



**UNIVERSIDAD DE TALCA
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA CIVIL EN COMPUTACIÓN**

**Desarrollo de sistema de gestión de proyectos con
inclusión de clientes**

IVÁN MARTÍNEZ

Profesor Guía: MATTHEW BARDEEN

Memoria para optar al título de
Ingeniero Civil en Computación

Curicó – Chile
julio, 2018

CONSTANCIA

La Dirección del Sistema de Bibliotecas a través de su encargado Biblioteca Campus Curicó certifica que el autor del siguiente trabajo de titulación ha firmado su autorización para la reproducción en forma total o parcial e ilimitada del mismo.



Two circular official stamps and handwritten signatures in blue ink. The left stamp is from the 'DIRECCIÓN SISTEMA DE BIBLIOTECAS' of the 'UNIVERSIDAD DE TALCA'. The right stamp is from the 'SISTEMA DE BIBLIOTECAS CAMPUS CURICO' of the 'UNIVERSIDAD DE TALCA'.

Curicó, 2022

*“Sólo podemos ver poco del futuro, pero lo suficiente para darnos cuenta de que hay
mucho que hacer”
Alan Turing*

AGRADECIMIENTOS

Agradezco a mi familia por acompañarme en todos los momentos importantes de mi vida.

A mis padres, Ivonne y Eliecer, por darme la capacidad de superar los obstáculos propios de la vida y la perseverancia para alcanzar mis objetivos.

A mi profesor guía Matthew Bardeen, por sus consejos y correcciones. A mis profesores y a todos quienes estos años aportaron en mi crecimiento personal, con conocimientos, apoyo, amistad y formaron parte de esta etapa.

TABLA DE CONTENIDOS

	página
Dedicatoria	I
Agradecimientos	II
Tabla de Contenidos	III
Índice de Figuras	V
Índice de Tablas	VII
Resumen	VIII
1. Introducción	9
1.1. Definición del problema	10
1.2. Propuesta de solución	11
1.3. Objetivo General	12
1.4. Objetivos Específicos	13
1.5. Alcances	13
1.6. Descripción de los contenidos	13
2. Marco teórico	15
2.1. Tecnologías y herramientas	18
2.2. Software de planificación y gestión de proyectos	19
2.3. Integración del Stakeholder	20
2.4. Trabajo relacionado	21
3. Metodología	23
3.1. Metodología de desarrollo	23
3.2. Evaluación del sistema	24
4. Implementación de la Solución	28
4.1. Concepción	29
4.2. Planificación	30

4.3. Sistema	32
4.4. Base de datos	41
4.5. Interfaz	43
5. Evaluación	53
5.1. Análisis de perfil de usuario	53
5.2. Definición de los objetivos de usabilidad	54
5.3. Implementación de prototipos	55
5.4. Realización de pruebas	56
5.5. Rediseño	57
5.6. Implementación del producto o servicio	60
5.7. Retroalimentación del usuario	60
5.8. Aplicación	62
5.9. Resultados	63
5.10. Objetivos	67
6. Conclusiones	68
6.1. Trabajo futuro	69
Glosario	70
Bibliografía	72
Anexos	
A: Historias de usuarios	74
A.1. Listado funcionalidades	75
A.2. Prioridad final de construcción con pesos	77
B: Encuestas	78
B.1. preguntas	78
C: Resultados evaluación	81
D: Guía rápida	83
E: Flujo Metodología Desarrollo	98

ÍNDICE DE FIGURAS

	página
1.1. Flujo de especificación	9
1.2. Flujo de especificación y construcción con ProjectMNG	12
2.1. Factores que afectan la escalabilidad de un proyecto [7]	16
3.1. Diagrama de flujo de evaluación	25
3.2. Diagrama de flujo de evaluación modificado	26
4.1. Primer Diagrama UML	29
4.2. Primer Diagrama de tablas	29
4.3. planificación simple	30
4.4. Arquitectura del sistema	33
4.5. UML del sistema	33
4.6. Objeto Project	34
4.7. Objeto Team	35
4.8. Objeto Characteristic	36
4.9. Objeto Module	37
4.10. Objeto Activity	37
4.11. Enum RequirementState	39
4.12. Objeto File	40
4.13. Modelo tablas	41
4.14. Tabla Characteristic	42
4.15. Tabla ChangeRequest	42
4.16. Tabla ChangerequestLog	43
4.17. Listado de Proyectos	43
4.18. Detalle de Proyecto	44
4.19. Características por Modulo de un Proyecto	45
4.20. Estados Característica	46
4.21. Características	47
4.22. Proyecto para Stakeholders	48
4.23. Detalle Característica	49
4.24. Detalle Característica sin permisos	50

4.25. Solicitud de Cambio	51
4.26. Línea de tiempo con cambios realizados en un proyecto	52
5.1. Flujo de trabajo	56
5.2. Listado de características luego del rediseño	57
5.3. Visualización móvil de las características	58
5.4. Visualización móvil de las características	59
5.5. Alerta de éxito al crear una actividad	60
5.6. Resultados de satisfacción de la plataforma a nivel funcional	61
5.7. Resultados de satisfacción de la plataforma a nivel interfaz	62
5.8. Flujo de trabajo	64
5.9. Resultados de satisfacción de la plataforma a nivel funcional	65
5.10. Resultados de satisfacción de la plataforma a nivel interfaz	65
5.11. Listado de características visto desde un móvil	66
A.1. Prioridad final de construcción con pesos	77

ÍNDICE DE TABLAS

	página
C.1. Detalle de resultados de la encuesta de evaluación	82

RESUMEN

Agilizar lo más posible la especificación, diseño y construcción de proyectos es un objetivo común para todos aquellos dedicados a esto.

Lograr finalizar un proyecto en donde el cada participante quede conforme con el resultado produciendo un sentimiento de pertenencia es una situación soñada para cualquier proyecto.

Al tratar de lograr todo lo anterior descrito nos encontramos un gran problema, tratar de compatibilizar tiempos, distintas maneras/canales de comunicación, teniendo siempre en consideración la participación de los distintos actores de un proyecto.

Para lograr controlar los problemas ya mencionados, se diseñó y se construyó un sistema web, que mediante una solución propia para la especificación y planificación de proyectos es capaz de crear un canal de comunicación para participantes de un proyecto que no necesariamente son desarrolladores, permitiéndoles participar en este de una forma controlada sin interferir de manera directa en este.

Para la evaluación del sistema web se realizan encuestas a usuarios del sistema, estos usuarios son seleccionados por sus aptitudes maximizando la calidad de la información.

La información recopilada por las encuestas se analizó para luego realizar los cambios de acuerdo a los resultados obtenidos.

Fruto de este trabajo de diseño, construcción y evaluación se tendrá un software, que permite gestionar proyectos y proveer un flujo de comunicación entre sus participantes en conjunto con un gestor de control de cambios.

1. Introducción

Dentro del desarrollo de proyectos existe una contraparte, que permiten dar comienzo y posteriormente un fin, concretando en un producto. Están aquellos encargados de planificar y llevar a cabo el proyecto, que por simplicidad llamaremos *Desarrolladores* y luego, la contraparte, quienes desean este proyecto o producto, a quienes denominamos como *stakeholders*.

En un proceso típico de desarrollo ágil hay una evolución iterativa del proyecto, en donde desarrolladores y stakeholders interactúan de manera constante con el fin de llegar a un producto que cumpla con las expectativas. Esta comunicación se realiza a lo largo de todo el proceso de desarrollo.

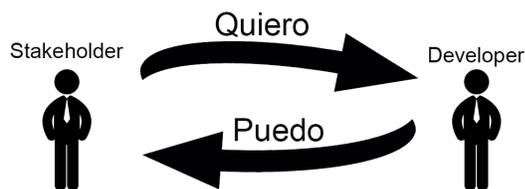


Figura 1.1: Flujo de especificación

La Figura 1.1 ilustra la comunicación directa y bilateral entre desarrolladores y stakeholders, esta comunicación se representa en forma de flechas que van de uno hacia el otro. Esta comunicación es independiente del medio que se utilice, ya sea esta por medio de chats o servicios de streaming.

El proyecto se define dentro de la comunicación entre desarrolladores y stakeholders, generando una planificación, que posteriormente se llevará a cabo. La comunicación tiene un gran protagonismo a lo largo del proceso planificación y desarrollo.

1.1. Definición del problema

La planificación de un proyecto puede ser abordada de diversas maneras, una de las más comunes hoy en día es por medio de distintos software.

Existe una gran cantidad de software dedicados a la planificación de proyectos, la mayoría de estos muy efectivos, incluso poseen herramientas que permiten verificar el proceso, pero en ningún caso permiten conectar esta verificación, directamente con stakeholders, limitando este proceso solo a los desarrolladores.

Desde esta problemática nace la necesidad de crear un canal de comunicación, que sea capaz de pasar a través de la planificación de un proyecto de software, conectando a desarrolladores y stakeholders de manera controlada, pero permitiendo la colaboración activa y efectiva de ambas partes.

Actualmente la problemática de la verificación está estrechamente relacionada con la comunicación, esto puede traer una serie de problemas dentro de un desarrollo, como lo aclara Mark Keil, en su artículo *Pulling the Plug: Software Project Management and the Problem of Project Escalation* [7], donde se indican los principales factores que afectan de manera directa la escalabilidad de un proyecto

ProjectMNG, es creado para trabajar como intermediario entre desarrolladores y stakeholders, dándoles una herramienta de especificación, planificación, verificación y control de cambios que buscan solucionar los problemas que nacen en cada uno de estos aspectos, entregando una plataforma que estará disponible a cualquier hora para la disposición de todos los participantes en un proyecto.

1.2. Propuesta de solución

ProjectMNG simplifica de visualizar proyectos y entrega una manera de participar activamente dentro de desarrollos, creando y verificando sus proyectos; permitiendo proponer cambios, visualizar avances y gestionar cambios en tiempo real.

ProjectMNG, logra una integración desarrolladores–stakeholders, sin dejar de lado el proceso de especificación, planificación y desarrollo, entregando, por medio de una plataforma web, herramientas simples que permiten crear y gestionar proyectos.

La especificación de proyectos se basa en la creación de *Características*, las cuales son definiciones atómicas de secciones de un proyecto.

Las características pasan a través de distintos *Estados* a lo largo del proyecto, permitiendo migrar esta especificación a una fase de planificación y luego a una fase de desarrollo dentro del mismo flujo de trabajo.

El sistema de comunicación desarrollador–stakeholder se obtiene por medio de una *participación acotada*, esto hace referencia a que el sistema proveerá las herramientas que permitan la participación, forzando, de esta manera, un flujo de trabajo que permita la participación interna de los stakeholders en un proyecto sin el riesgo de que entorpezca el proceso.

Una herramienta que provee el sistema para la participación de los stakeholders es la *solicitud de cambio*; esta funcionalidad, como bien lo aclara su nombre, permite realizar solicitudes de cambio sobre las características, estas son generadas ante cualquier modificación realizada.

Las solicitudes de cambio en el sistema deben ser aprobadas por la contraparte respectiva.

