET /profile','KM->profile');
$main->route('GET /profile/edit','KM->profile_edit');
$main->route('POST /profile/edit','KM->profile_save');
$main->route('GET /profile/@username','KM->profile');
$main->route('GET /profile/@username/add','KM->profile_add');
$main->route('GET /profile/@username/remove','KM->profile_remove');
$main->route('GET /search','KM->search');
$main->route('GET /search/@query','KM->search');
$main->route('GET /login','KM->login');
$main->route('POST /login','KM->auth');
$main->route('GET /logout','KM->logout');
$main->route('GET /about','KM->about');
$main->route('GET /image','Images->image');
$main->route('GET /image/@type','Images->get');
$main->route('GET /image/@type/@id','Images->get');
$main->route('GET /document/@id/@name','Documents->get');
$main->route('POST /post/gallery','KM->gallery_save');
$main->route('POST /post/document','KM->document_save');
$main->route('POST /post/wall','KM->wall_save');
$main->route('POST /post/qa','KM->qa_save');
$main->route('POST /post/sensibility','KM->sensibility_save');
$main->route('POST /post/location','KM->location_save');
$main->route('POST /post/wiki','KM->wiki_new');
$main->route('GET /post/sensibility/@type/@id','KM-
>sensibility_save');
$main->route('GET /post/sensibility/@type/@id/@data','KM-
>sensibility_save');
$main->route('GET /post/share/@type/@id','KM->share');
$main->route('GET /remove/@type/@id','KM->remove');
$main->route('GET /settings','KM->settings');
$main->route('POST /settings/password','KM->change_password');
$main->route('GET /policies','KM->policies');
$main->route('GET /invitations','KM->invitations');
$main->route('POST /invitations/invite','KM->invite');
$main->route('GET /activate/@id','KM->activate');
$main->route('POST /activate','KM->activate_op');
$main->route('GET /question/@id','KM->question');
$main->route('GET /question/@id/@url','KM->question');
$main->route('GET /answer/@id','KM->answer');

```



```

$main->route('GET /sos','Sos->sos');
$main->route('GET /chat','Chat->display');
$main->route('GET /chat/user/@user','Chat->display');
$main->route('POST /chat/save','Chat->save');
$main->route('GET /lostPassword','KM->lost_password');
$main->route('POST /lostPassword','KM->send_password');
$main->route('GET /resetPassword/@id/@code','KM->reset_password');
$main->route('POST /resetPassword','KM->reset_password_op');
$main->route('GET /wiki','KM->wiki');
$main->route('GET /wiki/@query','KM->wiki_detail');
$main->route('GET /wiki/@query/edit','KM->wiki_edit');
$main->route('POST /wiki/@query/edit','KM->wiki_save');
$main->route('GET /wiki/@query/history','KM->wiki_history');
$main->route('GET /wiki/@query/@id','KM->wiki_detail');
$main->route('GET /locations','KM->locations');

$main->run();

```

Tabla 7.2: Contenido y enrutamiento del archivo index.php.

## Anexo C

### Contenido del archivo KM.php

```

<?php
require('lib/class.phpmailer.php');
class KM {

    public $posts, $view, $layout, $db1, $db2, $km, $isAuth;
    function __construct() {
        $this->km = Base::instance();
        $this->db1 = new DB\Mongo($this->km->get('db_dns1'), $this->km->get('db_name1'));
        $this->db2 = new DB\SQL($this->km->get('db_dns2').$this->km->get('db_name2'), $this->km->get('db_user2'), $this->km->get('db_pass2'));
        $this->mimetype = 'text/html';
        $this->isAuth = $this->km->exists('SESSION.user') && $this->km->get('SESSION.user') != '';

        if($this->km->get('AJAX')) {
            $this->layout = 'blank.html';
        } else if($this->isAuth) {
            $this->layout = 'layout.html';
        } else {
            $this->layout = 'login.html';
        }
    }
    function dashboard() {}
    function about() {

```