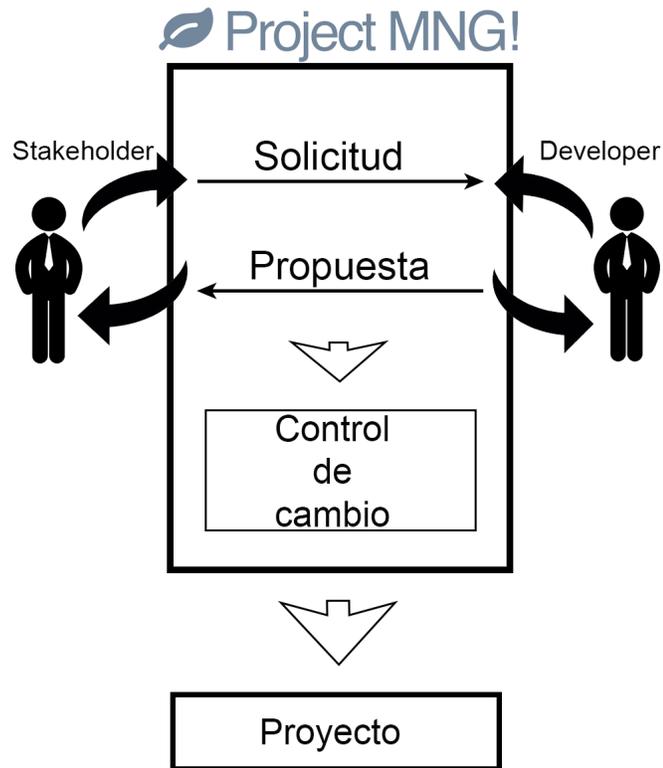


Figura 1.2: Flujo de especificación y construcción con ProjectMNG

En la Figura 1.2 se ilustra como ProjectMNG actúa como agente mediador entre desarrolladores y stakeholders, siendo representado como la caja que está en medio de los dos sujetos, permitiendo una participación activa, pero dentro de parámetros establecidos, que facilita la gestión de cambios y las hace parte del proyecto.

1.3. Objetivo General

Proporcionar una herramienta que permita especificar, verificar y planificar proyectos de software en conjunto con los stakeholders.

1.4. Objetivos Específicos

- Crear un canal de comunicación entre desarrolladores y stakeholder.
- Crear una herramienta simple de especificación y gestión de proyectos.
- Otorgar herramientas para que los stakeholders participe en su proyecto.
- Crear un sistema web capaz de cumplir con los objetivos mencionados anteriormente.

1.5. Alcances

- Construir un sistema de gestión de proyectos, implementando un sistema de especificación de características, un sistema de gestión de tareas asociado y un sistema de gestión de usuarios (Desarrolladores–stakeholders).
- Las características creadas en este sistema, no obedecen a ninguna estructura ni modelo existente, sino que buscan ser un complemento útil en la práctica del desarrollo de proyectos.
- La integración del stakeholder en el desarrollo de un proyecto, abarca la visualización del estado de un proyecto, la capacidad de interferir en la especificación y la definición de características por medio de una interfaz.

1.6. Descripción de los contenidos

Este documento, detalla específicamente el proceso completo de diseño, construcción y evaluación del sistema web ProjectMNG.

Capítulo 2: En este capítulo, se consideran los motivos por los cuales surgen problemas durante la fase de desarrollo y cómo darles solución, explicando como funciona un software de planificación y gestión de proyectos, la integración de los stakeholders a un proyecto y a través de qué herramientas y tecnologías será construida la solución.

Capítulo 3: En este capítulo, se describen los procesos y metodologías para el diseño, construcción y evaluación del sistema ProjectMNG.

Capítulo 4: En este capítulo, se detalla el diseño de la solución, como sistema y como interfaz de usuario.

Capítulo 5: En este capítulo, se muestran los resultados obtenidos de la evaluación y cómo estos fueron utilizados para mejorar el sistema.

Capítulo 6: En este capítulo, se mencionan las conclusiones obtenidas, así como también, detalles de la evaluación del sistema y brevemente sobre el trabajo futuro.

2. Marco teórico

La ingeniería de software [9] consta de seis etapas: análisis, diseño, arquitectura, construcción, pruebas y mantenimiento.

De estas seis etapas, la sexta etapa (mantenimiento) es excluida, debido a que Project-MNG no ha sido pensado para llegar hasta ese punto.

Dentro de cada una de las etapas de la ingeniería del software, se llevan a cabo las acciones de: *Planificación* y *Verificación*, que permiten llegar al objetivo deseado de forma controlada si estas son bien utilizadas.

La Planificación consiste en establecer una línea a seguir, determinando productos, hitos y plazos, los cuales deben ser seguidos para lograr la finalización del proyecto, dentro de los márgenes establecidos.

La Verificación hace referencia a la aceptación de una característica, requerimiento o idea, la cual define y/o delimita el proyecto, estos requerimientos se encuentran presentes en cada etapa del desarrollo.

Las acciones de planificación y verificación, se convierten en algo fundamental dentro de un proyecto o desarrollo, acotando tiempos, alcances y recursos cuando son bien ejecutados.

La verificación está estrechamente relacionada con la comunicación, lo que puede arrastrar una serie de problemas, incluso, hasta el desarrollo, como menciona Mark Keil, en su artículo *Pulling the Plug: Software Project Management and the Problem of Project Escalation* [7], indicando los principales factores que afectan de manera directa la escalabilidad de un proyecto

Los factores del proyecto y organizacionales, tienen directa relación con los desarrolladores del proyecto, pero, tanto factores sociales como psicológicos, pese a tener consecuencias en los desarrolladores, tienen una mayor presencia en los stakeholders.

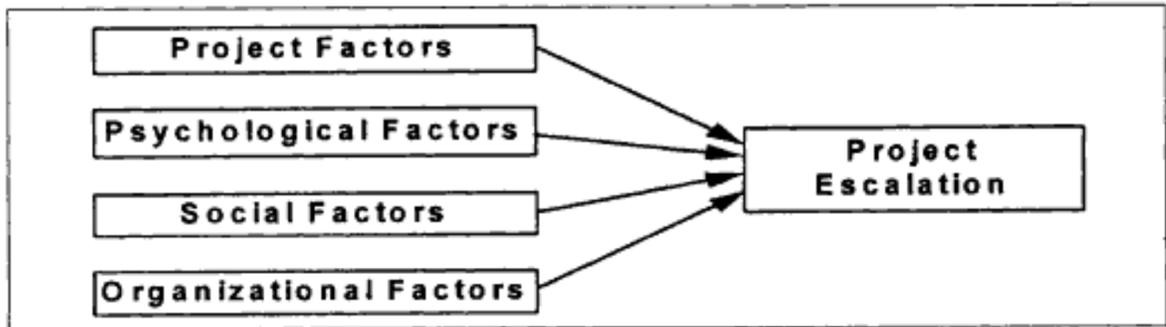


Figure 1. Model of Project Escalation

Figura 2.1: Factores que afectan la escalabilidad de un proyecto [7]

Un claro ejemplo de las repercusiones de factores sociales en el desarrollo ocurre cuando existe más de un stakeholder en la organización y estos poseen distintos grados jerárquicos entre si, provocando desautorizaciones entre ellos, esto es muy común cuando el desarrollo no está enfocado a quien paga el proyecto.

El factor social también puede estar presente entre desarrolladores con distintas posiciones jerárquicas dentro de una organización.

En cuanto a factores psicológicos, éstos son más fáciles de identificar, pero muy difíciles de erradicar, debido a que está ligado al comportamiento de una persona, un ejemplo claro de este factor ocurre cuando un stakeholder no es capaz de explicar qué es lo que quiere, ya sea por medio de sus palabras u otro medio, este comportamiento es parte de él como individuo y como es de esperar, no es un aspecto que se pueda mejorar en corto plazo.

Cuando existen problemas de comunicación, se debe establecer un canal que proporcione comodidad al stakeholder, debido a que el intercambio de información, en la mayoría de los casos, se realiza de manera presencial. ProjectMNG busca dar solución a los problemas de comunicación que nacen de los factores sociales y psicológicos creando un canal de comunicación que va de la mano con el flujo de trabajo en el desarrollo de un proyecto, implementando una *perspectiva vertical* [2], que permite unificar conceptos internos (Desarrolladores) y externos (Stakeholders) de organización o definiciones en general que permiten disminuir la ambigüedad.

También se busca dar solución a los factores de proyecto y organizacionales entregando herramientas simples de especificación, planificación y control de avance para proyectos.

Esto generalmente es poco efectivo debido a la ambigüedad que existe en las comuni-

caciones persona a persona, además, las herramientas que han sido creadas para la gestión de proyectos, como Wrike, Sinnaps que se orientan en el trabajo ágil o IBM Rational Doors que es tradicional, se limitan a la entrega de información del proyecto y no entregan ningún tipo de integración para stakeholders dentro de este.

El software actual a pesar de entregar herramientas de comunicación mediante servicios de streaming o chats dentro de sus mismas plataformas de gestión, no se ofrece control ni participación activa para stakeholders.

Por esta razón ProjectMNG entrega una manera de gestionar un proyecto en forma conjunta con los stakeholders, creando un canal de comunicación directo, estable que reduce la ambigüedad, que siempre está disponible y que permite acelerar el proceso de desarrollo, mejorando el producto final y creando una experiencia para los stakeholders.

ProjectMNG, es capaz de gestionar Proyectos que constan de tres partes: Equipo, Stakeholders y Características.

- Los Equipos, son los participantes del proyecto y son quienes pueden participar activamente dentro de la planificación del proyecto en la plataforma.
- Los Stakeholders son clientes, usuarios, dueños o cualquier interesado o participante del proyecto.

Estos actores, son representados a través de cuentas temporales creadas dentro de un proyecto que permiten a estos acceder al sistema por medio de una interfaz dedicada a ellos y de esta forma poder participar, interactuar o simplemente mantenerse al tanto de un proyecto.

La participación de los stakeholders, es por medio de la aceptación o rechazo de una característica, ya sea en su definición o en su ejecución y a través de solicitudes de cambio, todas estas acciones se ven reflejadas en las características a través de cambios de estados, los cuales son representados de manera gráfica en el sistema.

- Las Características son, como su nombre lo dice, características del proyecto, estas son muy versátiles pudiendo representar desde una historia de usuario o un requisito hasta un hito dentro de una planificación. Las características están compuestas de Actividades que son aquellas tareas que se deben llevar a cabo para dar por concluida una Característica.

2.1. Tecnologías y herramientas

ProjectMNG, es un sistema Web, lo que le da la capacidad de adaptarse a los distintos dispositivos donde se utilice, ya sea teléfonos inteligentes, tabletas o computadores.

Desarrollado en el framework .net core 2, bajo el lenguaje de programación C# 7.1, que es una plataforma de código abierto [6] con mucha versatilidad, velocidad, estabilidad y una gran comunidad detrás, que además permite la utilización del modelo de desarrollo de tres capas MVC (Modelo-vista-controlador), que es con el cual se diseñó el sistema y que permite un desacoplamiento entre el negocio (modelo) y la vista.

.Net core 2, está pensado para sistemas escalables y con altas necesidades de rendimiento, permitiendo su ejecución en servidores Linux, por lo tanto encaja muy bien con ProjectMNG.

Asp.net core, posee su propio entorno de desarrollo (IDE), desarrollado por la misma empresa, Visual studio code, que logra disminuir considerablemente los tiempos de desarrollo debido a su excelente sistema de debugging, siendo esta una de las razones por las cuales no se utilizó frameworks tan famosos como laravel, django, entre otros.

La tecnología utilizada para la base de datos es mySql community 8.0.13 [8] , debido a que este permite modelar y crear bases de datos relacionales, que es un tipo de base de datos en donde todos los datos se almacenan y se accede a ellos por medio de relaciones previamente establecidas, además es multi-plataforma: Para S.O. como Windows, Linux y Mac.

mySql, posee una gran comunidad detrás lo que hace fácil encontrar información sobre su funcionamiento por medio de tutoriales, foros, etc. Al ser una base de datos que se utiliza en multitud de aplicaciones web existentes, mySql es una base de datos ampliamente probada por distintos usuarios y empresas con alto éxito.

mySql, para el funcionamiento habitual de una aplicación incluye características mínimas que sirven ampliamente para la aplicación sin tener que recurrir a un DBA para que administre la base datos.

El coste total de inicio es ampliamente inferior al de Oracle o Microsoft SQL Server, podrá crear servidores mySql sin incurrir en altos costes. Una de las grandes ventajas de utilizar mySql, es la gestión de las transacciones. Las transacciones en mySql son Atómicas, consistentes, aisladas (isolated) y durable; ACID.

Para el diseño de la interfaz, se utilizó CSS3 por medio del framework Bootstrap, debido a que esta tecnología permite crear fácilmente interfaces responsive la cual es una

característica indispensable si se quiere desarrollar un sistema versátil.

También se utiliza JavaScript a través del framework JQuery, que en conjunto con Ajax, permite realizar acciones en tiempo real sin la necesidad de recargar la página, logrando un acabado de mejor calidad en las interfaces de usuario y creando una experiencia mucho más agradable de uso.

2.2. Software de planificación y gestión de proyectos

Las estimaciones, se hacen dentro de un marco de tiempo limitado, definido al comienzo de un proyecto, y pueden actualizarse regularmente durante el progreso.

Al momento de planificar un proyecto se debe permitir definir los escenarios del mejor al peor caso, de modo que los resultados del proyecto puedan limitarse, permitiendo una mejor gestión de cumplimientos.

Generalmente una planificación se desglosa al estilo de una carta Gantt, en el caso del software es igual, pero con funcionalidades amigables que permiten crear alertas o avisos y participar con más de un integrante a la vez en un mismo proyecto.

Existe software que responde a metodologías de desarrollo específicas, como por ejemplo, la forma de trabajo llamada *kanvan*, que a través de una pizarra digital gestiona las tareas a realizar y mantiene un flujo de trabajo, aunque no es capaz de gestionar ese flujo. Para el caso de metodologías tradicionales, se aplica de la misma forma, ProjectMNG no afecta la metodología, es una forma de gestionar tareas y procesos, siendo un complemento al trabajo.

En el caso de ProjectMNG pasa algo similar, pero no se limita a ser una agenda de actividades, sino que responde a una necesidad de gestión que es independiente de la metodología de trabajo, capaz de responder a distintas problemáticas, en donde los distintos tipos de actividades, pueden agruparse en conjuntos llamados característica.

La Característica, puede llegar a transformarse en otros tipos de entidades, ya sea Requisito o Historia de usuario, todo esto depende de las necesidades del proyecto o del equipo que lo está llevando a cabo.

Dejando de lado el desglose del proyecto en Características, la gestión de proyectos dentro de ProjectMNG y la planificación es similar a la que todo software de planificación entrega, osea una representación amigable de lo que sería una carta Gantt, permitiendo planificar actividades y asignar recursos, que en en el caso de ProjectMNG sólo son los miembros del equipo de un proyecto.

Este cambio en la forma de definir un proyecto no sólo trae beneficios al stakeholder, sino que también al proyecto, el cual debe ser desmembrado característica a característica, forzando un entendimiento detallado de este y de cada uno de sus componentes, tal como ya se ha mencionado, esta forma de definición de un proyecto va acompañada por la constante validación del stakeholder.

2.3. Integración del Stakeholder

Dentro del desarrollo de un proyecto, se utilizan herramientas, procesos, metodologías que ayudan y guían hacia el producto deseado. En general todas estas están orientadas al responsable de llevar a cabo el proyecto, dejando de lado, desde el punto de vista del proceso, a los interesados en el proyecto (Stakeholder), quienes muchas veces son los dueños del proyecto.

Hace años en el mundo del software, no poder integrar al stakeholder en los procesos, se convirtió en un problema recurrente, debido a que si durante el desarrollo del proyecto el objetivo se perdía, no había forma de que alguien pudiera percatarse de esto y corregir, sino hasta etapas finales del proyecto, para solucionar y solventar este problema es que nacieron metodologías ágiles de desarrollo.

Las metodologías ágiles, crean una forma de abordar proyectos que integra stakeholders, dándoles un espacio a los dentro del desarrollo como un agente activo [9].

Esta es una buena solución, aunque con problemas, uno muy recurrente es que los stakeholder, generalmente, al no ser desarrolladores, no manejan ningún nivel de conocimientos en las ciencias de la computación, por lo tanto no entienden la manera de proceder dentro de un desarrollo, ni tampoco logran expresar sus ideas de forma efectiva.

Por esta razón es que se crea un sistema de gestión y planificación de proyectos con un espacio donde los stakeholders puedan participar y gestionar su proyecto sin tener que recurrir a reuniones presenciales rutinarias.

Para los stakeholder se crea una sección en la plataforma, especialmente para ellos y sus necesidades, dando libertad de participar aceptando y rechazando características, peticiones de cambios, nuevas ideas y validando el avance de sus ideas y proyectos, permitiendo así también estar totalmente al tanto del avance, percances y retrasos, dándoles un control que nunca antes habían tenido y una vía de comunicación que genera información constantemente, directo desde la fuente para los desarrolladores.

La sección de la plataforma enfocada a los stakeholder, entrega una forma de visuali-

zación de características especializadas en ellos, presentando aquellas características con mayor importancia, ya sea a nivel logístico del proyecto o a nivel operativo de ProjectMNG, para que pueda interactuar. También se incluye una sección donde puede administrar los cambios que se van realizando al sistema por medio de las solicitudes de cambio. Posee además una sección en donde puede ver el estado de avance a nivel macro y micro del proyecto con los detalles que el encargado decidiera dejar para el stakeholder.

2.4. Trabajo relacionado

En el mundo de la planificación de proyectos existe un software para todos los gustos, que cubren muchos ámbitos de las necesidades de gestión, algunos muy avanzados, populares y trabajados como Wrike o Sinnaps, que permite gestionar tareas, control de tiempos, carta Gantt, gestión de carga de trabajo y una infinidad de herramientas útiles al momento de gestionar un proyecto, que al igual que ProjectMNG son un complemento a la gestión de un proyecto.

Cabe destacar herramientas como Trello, Kanban flow que permiten llevar un control simple de tareas, pero estas herramientas son parte de un tipo de metodología, llevando a la digitalización la pizarra, ProjectMNG busca ser más que eso y ser un complemento a la gestión de proyectos, equipos (developers) y usuarios (stakeholders). El caso de google docs o herramientas cooperativas online por el estilo, es como decir *¿Para qué utilizar calculadora si hay ábaco?* o *¿Para que viajar en avión si a pie llego?* por eso y para evitar comparaciones sin ningún sentido se decide dejar de lado plataformas de ese estilo, pero dejando claro que si se quisiera llevar un proyecto con papel y lápiz, esto es posible pero no es conveniente bajo ninguna circunstancia.

Existen otras herramientas de gestión de empresas reconocidas mundialmente, como IBM con Rational DOORS, que permite la colaboración en tiempo real, gestión y control de cambio garantizando la trazabilidad, consistencia y la *posibilidad de reutilización* de los *requisitos*, esto no es más que un gran data warehouse de proyectos, siendo muy similar a lo que ofrece cualquier herramienta de gestión y como ya se mencionó, la gestión de proyectos en cualquier software es algo que se lleva de forma excepcional a nivel operacional y ProjectMNG busca llegar posteriormente a esa excelencia funcional, que para efectos del proyecto de esta memoria se a simplificado.

Rational DOORS además provee una funcionalidad que permite crear *Vistas* con recursos para que pueda ser enviada a los stakeholders. Esto es un claro intento de inclusión

para stakeholders dentro de un desarrollo, pero más que inclusión es una forma de reportar, no es más poderoso que un simple pantallazo. Este software está claramente orientado a empresas, entregando herramientas de organización y documentación directamente relacionado a la organización grupal, control gerencial y no a proveer un medio de comunicación y participación, al igual que las herramientas mencionadas antes.

3. Metodología

3.1. Metodología de desarrollo

Este proyecto se divide en tres fases: Especificación, Desarrollo y la Evaluación.

En el desarrollo, se utilizó una metodología ágil muy efectiva para desarrollos que buscan resultados rápidos, llamada *Extreme Programming for a single person team (PXP)*, que es una adaptación de la metodología ágil XP [1] para desarrollo de una sola persona.

Las ventajas más claras en *PXP* es su orientación tan efectiva al producto, permitiendo comenzar con las pruebas muy tempranamente en el desarrollo, sin la necesidad de tener un producto terminado 100%.

La gran adaptabilidad al cambio de *PXP* es fundamental en cualquier desarrollo que no está definido ni estructurado en su totalidad desde sus inicios, forzando en cada iteración a replantearse las prioridades de desarrollo y las dificultades de cada requerimiento, permitiendo un crecimiento del software desarrollado consistente y sin sorpresas en cuanto a dificultades se refiere.

A pesar de ser una metodología que encaja de una manera excepcional a lo que se requiere en este desarrollo, es imperativo resaltar su gran debilidad, que aunque sea muy marcada esta debilidad, es análoga a todas las metodologías ágiles, la falta de documentación que se genera, aunque se generen listas de prioridad e historias de usuario, estas no pueden ser consideradas un documento formal que nace de esta metodología.

Volviendo a la *PXP*, a continuación, se muestra un resumen de lo que es. En *PXP*, existen cuatro fases las cuales se resumen brevemente a continuación:

- **Planning:** Como su nombre lo dice, es aquí donde se planifica la iteración, aquí es donde nacen las listas de prioridad y se crean las historias de usuarios, para posteriormente transformarlas en requerimientos que desarrollar.

- **Development:** En este punto, se toma la lista de prioridad creada anteriormente y se trabaja sobre ella.
- **Post Mortem:** Una vez terminado el desarrollo, aquí se verifica que el código haya pasado por todos los test y se encuentre en proceso de producción de forma exitosa.
- **Exit:** El software está finalizado.

Para mayor detalle sobre la metodología y su flujo ver Anexo E.

3.2. Evaluación del sistema

La evaluación se realiza por medio de un módulo de encuestas embebido en el sistema, con el cual se recopila información a través de encuestas personalizadas para cada vista.

Al realizar encuestas específicas para cada vista se obtiene una mayor cantidad de información permitiendo mejorar de manera más rápida y efectiva el sistema.

La recolección de información se logra mediante la liberación de un Beta a usuarios seleccionados.

Esta evaluación, se basa en los modelos propuestos por Beltré Ferreras [4], la decisión de utilizar este modelo va directamente ligado a la complejidad y el tiempo, ya que modelos de evaluaciones basados en normas como el ISO 9126-1, a pesar de ser muy precisos requieren un gran costo operacional para ser llevados a la práctica.