```

$this->km->set('pagetitle','about');
if($this->isAuth) {
    $this->km->set('template','about.html');
    $this->km->set('onLoad','$(\'#columns\').isotope({ filter:
\\\'\' });});');
} else {
    $this->layout = 'about2.html';
}
}

function settings() {}
function invitations() {}
function invite() {}
function activate() {}
function activate_op() {}
function policies() {}
function profile() {}
function profile_edit() {}
function profile_save() {}
function profile_add() {}
function profile_remove() {}
function gallery_save() {}
function document_save() {}
function wall_save() {}
function qa_save() {}
function location_save() {}
function wiki_new() {}
function sensibility_save() {}
function share() {}
function search() {}
function question() {}
function answer() {}
function login() {}
function auth() {}
function logout() {}
function remove_all() {}
function remove() {}
function notification() {}
function lost_password() {}
function send_password() {}
function reset_password() {}
function reset_password_op() {}
function change_password() {}
function wiki() {}
function wiki_detail() {}
function wiki_edit() {}
function wiki_save() {}
function wiki_history() {}
function locations() {}
function beforeroute() {}
function afterroute() {}
}

```

**Tabla 7.3: Contenido del controlador KM.php.**

## Anexo D

| Contenido del archivo Chat.php  |
|---|
| <pre>&lt;?php class Chat {      public \$posts, \$view, \$mimetype, \$layout, \$db1, \$chat, \$isAuth;      function __construct() {         \$this-&gt;chat = Base::instance();          \$this-&gt;isAuth = \$this-&gt;chat-&gt;exists('SESSION.user') &amp;&amp; \$this-&gt;chat-&gt;get('SESSION.user') != '';         \$this-&gt;chat-&gt;set('user', \$this-&gt;chat-&gt;get('SESSION.user'));          \$this-&gt;layout = 'chat.html';         \$this-&gt;mimetype = 'text/html';     }      function display() {}      function save() {}      function afterroute() {}  }</pre> |

Tabla 7.4: Contenido del controlador Chat.php.

## Anexo E

| Contenido del archivo Documents.php   |
|---|
| <pre>&lt;?php class Documents {      public \$id;      function __construct() {         \$this-&gt;documents = Base::instance();          \$this-&gt;id = \$this-&gt;documents-&gt;get('PARAMS.id');         \$this-&gt;name = \$this-&gt;documents-&gt;get('PARAMS.name');     }      function get() {}  }</pre> |

```
function save() {}
}
```

Tabla 7.5: Contenido del controlador Documents.php.

## Anexo F

| Contenido del archivo Images.php  |
|---|
| <pre>&lt;?php class Images {      public \$conf, \$type, \$id, \$format;      function __construct() {         \$this-&gt;images = Base::instance();         \$this-&gt;conf = array();         \$this-&gt;type = \$this-&gt;images-&gt;get('PARAMS.type');         \$this-&gt;id = \$this-&gt;images-&gt;get('PARAMS.id');     }      function get() {}     function save() {} }</pre> |

Tabla 7.6: Contenido del controlador Images.php.

## Anexo G

| Contenido del archivo Activities.php   |
|--|
| <pre>&lt;?php class Activities extends DB\Mongo\Mapper {      public function __construct(DB\Mongo \$db) {         parent::__construct(\$db, 'activities');     }      public function all() {         \$this-&gt;load();         return \$this-&gt;query;     }      public function add() {         \$this-&gt;copyFrom('POST');     } }</pre> |

```

        $this->save();
    }

    public function getById($id) {
        $this->load(array('id=?', $id));
        $this->copyTo('POST');
    }

    public function edit($id) {
        $this->load(array('id=?', $id));
        $this->copyFrom('POST');
        $this->update();
    }

    public function delete($id) {
        $this->load(array('id=?', $id));
        $this->erase();
    }
}

```

**Tabla 7.7: Contenido del modelo Activities.php.**

| Contenido del archivo Entries.php  |
|--|
| <pre> &lt;?php class Entries extends DB\Mongo\Mapper {     public function __construct(DB\Mongo \$db) {}     public function all() {}     public function add() {}     public function getById(\$id) {}     public function edit(\$id) {}     public function delete(\$id) {} } </pre> |

**Tabla 7.8: Contenido del modelo Entries.php.**

| Contenido del archivo Galleries.php  |
|--|
| <pre> &lt;?php class Galleries extends DB\Mongo\Mapper {     public function __construct(DB\Mongo \$db) {}     public function all() {}     public function add() {}     public function getById(\$id) {}     public function edit(\$id) {}     public function delete(\$id) {} } </pre> |

**Tabla 7.9: Contenido del modelo Galleries.php.**

| Contenido del archivo Groups.php  |
|---|
| <pre> &lt;?php class Groups extends DB\Mongo\Mapper {     public function __construct(DB\Mongo \$db) {}     public function all() {}     public function add() {}     public function getById(\$id) {}     public function edit(\$id) {}     public function delete(\$id) {} } </pre> |

Tabla 7.10: Contenido del modelo Groups.php.

| Contenido del archivo Logs.php  |
|---|
| <pre> &lt;?php class Logs extends DB\Mongo\Mapper {     public function __construct(DB\Mongo \$db) {}     public function all() {}     public function add() {}     public function getById(\$id) {}     public function edit(\$id) {}     public function delete(\$id) {} } </pre> |

Tabla 7.11: Contenido del modelo Logs.php.

| Contenido del archivo Messages.php  |
|---|
| <pre> &lt;?php class Messages extends DB\Mongo\Mapper {     public function __construct(DB\Mongo \$db) {}     public function all() {}     public function add() {}     public function getById(\$id) {}     public function edit(\$id) {}     public function delete(\$id) {} } </pre> |

Tabla 7.12: Contenido del modelo Messages.php.

| Contenido del archivo Sensibilities.php  |
|--|
| <pre> &lt;?php class Sensibilities extends DB\Mongo\Mapper {     public function __construct(DB\Mongo \$db) {}     public function all() {} } </pre> |