A continuación se define el camino a tomar en cuenta al momento de evaluar un sistema:

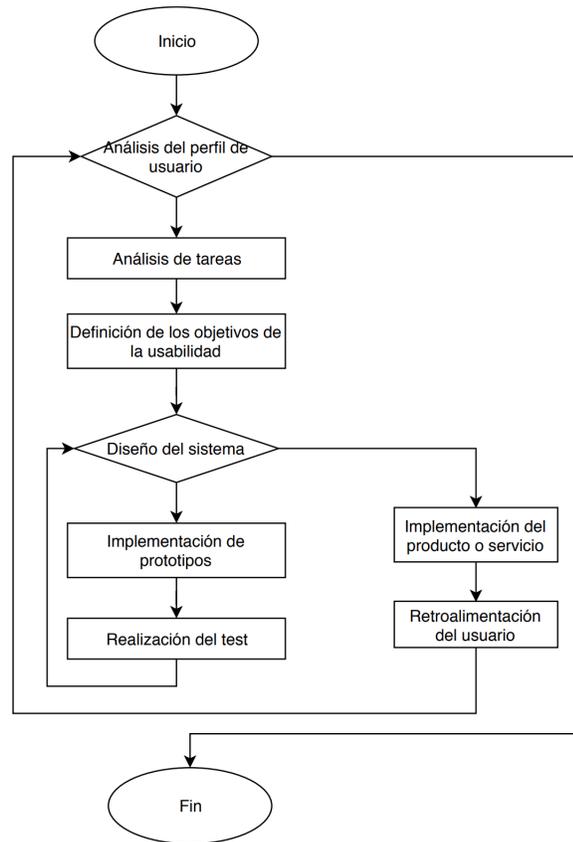


Figura 3.1: Diagrama de flujo de evaluación

Para el caso específico de este sistema, utilizamos una variación de este modelo que se acomoda mejor a la forma de operar, debido a que ProjectMNG, posee su sistema de evaluación interna dejando el flujo de evaluación de la siguiente manera:

Otra razón para la variación que se ve en la Figura 3.2, se debe a que el diseño del sistema fue una etapa previa en donde no se tomó en cuenta ningún tipo de encuesta ni opinión de usuarios.

A continuación, se describen las partes y características de la metodología de evaluación:

- **Análisis del perfil del usuario** – se define el perfil de usuarios potenciales a partir de los cuestionarios y entrevistas. Se describen los factores más relevantes de impacto que inciden sobre la usabilidad del producto.
- **Análisis de tareas** – se describen las tareas que realizan los usuarios, los flujos de trabajo y se levantan las necesidades de información que requieren para realizar sus

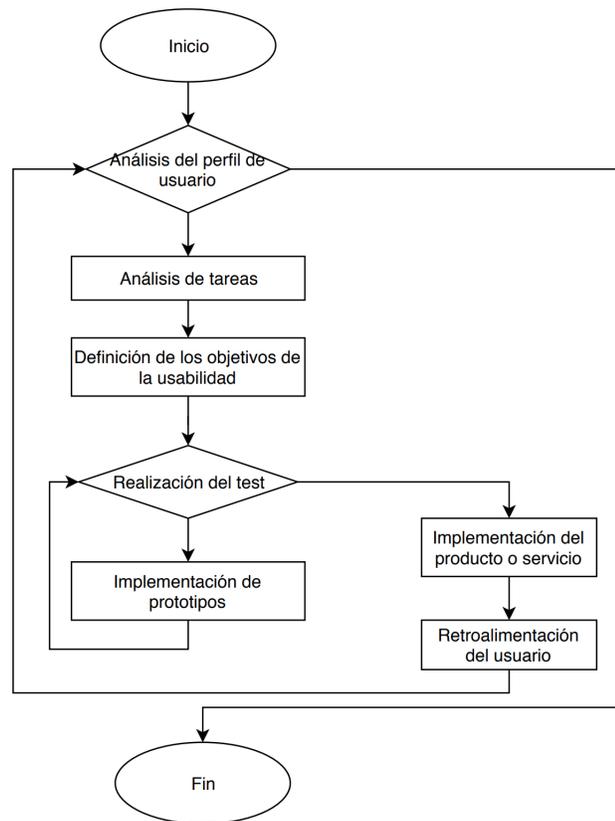


Figura 3.2: Diagrama de flujo de evaluación modificado

trabajos.

- **definición de los objetivos de usabilidad** – se especifican los objetivos cualitativos y cuantitativos de la usabilidad y los parámetros claves, que se utilizan durante los procedimientos de las pruebas.
- **Implementación de prototipos** – se realiza un estudio experimental de determinados aspectos del sistema, reduce el tiempo y costo de desarrollo del producto o servicio y permite, de esta manera, la realización pruebas con los usuarios potenciales.
- **Realización de pruebas** – es un proceso donde no sólo se verifican y validan los prototipos, sino también se evalúa su usabilidad.
- **Rediseño** – es un indicador de decisión basado en los resultados de los análisis de las pruebas. Se identifica el prototipo del producto o servicio que no cumple con los

requerimientos y estándares establecidos; se desvía el flujo del ciclo de desarrollo a la definición de los objetivos de usabilidad. En otros casos se inicia el rediseño a partir del proceso de análisis de tareas.

- **Implementación del producto o servicio** – se realiza después de la evaluación de los prototipos y de su aceptación, se inicia la implementación del producto o servicio con toda su funcionalidad y prestaciones previstas.

Este proceso se relaciona con las actividades de actualización y mantenimiento del sistema.

- **Retroalimentación del usuario** – comienza concluida la instalación del proyecto para ser utilizado en línea, se obtienen nuevas informaciones complementarias del usuario con el propósito de usarlas para mejorar el diseño del sistema, de nuevas versiones y de nuevas funcionalidades y características del sistema, para lo cual se utiliza el sistema de evaluación de ProjectMNG.

Las encuestas no son anónimas, debido a que son pruebas controladas, obedeciendo en algunos aspectos al modelo de evaluación planteado por Hix y Hartson [5].

El modelo de evaluación se aplica en todas las áreas y no sólo en interfaz, tratando de crear una evaluación automática, empírica [5] y formal a la vez, con la finalidad de reducir los tiempos.

Luego de las encuestas se contacta a cada uno de los testers para realizar diversas preguntas relacionadas con sus respuestas dadas en el sistema.

Para las preguntas de la encuesta, el sistema se divide en siete partes que, además, corresponden a los módulos del sistema: Administration, Activity, Module, Project, Characteristic, ChangeRequest y StakeHolder.

En cada una de las secciones del sistema, se formulan preguntas específicas de acuerdo a su uso, en el caso de StakeHolder, este contiene una visualización simplificada de módulos y características a demás de todas las visualizaciones más completas, pero por la simpleza se hace solo una evaluación, ya que el objetivo del conjunto de todas las vistas es común entre todas ellas y por sobre todo se evalúa la disposición y la comodidad al momento de visualizar los elementos.

Las respuestas a esas encuestas son almacenadas en en una base datos, concretamente en tres tablas que posteriormente son analizadas.

4. Implementación de la Solución

Al construir un proyecto de software, sin importar la metodología, se definen entidades atómicas que representan partes o funcionalidades del mismo, en el caso de metodologías tradicionales, estas entidades son los *requisitos* y en metodologías ágiles estas son denominadas *historias de usuarios*, que de igual manera son requisitos, pero escritos en un lenguaje que es común para el stakeholder.

En la creación de este sistema de gestión de proyectos, se define una abstracción de estas definiciones atómicas, llegando de esta manera a crear la *Característica*, esta entidad es capaz de englobar en simples términos una gran cantidad de formatos existentes en distintos tipos de metodologías, que incluso van más allá del rubro desarrollo de software.

La entidad Característica, es quien contiene la planificación real del proyecto en forma de *Actividades*, que responden a la misma idea que entregan las tareas de una carta Gantt, aunque con menos atributos, a su vez existen otras entidades llamadas Módulos, que están encargadas de clasificar a las Características y darles una forma de clasificación simple.

La Característica, es la ventana que tienen los StakeHolders para ver hacia el interior de un proyecto, la herramienta que esta le entrega para poder interactuar con esta, es la *Solicitud de cambio*, que es el medio formal para solicitar la realización de modificaciones dentro del sistema.

La solicitud de cambios, también debe ser utilizado por los desarrolladores, lo que obliga a ambas partes a estar de acuerdo para poder llevarlas a cabo. Una vez realizado el cambio se crea un registro histórico, con el fin de poder dilucidar posteriormente los cambios sufridos del proyecto y sus responsables.

4.1. Concepción

A continuación se muestran los modelos UML Figura 4.1 y modelo de tablas Figura 4.2 que se pensaron desde un comienzo para el desarrollo de ProjectMNG:

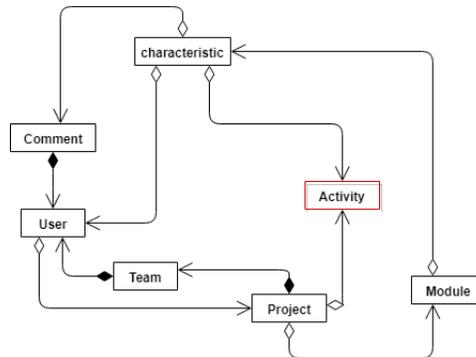


Figura 4.1: Primer Diagrama UML

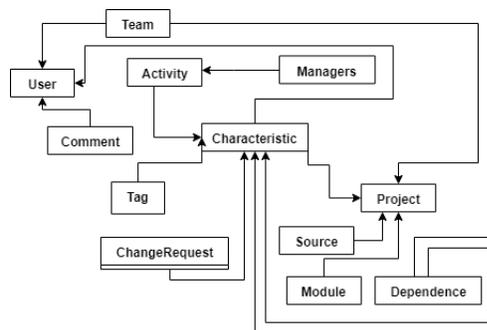


Figura 4.2: Primer Diagrama de tablas

Como se puede observar en un principio se consideró que el manejo de archivos no sería a través de objetos, por lo cual solo se consideró una tabla, claramente este modelo es capaz de cumplir con las necesidades del sistema, pero no es capaz de dar una trazabilidad y sustentabilidad de datos en el tiempo.

El detalle de las implementaciones del modelo y la base de datos y su evolución en el tiempo se verán a continuación.

4.2. Planificación

A modo de una planificación preliminar, se crea una pseudo Carta Gantt con la finalidad de definir fechas de cumplimiento para las distintas tareas que se debían abarcar, pero esta planificación no tiene cabida dentro de la planificación y las acciones tomadas dentro de la metodología PXP aplicada.

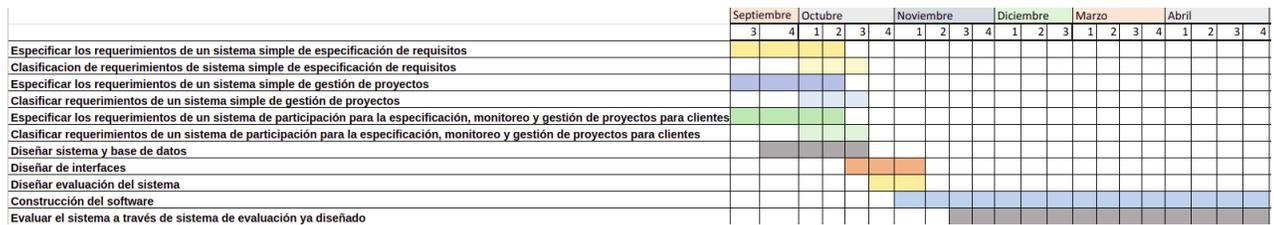


Figura 4.3: planificación simple

Para la planificación aplicada a PXP se divide el trabajo en sprint de 2 semanas según el tiempo designado para la construcción del proyecto.