```

public function add() {}
public function getById($id) {}
public function edit($id) {}
public function delete($id) {}
}

```

**Tabla 7.13: Contenido del modelo Sensibilities.php.**

| Contenido del archivo Users.php  |
|--|
| <pre> &lt;?php class Users extends DB\Mongo\Mapper {     public function __construct(DB\Mongo \$db) {}     public function all() {}     public function add() {}     public function getById(\$id) {}     public function edit(\$id) {}     public function delete(\$id) {} } </pre> |

**Tabla 7.14: Contenido del modelo Users.php.**

## Anexo H

| Archivo                           | Contenido   |
|-----------------------------------|---|
| /etc/bind/named.conf.local        | <pre> zone "mykommunity.cl" {     type master;     file "/var/cache/bind/db.mykommunity.cl"; };  zone "254.114.190.in-addr.arpa" {     type master;     file "/var/cache/bind/db.190.114.254"; }; </pre>  |
| /var/cache/bind/db.mykommunity.cl | <pre> \$TTL 1800 @ IN SOA km.mykommunity.cl admin.mykommunity.cl. (     1     1800     3600     604800     86400 ) @ IN NS km.mykommunity.cl.  mykommunity.cl. IN A 190.114.254.196 km IN A 190.114.254.196 www IN CNAME mykommunity.cl. </pre> |

|                                    |  |
|------------------------------------|--|
| /var/cache/bind/<br>db.190.114.254 | <pre> \$TTL 86400 254.114.190.in-addr.arpa. IN SOA km.mykommunity.cl. admin.mykommunity.cl. (     1     3h     1h     1w     1h )  254.114.190.in-addr.arpa. IN NS km.mykommunity.cl.  196.254.114.190.in-addr.arpa. IN PTR mykommunity.cl. </pre> |
|------------------------------------|--|

Tabla 7.15: Detalle de la configuración del servidor DNS.

## Anexo I

| Contenido de la configuración del sitio   |
|---|
| <pre> &lt;VirtualHost *:80&gt;     ServerAdmin webmaster@localhost     ServerName mykommunity.cl     ServerAlias www.mykommunity.cl      DocumentRoot /var/www/mykommunity.cl     &lt;Directory /&gt;         Options FollowSymLinks         AllowOverride All     &lt;/Directory&gt;     &lt;Directory /var/www/mykommunity.cl/&gt;         Options Indexes FollowSymLinks MultiViews         AllowOverride All         Order allow,deny         allow from all     &lt;/Directory&gt;      ErrorLog \${APACHE_LOG_DIR}/error.log     LogLevel warn     CustomLog \${APACHE_LOG_DIR}/access.log combined &lt;/VirtualHost&gt; </pre> |

Tabla 7.16: Detalle de la configuración del sitio web.



## Anexo J

| Contenido de la configuración del servidor MySQL |                               |
|--|-------------------------------|
| <b>[client]</b>                                  |                               |
| port   | = 3306                        |
| socket   | = /var/run/mysqld/mysqld.sock |
| <b>[mysqld_safe]</b>                             |                               |
| socket   | = /var/run/mysqld/mysqld.sock |
| nice   | = 0                           |
| <b>[mysqld]</b>                                  |                               |
| user   | = mysql                       |
| pid-file   | = /var/run/mysqld/mysqld.pid  |
| socket   | = /var/run/mysqld/mysqld.sock |
| port   | = 3306                        |
| basedir  | = /usr                        |
| datadir  | = /var/lib/mysql              |
| tmpdir   | = /tmp                        |
| lc-messages-dir                                  | = /usr/share/mysql            |
| skip-external-locking                            |                               |
| key_buffer                                       | = 16M                         |
| max_allowed_packet                               | = 16M                         |
| thread_stack                                     | = 192K                        |
| thread_cache_size                                | = 8                           |
| myisam-recover                                   | = BACKUP                      |
| query_cache_limit                                | = 1M                          |
| query_cache_size                                 | = 16M                         |
| log_error  | = /var/log/mysql/error.log    |

Tabla 7.17: Detalle de la configuración del servidor MySQL.