- Planning

- Sprint 0 (Fase planning): De todas las historias de usuario disponibles anexo x se traspasan a características obteniendo lo mostrado en el Anexo A. Se ordena por prioridad todas las tareas y se crear su estimación de esfuerzo en días como se muestra en el Anexo A.2, su test está asociado directamente a la descripción de la característica y al cumplimiento de esta misma interactuando con las características relacionadas de acuerdo al diagrama UML y al diagrama de tablas creado.

- Development

- Sprint 1: de acuerdo al tiempo del sprint se toman las siguientes tareas:
 - (Características)CRUD
 - (Características)Listar para usuario o para stakeholder
 - (Características)Solicitud de cambio

Todas las tareas cumplidas con éxito dentro del tiempo. Se revisa la lista de prioridad dentro de este sprint debido a que era necesario un cambio subiendo un puesto la creación de equipos y bajando el historial de cambios, ya que esto tenía más sentido.

- sprint 2: Lista de prioridad sin cambio, se toman las siguientes tareas y se cumplen con éxito:
 - (Características)Subir archivos
 - (Características)Historial de cambios
 - (Características)Comentarios entre el equipo de desarrollo
 - (Usuarios)CRUD
- sprint 3: se agrega una nueva historia de usuario (Notificaciones), agregando características a la lista, la prioridad queda de igual manera, agregando las nuevas tareas al final. Se toman las siguientes tareas y se cumplen con éxito:
 - (Características)Estados en Características
 - (Características)Con dependencias
 - (Usuarios)Poder registrarse como usuario
 - (Usuarios)Poder crear cuentas temporales para stakeholders

en cuanto al testing hay comunicación entre módulos.
- sprint 4: Se necesita un cambio en el UML debido a que los archivos no fueron bien considerados, se modifica y actualiza el modelo en el código, no afecta en cuanto tiempo, cambian las prioridades agregando Subir archivos a la sección de características ya que no existía hasta este momento. Se toman las siguientes tareas y se finalizan con éxito:
 - (Proyecto)CRUD
 - (Proyecto)visualización de avance
 - (Proyecto)Listar proyectos
 - (Proyecto)Tienen equipos de usuarios
- sprint 5: No se realizan cambios en la lista de prioridad. Se toman las siguientes tareas y se finalizan con éxito:
 - (Proyecto)historial de cambios
 - (Stakeholder)Vista de proyecto resumido
 - (Stakeholder)Realizar solicitudes de cambio
- sprint 6: No se realizan cambios en la lista de prioridad. Se toman las siguientes tareas y se finalizan con éxito:

- (Stakeholder)Solicitud de creación de requisitos
- (Actividades)CRUD
- (Actividades)Asociar a un requisitos
- (Actividades)Con tiempo de duración
- (Actividades)visualizar en un calendario

en cuanto al testing hay comunicación entre módulos.

- sprint 7: No se realizan cambios en la lista de prioridad. Se toman las siguientes tareas y se finalizan con éxito:
 - (Actividades)visualizar en un calendario
 - Notificaciones cuando seas agregado a un equipo.
 - Notificaciones cuando alguien se integre a tu equipo.

en cuanto al testing hay comunicación entre módulos. Se dejan pendientes las funciones de notificaciones restantes.

- Post Mortem: esta fase no se puede cumplir realmente, ya que no se pasa a "Producción" en el estricto rigor, pero se realiza un testing en un ambiente controlado de satisfacción y este resulta exitoso al finalizar.
- Exit: Esta fase no se considera dentro del proyecto debido a su acotado alcance, pero sin ser del todo rigurosos podríamos decir que se obtiene un proyecto testeado

4.3. Sistema

En cuanto a la arquitectura del sistema se optó por una basada en capas como se ve a continuación en la Figura 4.4:

Claramente esto es una estructura MVC genérica, pero en este caso el modelo se separó de la base de datos, ya que se decide diseñar una base de datos, para que de esta forma pueda ser optimizada y personalizada a libre opción a lo largo del proyecto. Si se hubiera optado por trabajar con Entity Framework para MySQL, la libertad de cambio habría estado limitada a lo que ofrece este framework.

Cabe destacar que en el modulo *Modelo* reside el modelo UML de la Figura 4.5, en el modulo *Base de Datos* contiene el modelo de base de datos presente en la Figura 4.13 y el modulo vista está representado en toda la Sección 4.5

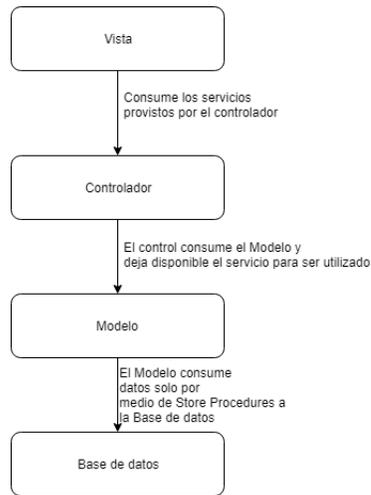


Figura 4.4: Arquitectura del sistema

Para lograr un sistema capaz de responder a la gestión de proyecto y la integración de los stakeholders, fue esencial identificar cada uno de los desafíos que este representaba el modelo 4.5 muestra la culminación de este trabajo de reconocimiento.

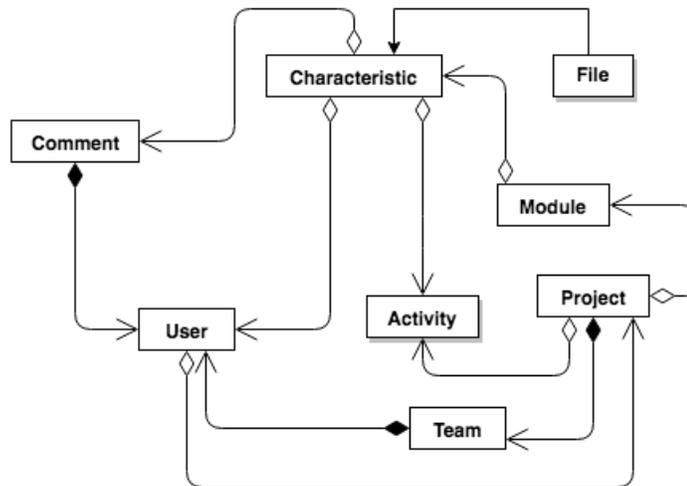


Figura 4.5: UML del sistema

En la gestión de proyecto, las problemáticas fueron resueltas por otros software con

mucho éxito, por tanto nos limitamos a imitar aspectos y características de software ya existente, obviamente adaptando todo esto a nuestra forma de clasificar un proyecto y la Característica.

En el caso de la integración de los stakeholders, cuyo problema es la comunicación y la falta de control, lo solucionamos por medio de la definición de Características y de las solicitudes de cambio.

Ahora desglosaremos cada uno de estos puntos, mostrando como cada decisión en el diseño responde a cada uno de los problemas mencionados.

- **Proyectos:** Esta definición es bastante simple, es a grandes rasgos un título que es lo que se muestra a simple vista y una descripción, que es donde se entra en detalle sobre este.

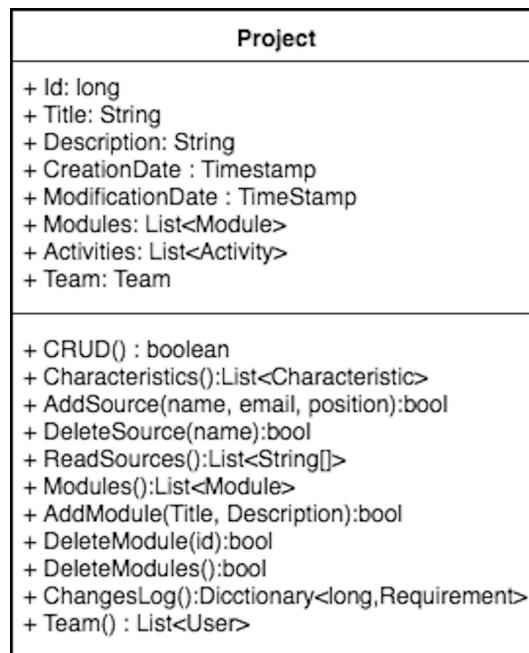


Figura 4.6: Objeto Project

Un proyecto contiene *Sources*, que hace referencia a las fuentes de origen de lo que luego definiremos como *Characteristic*, *Source*, luego también se transforma en una cuenta temporal para que el sistema pueda ser accedido como un stakeholder, *Module*, que es una forma de organización para *Characteristic* y *Team* que se explica a continuación.

- **Equipos:** La necesidad de poder trabajar en equipo dentro de un mismo proyecto pudiendo interactuar unos con otros sin necesidad de utilizar herramientas externas al sistema.

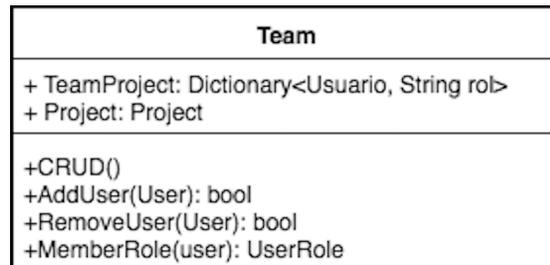


Figura 4.7: Objeto Team

MemberRole, retorna el rol que tiene un usuario dentro del equipo, esto se utiliza para dar o quitar permisos de algunas funcionalidades del sistema, como por ejemplo, poder eliminar usuarios de un equipo.

- **Especificación de un proyecto:** este problema se abarca por medio de la definición de Características (Characteristic) y módulos.

Las características, poseen un conjunto de tareas o actividades (*Activity*), en este sentido la clase Characteristic busca ser una definición estándar para modularizar un proyecto.

Characteristic tiene la posibilidad de ser un requerimiento de software, historias de usuario o como dice su nombre, una característica en general, sólo siendo limitado por las necesidades de quien use el sistema. Algunos ejemplos de uso son:

- Utilizar una Characteristic como historia de usuario y a través de la dependencia de Characteristics definir sus requerimientos posteriormente.
- Utilizar una Characteristic como historia de usuario y a través de las Activities asociadas a un Characteristic definir sus requerimientos.
- Utilizar una Characteristic directamente como requerimiento o característica y definir sus tareas a través de Activities.

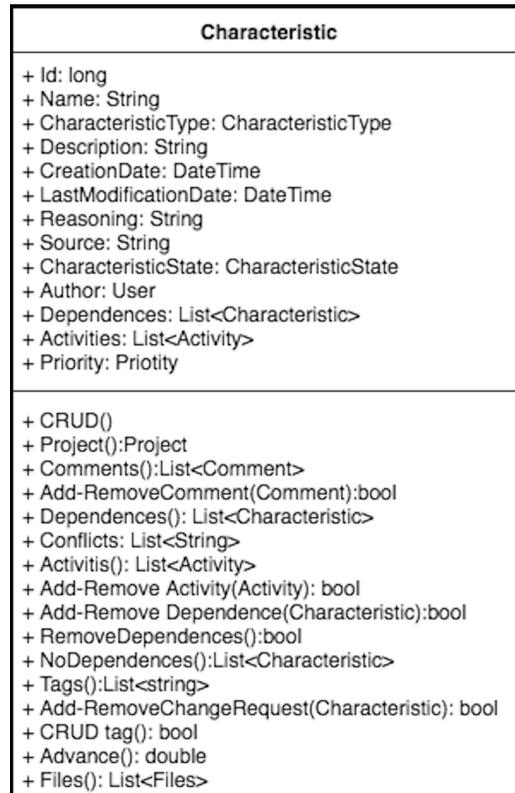


Figura 4.8: Objeto Characteristic

Como se puede observar en la definición Figura 4.8 del objeto Charasteristic, existen atributos como *Priority* y *RequirementType*, que tienen como objetivo clasificar las características en subgrupos al igual que los *Tags*, la única diferencia entre estos, es que los *Tags* son muchísimo más ambiguos y dinámicos.

Las características pueden ser agrupadas en *Modules*:

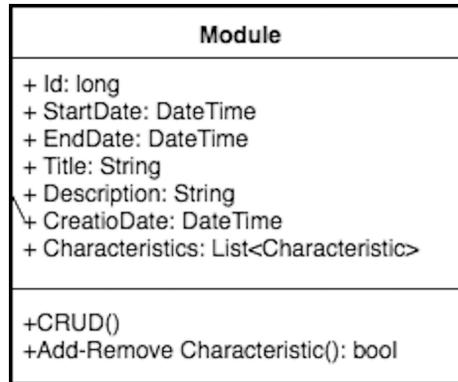


Figura 4.9: Objeto Module

Cuya única funcionalidad es organizar a las características modularizandola, contribuyendo con la planificación de estas, pudiendo asignarles una fecha de comienzo (*StartDate*) y una de finalización (*EndDate*).

- **Planificación:** La planificación de un proyecto se hace por medio de “Activity”, las que son contenidas en “Characteristic”

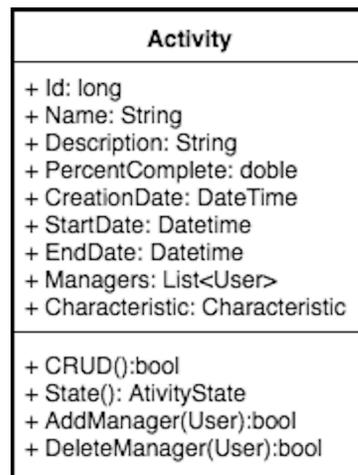


Figura 4.10: Objeto Activity

Activity, tiene la posibilidad de agregar Usuarios (*AddManager(User)*), teniendo como objetivo identificar a esos usuarios como los encargados de llevar a cabo esa actividad, estos usuarios sólo pueden ser aquellos que pertenecen al equipo del proyecto que contiene esta actividad.

- **Comunicación:** La comunicación abarca dos ámbitos y se explica la solución creada por separado a continuación:
 - **Equipo:** La comunicación entre el equipo se realizará mediante un chat provisto por el sistema y por medio de comentarios que estarán disponibles en los distintos módulos del sistema (características, módulos, actividades).
 - **Desarrolladores – Stakeholders:** Esta comunicación se realiza por medio de un chat exclusivo entre desarrolladores y stakeholder, pero principalmente por medio de las solicitudes de cambio (*ChangeRequest*) que el sistema provee para la proposición de cambios dentro de un proyecto.
- **Control de cambio:** Esta sección es controlada por medio de lo que se ha denominado: *ChangeRequest*, este objeto es estrictamente igual a un *Characteristic*, pero temporal, que vive dentro de otro *Characteristic*, pero con su información actualizada.

Si se realiza un cambio dentro de un *Characteristic* este no se actualiza inmediatamente, sino que crea una copia temporal de el mismo con las actualizaciones y estas no se aplican hasta que la contraparte correspondiente acepte.

Hay restricciones de estado para las *ChangeRequest*, por ejemplo, si una *Characteristic* está en espera, el cambio no pasa por un *ChangeRequest*, sino que la actualización se aplica directamente.

Existe un historial de *ChangeRequest*, los que son guardados desde la creación de un *Characteristic*, así como también cada cambio aceptado por medio de un *ChangeRequest*.

El historial de cambios se encuentra en la clase *Project* y se puede ser visto en la Figura 4.6

- **Evaluación y verificación:** La evaluación o verificación de un proyecto se hace por medio de sus Characteristic, estas pueden pasar por diversos estados, los cuales permiten dilucidar claramente su posición dentro del proyecto:

CharacteristicState
Waiting
ChangeRequest
Accepted
Refused
Approved
Reprobed

Figura 4.11: Enum RequirementState

Las Charasteristic constan de seis estados:

- **Waiting:** Significa que está esperando por aprobación.
- **Accepted:** Cuando un Characteristic en estado Waiting a sido aprobado, pasa al estado Accepted.
- **Refused:** Cuando un Characteristic en estado Waiting a sido rechazado, pasa al estado Refused.
- **ChangeRequest:** Cuando se hace una solicitud de cambio desde cualquier estado, que no sea Waitintg o Refused.
- **Approved:** Si un Charasteristic en estado Accepted es finalizado y aprobado por los stakeholders, pasa al estado Approved.
- **Refused:** Si un Charasteristic en estado Accepted es finalizado y rechazado por los stakeholders, pasa al estado Refused.

Para poder pasar del estado Accepted al estado Approved, el stakeholder debe realizar el cambio de estado.

Los desarrolladores pueden facilitar al stakeholder pruebas, en forma de documentos, capturas de pantallas, links o lo que sea necesario para verificar que la característica ha sido finalizada con éxito.

La clase File, se encarga de guardar estos archivos, que posteriormente son asociados a la clase Characteristic en la Figura 4.8:

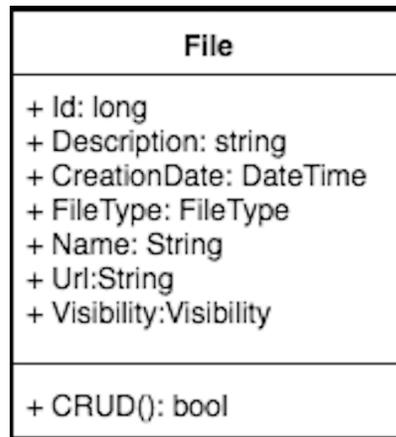


Figura 4.12: Objeto File

Como se puede observar en la Figura 4.12, existe el atributo *Visibility*, que da o quita a los Stakeholders la posibilidad de visualizar los documentos subidos a la plataforma.

Aunque el uso de archivos tiene especial utilidad al momento de probar el avance, para pasar a un estado de aprobación, estos no están restringidos ni limitados de ninguna forma y pueden ser utilizados durante toda la vida del proyecto.

4.4. Base de datos

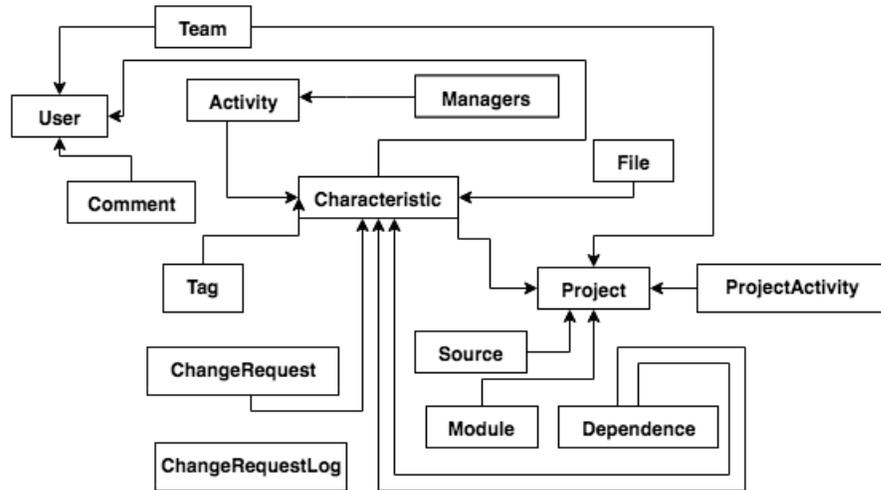


Figura 4.13: Modelo tablas

Como se puede observar en la Figura 4.13 el modelo está muy centralizado en *Characteristic*, esto puede dar la sensación de que se está afectando la escalabilidad del software, pero cabe recordar que *Characteristic* es el corazón y centro del proyecto, esta parte del sistema no debiera ser nunca cambiada, y cualquier funcionalidad que sea incluida en un futuro, bajo ninguna circunstancia debiera modificar esta entidad.

El diseño de la base de datos responde al diseño UML antes señalado, teniendo en su mayoría tablas que corresponden directamente al objeto como:

- **Characteristic:**

Se podría decir que la Foreign Key (FK) *project* es redundante, ya que *Module* pertenece sólo a un *Project*, pero debido a las constantes consultas se decide crear esta FK y así evitar una sobre carga de consultas a tablas innecesarias.

- **ChangeRequest:** La solución para la gestión y control de cambio es llama *ChangeRequest*.

Characteristic
PK id: serial
FK autor: bigint
FK project: bigint
FK module: bigint
code: varchar(256)
name: text
type_characteristic: int
description: text
creation_date: datetime
last_modification_date: datetime
reasoning:text
source: varchar(255)

Figura 4.14: Tabla Characteristic

Este objeto se encuentra almacenado dentro de Characteristic y funciona como una copia temporal de sí misma, la cual guarda todos los cambios realizados, de manera que no sean aplicados de inmediato, aguardando hasta que sean aprobados por la contraparte correspondiente.

Como restricción de esta solución, se da la posibilidad de guardar sólo una copia temporal, limitando la posibilidad de crear más de una solicitud de cambio a la vez y de esta manera no complejizando de forma indebida el control de cambio.

ChangeRequest
PK id: serial
FK requirement: bigint
FK project: bigint
FK author: bigint
name: text
description: text
modification_date: datetime
reasoning:text
source: varchar(255)

Figura 4.15: Tabla ChangeRequest

- **ChangeRequestLog:** Aquí se almacena un registro histórico de los cambios aceptados, realizados por medio de las solicitudes de cambio.

En tabla ChangeRequest se guarda sólo el último cambio vigente, que es eliminado

una vez el ChangeRequest, siendo este aceptado o rechazado, dejando el espacio libre para la siguiente solicitud de cambio.

Este registro de cambios, es vital para el propósito del sistema, esto debido a que permite tener una clara visión de todas las alteraciones que ha sufrido el proyecto, ya sean dentro de los parámetros definidos, así como también todas aquellas cosas que se fueron adoptando a lo largo del desarrollo.