## Anexo K

| Carpeta       | Contenido de la carpeta   |
|---------------|---|
| plataforma    | .htaccess, favicon.ico, index.php, moadmin.php, phpinfo.php   |
| _app          |   |
| __controllers | Chat.php, Documents.php, Images.php, KM.php   |
| __models      | Activities.php, Entries.php, Galleries.php, Groups.php, Logs.php, Messages.php, Sensibilities.php, Users.php  |
| __views       | about.html, about2.html, activate.html, blank.html, chat.html, dashboard.html, invitations.html, js-dashboard.html, js-profile.html, js-search.html, js-wiki.html, layout.html, layout2.html, locations.html, |

|          |  |
|----------|--|
|          | login.html, lostPassword.html, policies.html, profile.html, profile_edit.html, question.html, search.html, settings.html, wiki.html, wiki_detail.html, wiki_edit.html, wiki_history.html   |
| __config | setup.cfg  |
| __lib    | base.php, class.phpmailer.php, magic.php, template.php, tools.php  |
| __db     | cursor.php, mongo.php, sql.php   |
| ___mongo | mapper.php, session.php  |
| ___sql   | mapper.php, session.php  |
| __tmp    |  |
| __ui     |  |
| __css    | dashboard.css, jquery-ui-1.10.4.custom.css, lightbox.css, style.css  |
| ___img   | animated-overlay.gif, boxBottomLeftCorner.png, boxBottomRightCorner.png, boxSideWallPx.gif, boxSideWallPx.png, boxTopLeftCorner.png, boxTopPx.gif, boxTopPx.png, boxTopRightCorner.png, closeButton_down.gif, closeButton_normal.gif, closeButton_over.gif, icons.gif, monthBackward_down.gif, monthBackward_normal.gif, monthBackward_over.gif, monthForward_down.gif, monthForward_normal.gif, monthForward_over.gif, orange_dayDown.gif, orange_dayNormal.gif, orange_dayOver.gif, ui-bg_flat_0_aaaaaa_40x100.png, ui-bg_glass_65_ffffff_1x400.png, ui-bg_glass_75_dadada_1x400.png, ui-bg_glass_75_e6e6e6_1x400.png, ui-bg_glass_95_fef1ec_1x400.png, ui-bg_highlight-soft_75_cccccc_1x100.png, ui-icons_222222_256x240.png, ui-icons_2e83ff_256x240.png, ui-icons_454545_256x240.png, ui-icons_888888_256x240.png, ui-icons_cd0a0a_256x240.png, yearBackward_down.gif, yearBackward_normal.gif, yearBackward_over.gif, yearForward_down.gif, yearForward_normal.gif, yearForward_over.gif |
| ___img   | 1place.png, 2place.png, 3place.png, ads.png, albums.png, alert, alerta.png, apps, apps.png, birthday.png, blog.png, buzon.png, calendar.png, calendario.png, camara.png, cantidad.png, cartilla.png, chat, chat.png, chaticon.png, close.png, cloud.png, compartir.png, configuration.png, deloitte.jpg, dhl.jpg, documento.png, documents.png, edit.png, emails.png, etiquetar.png, fedex.jpg, grafico.png, groups.png, home.png, home.svg, ideas.png, invitation.gif, invitation.png, like-not.png, like.png, like2.png, loading.gif,  |

|     |  |
|-----|--|
|     | locations.png, logo, logo.png, lupa.png, mail, mcafee.jpg, megusta.png, megusta2.png, muro.png, next.png, perfil.png, pin.png, policie.png, polls.png, preguntas.png, prev.png, profile.png, puzzle.png, pwc.png, questions.png, send.png, settings.png, toshiba.png, triangle.png, tyco.png, ubicacion.png, video.png, videos.png, wall, wiki.png, workflow.png, worldcup.png, x.png, xerox.jpg |
| —js | dashboard.js, htmlparser.js, jquery-1.10.2.min.js, jquery-ui-1.10.4.custom.js, jquery.isotope.min.js, jquery.tagcanvas.min.js, jquery.tagcloud.js, jquery.timeago.js, jquery.timer.js, jsDatePick.min.1.3.js, lightbox-2.6.min.js, modernizr.custom.js, zaria.js   |

**Tabla 7.18: Contenido de las carpetas subidas por FTP.**

## Anexo L

| Contenido del archivo de configuración de la plataforma   |
|---|
| <pre>[globals] root_url=/ img_path=/var/www/mykommunity.cl/img AUTOLOAD=app/controllers/ app/models/ DEBUG=3 UI=app/views/ db_dns1=mongodb://localhost:27017 db_name1=km db_user1=root db_pass1=45756 db_dns2=mysql:host=localhost;port=3306;dbname= db_name2=km db_user2=root db_pass2=87976</pre> |

**Tabla 7.19: Detalle del archivo de configuración de la plataforma.**

| Contenido del archivo .htaccess   |
|---|
| <pre>RewriteEngine On RewriteCond %{ENV:REDIRECT_STATUS} ^\$ RewriteCond %{REQUEST_FILENAME} !-l RewriteCond %{REQUEST_FILENAME} !-f RewriteCond %{REQUEST_FILENAME} !-d RewriteRule .* index.php [L,QSA]</pre> |

**Tabla 7.20: Contenido del archivo .htaccess.**