ChangeRequestLog
PK id: serial
FK requirement: bigint
FK project: bigint
FK author: bigint
name: text
description: text
modification_date: datetime
reasoning: text
source: varchar(255)
change_description: text

Figura 4.16: Tabla ChangerequestLog

4.5. Interfaz

- **Pantalla Inicio:** Una vez iniciada la sesión, la primera vista del sistema es la siguiente:

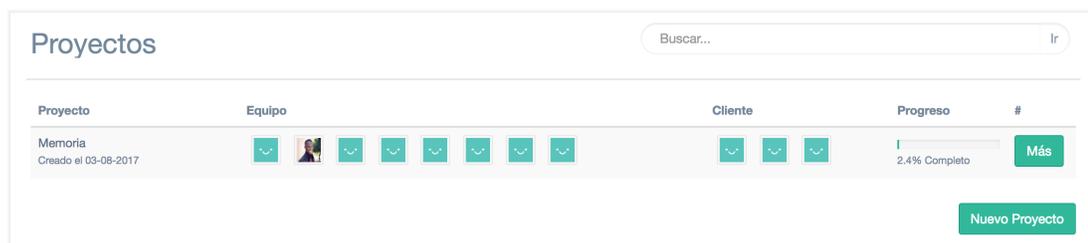


Figura 4.17: Listado de Proyectos

La Figura 4.17 es una lista de todos los proyectos en los que participa, pudiendo visualizar el nombre del proyecto (Memoria), las miniaturas de perfil de usuario,

tanto de los participantes del equipo, como la de los stakeholders (ocho miembros de equipo y tres stakeholders), el progreso del proyecto (2.3% en este caso), luego un botón *Más*, que permite acceder al detalle del proyecto y por último un botón *Nuevo Proyecto*, que permite crear un nuevo proyecto.

- **Detalle de un proyecto:** Al acceder por medio del botón *Más* antes comentado, accedemos a la siguiente vista:

The screenshot displays the 'Detalle de Proyecto' interface. At the top left, the title 'Memoria' is visible. Below it, a grid of circular icons represents team members and stakeholders. The first row includes Pedro Pereira (USER), Ivan Martínez (OWNER, with a real photo), Noah Zamora (USER), Amity Waller (USER), Danielle Massey (USER), and Lee Blankenship (USER). The second row includes Macon Schultz (USER), Caesar Rivas (USER), Casey Roth (USER), Cherokee Reid (USER), and Iván Eduardo Martínez Merino (USER). Below the icons, a section titled 'Stakeholders o Fuentes' lists 'Ivan Eduardo Martinez Merino' with the email 'imartinez11@alumnos.utalca.cl' and the role 'Gerente General'. A green 'Agregar Nuevo' button is positioned to the right. The right-hand side features a 'Descripción' section with the text 'Proyecto para finalizar la carrera', 'Lider de Proyecto: Ivan Martínez', and a list of files: 'Funcional-requirements.docx', 'UAT.pdf', 'Email-from-flatbal.mln', 'Logo.png', 'Contract-10_12_2014.docx', and 'Contract-10_12_2014.docx'. Below the files is a 'Ver Más' link and a form to 'Agregar a tu equipo' with a '+ Escribe su email' input field and 'Modificar' and 'Eliminar' buttons. At the bottom left, there are sections for 'Solicitudes de cambio' and 'Actividades', with a note: 'Se deben poder crear actividades asociadas a al cumplimiento de requisitos'. The date '14 Jun' is shown at the bottom right.

Figura 4.18: Detalle de Proyecto

Como se puede observar en la Figura 4.18 nuevamente podemos ver las miniaturas de los participantes del equipo.

Podemos interactuar con estos elementos, permitiéndonos eliminar y agregar miembros, presionando en la miniatura correspondiente que se quiere eliminar, e ingresando el correo del miembro que se quiera incorporar al equipo en el campo posicionado en la esquina inferior derecha.

Todas estas acciones sólo son permitidas para un usuario con permisos de *Owner*.

En el panel derecho de la Figura 4.18 podemos observar la descripción del proyecto y una lista de documentos que son subidos por los miembros del equipo y stakeholders.

Debajo de los miembros del equipo, se encuentran los StakeHolders (Auspiciadores), que pueden ser agregados fácilmente por medio del botón *Agregar nuevo*, requiriendo un Email, nombre y cargo.

Luego debajo de los stakeholders, se encuentran las solicitudes de cambio pendientes del proyecto.

- **Proyecto:** Al presionar la fila del proyecto en la Pantalla Inicio (Figura 4.17), es posible acceder al contenido del proyecto:

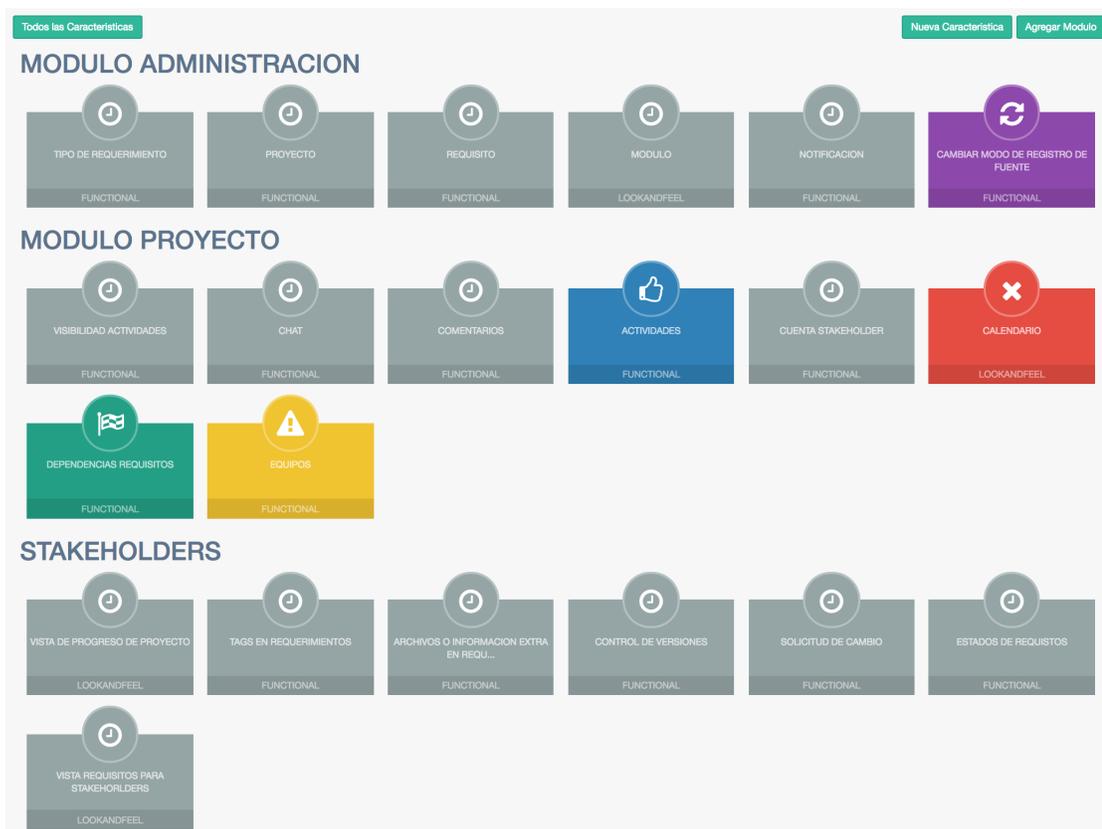


Figura 4.19: Características por Modulo de un Proyecto

En la Figura 4.19 se presenta la información más organizada, ya que divide las Características en grupos llamados Módulos, para mejorar la organización.

En la presentación de las características de esta vista, se decide ir por un concepto colorido, que facilita el reconocimiento de las Características.

El color en las características representa los posibles estados en los que pueden llegar a estar a lo largo de su existencia, siendo la representación de colores explicada en la Figura 4.20:



Figura 4.20: Estados Característica

En la esquina superior derecha de la Figura 4.19 se encuentran dos botones, el primero de ellos tiene la funcionalidad de crear una nueva característica y el siguiente es utilizado para crear un nuevo módulo dentro del proyecto actual.

En la esquina superior izquierda de la Figura 4.19, es posible encontrar el botón que permite cambiar la forma de visualización las características.

Si es presionado este botón *Todas las Características* de la Figura 4.19, la visual cambia quitando los módulos y mostrando todo de forma continua como se muestra en la Figura 4.21:

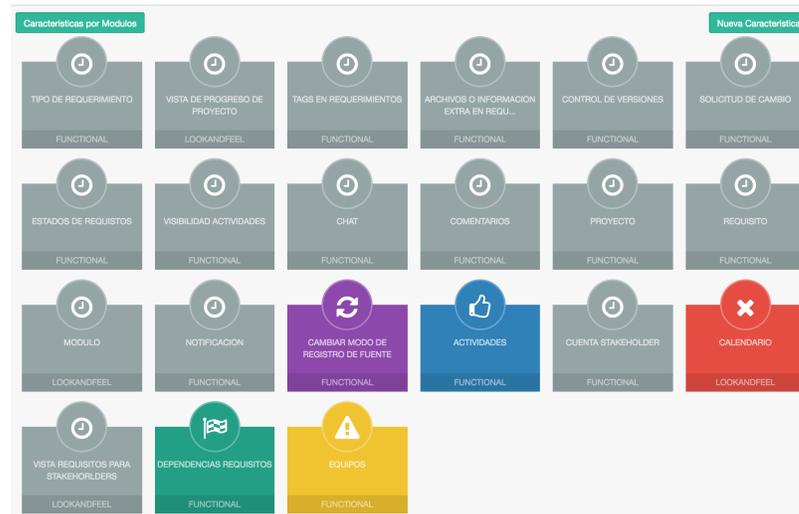


Figura 4.21: Características

En cualquiera de las dos formas de visualización, se puede presionar una de las características para acceder a su detalle.

- **Proyecto para Stakeholders:** La visualización de un proyecto ha sido resumida para los Stakeholder de manera que sólo vean lo que sea necesario y en el orden más apropiado:

The screenshot displays a project management interface. On the left, under the heading "Memoria", there is a list of tasks with progress indicators:

- Vista de progreso de proyecto 35%**: Crear una interfaz simple para la visualización del avance del proyecto
- Cambiar modo de registro de fuente 2%**: Que la fuente se pueda agregar antes y que luego al crear un requisito se seleccione este, que ya fue previamente creado
- Cuenta StakeHolder 0%**: Poder crear cuentas para los stakeholder que tengan acceso a la vista para cliente
- Vista Requisitos para stakeholders 0%**: Vista de visualización de requisitos, que permite ver avances y características y también permite aceptar rechazar o hacer una petición de cambio

Below the tasks, there is a section for "Solicitudes de cambio" and "Actividades". The "Actividades" section shows a task: "Se deben poder crear actividades asociadas a al cumplimiento de requisitos" with a count of 14 and the date "Jun".

On the right side, there is a detailed view for the selected task, titled "Descripción". It includes the project name "Proyecto para finalizar la carrera", the project leader "Lider de Proyecto Ivan Martinez", and a list of related files under "Archivos":

- Functional-requirements.docx
- UAT.pdf
- Email-from-flatbal.mln
- Logo.png
- Contract-10_12_2014.docx
- Contract-10_12_2014.docx

A "Ver Más" link is provided at the bottom of the file list. A button at the top right of the interface reads "Ver Características por Modulos".

Figura 4.22: Proyecto para Stakeholders

La representación de las características, ahora se encuentra en forma de lista, esta es ordenada desde lo más urgente de atender en la cima de la lista hasta lo menos urgente al final de la lista.

Es posible hacer click en la característica para desplegar un menú que contiene los archivos relacionados a esa característica y un botón que lleva hacia el detalle de esta.

En la zona inferior se muestran las solicitudes de cambio pendientes y en el panel derecho una lista de archivos en conjunto con el nombre del encargado del proyecto y una pequeña descripción.

- **Característica:** Las opciones de interacción con una característica están limitadas por quien las está visualizando y el estado en el que se encuentren. A continuación se mostrará una con acceso total debido a que es accedida en un estado de Waiting:

Característica FUNCIONAL StakeHolders volver

Archivos o informacion extra en requerimientos Completado 0% Prioridad: LOW Creación: 09/19/2017 22:38:50

Descripción: Poder subir archivos o informacion(links) a modo de prueba para el cumplimiento de los requisitos

Razonamiento: Darle la facilidad al stakeholder de poder verificar el avance correcto de su proyecto

Información

Creado por: Ivan Martinez CEO a.com Email: llaoku_saope@hotmail.com

Auspiciador: Ivan Eduardo Martinez Merino

Última actualización: 09/19/2017 22:50:56

Autor: Ivan Martinez

Pre-Requisitos:

Requisito	Descripcion	Progreso
Requisito	Poder especificar una característica del proyecto	0

Print Actividades Archivos Eliminar Actualizar

Figura 4.23: Detalle Característica

En este estado, aún se puede eliminar sin confirmación una característica, así como también actualizar su información, esto se puede observar en los botones de la esquina inferior derecha que están disponibles en la Figura 4.23.

En la zona superior de la Figura 4.23 podemos observar qué tipo de característica es, luego en qué módulo está ubicado, esto puede ser cambiado sin ningún tipo de permiso por parte de stakeholders, ya que esto sólo afecta en la organización de las características, luego hay un botón que permite volver a la vista anterior.

Debajo de la zona mencionada anteriormente, se encuentra el nombre de la característica, su avance, su prioridad y a continuación, una descripción y el razonamiento, seguido de información de su creación, así como también de sus últimos cambios.

En la zona inferior se encuentra un listado de los pre-requisitos de esta característica, luego tres botones, que permiten hacer: impresión de pantalla, ver actividades relacionadas a esta característica y los ver los archivos de esta misma, respectivamente.

La disposición de esta vista está pensada para dar fácil acceso a la información, mediante una visualización rápida, ubicando lo más importante arriba, aunque la lista

de pre-requisitos puede parecer más importante que la información de creación de la característica, la cual puede crecer bastante, dejando esta información inaccesible, por esa razón se decide dejar al final.

A continuación en la Figura 4.24 muestra la interfaz *detalle de característica* en otro estado:

Característica FUNCIONAL Modulo Proyecto [← volver](#)

Equipos Creación: 09/15/2017 22:31:48

Descripción: Poder generar equipos entre usuarios por proyecto

Razonamiento: Poder trabajar en equipo dentro de un mismo proyecto

Información

Creado por: Ivan Martinez
CEO
a.com
Email: laoku_saope@hotmail.com

Auspiciador: Ivan Eduardo Martinez Merino

Última actualización: 09/15/2017 22:55:45

Autor: Ivan Martinez

[Print](#) [Actividades](#) [Archivos](#) [Actualizar](#)

Figura 4.24: Detalle Característica sin permisos

En la Figura 4.24 no está presente el botón *Eliminar* y aunque siga presente el botón *Actualizar*, este no genera un cambio inmediato sino que produce una Solicitud de Cambio.

- **Solicitud de cambio:** Estas son generadas cuando un cambio es realizado en una característica, que se encuentra en un estado Accepted, permitiendo visualizar y registrar fácilmente los cambios, dando la facilidad de rechazar o aceptar los cambios por ambas partes, aplicando inmediatamente los cambios y guardando el el registro de cambio correspondiente en caso de ser aceptada:

Solicitud de cambio

Actividades Completado 0% Prioridad: LOW

Antes

Descripción: Razonamiento:

Se deben poder crear actividades asociadas a al cumplimiento de un requisito

Gestionar más atómicamente la realización y el cumplimiento de un requisito

Información

Creado por: Ivan Martínez CEO a.com Email: liaoku_saope@hotmail.com

Fuente: Ivan Eduardo Martínez Merino

Última actualización: 03/17/2018 21:40:00

Autor: Ivan Martínez

Cancelar

Después

Descripción: Razonamiento:

Se deben poder crear actividades asociadas al cumplimiento de una característica

Gestionar más atómicamente la realización y el cumplimiento de un requisito

Información

Creado por: Ivan Martínez CEO a.com Email: liaoku_saope@hotmail.com

Fuente: Ivan Eduardo Martínez Merino

Última actualización: 03/17/2018 21:40:01

Autor: Ivan Martínez

Rechazar Aceptar

Figura 4.25: Solicitud de Cambio

En la Figura 4.25, los campos que han sufrido cambios son resaltados con un borde rojo, esto tiene como finalidad poder realizar un rápido diagnóstico, permitiendo detectar rápidamente los cambios y así finalizar lo antes posible, dejando a criterio *aceptar* o *rechazar* los cambios mediante los botones localizados en la esquina inferior derecha.

- **Historial de cambios:** Aquí es posible visualizar todos los cambios realizados en el proyecto, desde la creación de características, hasta cualquier modificación realizada en ellas, mediante una línea de tiempo, que contiene esta información en forma de hitos, tal como se ve en la Figura 4.26:

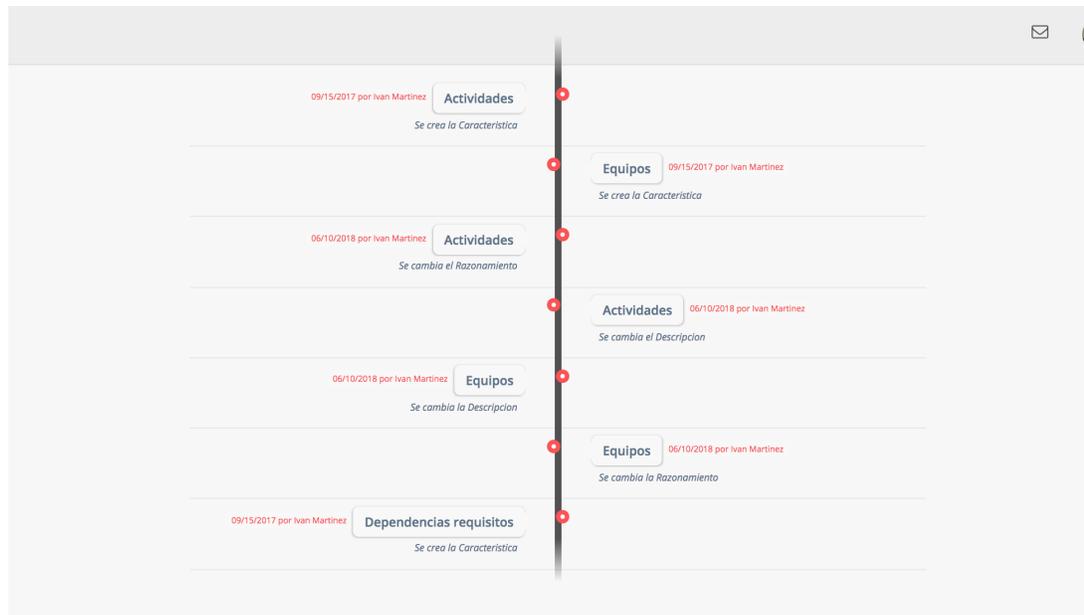


Figura 4.26: Línea de tiempo con cambios realizados en un proyecto

5. Evaluación

5.1. Análisis de perfil de usuario

El sistema esta pensado para que usuarios de todo tipo puedan interactuar con ella, bajo este objetivo, definiremos "todo tipo de usuario", como un desarrollador, jefe de proyectos, cliente o interesado (Stakeholder), los cuales pueden interactuar en un proyecto dentro de este sistema de forma colaborativa.

Es por ello que se han seleccionado perfiles muy específicos para el testeo del sistema:

- Perfil 1: Ingeniero comercial, dueño de una consultora que presta servicios tecnológicos a servicios públicos de la república de Chile, perfil que ha visto triunfar y fracasar innumerables proyectos de software, por muchísimas razones, una de ellas es debido a la falta de control en el avance y el nulo entendimiento de los requisitos del cliente.
- Perfil 2: Ingeniero Eléctrico, dueño de una consultora que presta servicios tecnológicos a empresas mineras, antiguo dueño de una empresa desarrolladora de software, corresponde a un perfil con alto conocimiento técnico en gestión de proyectos.
- Perfil 3,4,5: Para perfiles de desarrolladores, se escogió a tres ingenieros civiles en computación, uno con cinco años de experiencia en el mercado y quien tendrá el papel de Developer Owner, lo que significa que es quien crea el proyecto e ingenieros civiles en computación de un año de experiencia y un memorista quienes formarán parte del equipo del proyecto como Developers.

Claramente los perfiles escogidos tienen un propósito específico, perfiles 1 y 2 actuarán como stakeholder en esta interacción, permitiendo a los perfiles 3, 4 y 5 para gestionar el

proyecto. Los niveles Alto, Medio y Bajo de los perfiles desarrolladores permite hacer una evaluación transversal en el sistema, y los perfiles senior para los stakeholder permiten una mirada desde la experiencia de los proyectos, el perfil 2 además tendrá acceso total al sistema para evaluar todo el funcionamiento y el perfil 1 podrá ver el resultado generado para una evaluación del resultado sobre el camino con una avance de 2 semanas. Toda esta libertad para la revisión del sistema se da para mitigar el hecho de que es imposible crear desde cero un proyecto real y llevarlo hasta el final por las siguientes razones:

- Tiempo: un proyecto tan pequeño como para poder ser utilizado dentro de los tiempos disponibles está fuera del alcance del sistema, ya que no tendría ningún sentido utilizar el software ProjectMNG en algo tan pequeño.
- Estado del software: El sistema, debido a que está en proceso de desarrollo, se trata sólo de un beta que esta en proceso de testeo, lo que no permite que alguien tome el riesgo de gestionar el proyecto por medio de este sistema, ya que no garantiza su buen funcionamiento en el tiempo.

5.2. Definición de los objetivos de usabilidad

Los objetivos que se buscan cumplir dentro del sistema están definidos con detalle en el Anexo A en forma de historias de usuario, bajo estas definiciones podemos hacer un resumen de los objetivos del sistema, que son los que utilizamos para realizar las pruebas de usuario posteriormente:

- Usuarios: Dentro de este apartado se desea verificar la usabilidad de:
 - Usuario Stakeholder: Este tipo de usuario debe cumplir los siguientes objetivos:
 - Creación de usuarios stakeholder
 - Interactuar con Características (Solicitar y aprobar cambios)
 - Usuario Developer:
 - Registrar nuevo usuario
 - Interactuar con Características (Crear, Eliminar, Visualizar y modificar)
- Proyectos:

- Interactuar con Proyecto (Crear, Eliminar, Visualizar y modificar)
- Interactuar con los equipos (Crear, Eliminar, Visualizar y modificar)

Para los objetivos definidos anteriormente se miden por medio de los siguientes puntos enfocados a la usabilidad:

- Fácil aprendizaje y recordar: Este punto es medido durante la introducción al sistema y posterior uso, a través de la comunicación directa con los perfiles, quienes podían hacer preguntas del sistema durante el uso del mismo.
- Satisfacción: La medición de este aspecto hace por medio de la encuesta realizada.
- Eficacia: Este punto se mide por medio del cumplimiento del flujo de trabajo definido en la Figura 5.8
- Eficiencia: Este apartado es medido por medio de los resultados obtenidos por el uso limitado del sistema y la opinión directa de los perfiles, el hecho de que la prueba no sea de un ciclo completo de desarrollo afecta directamente a este punto, sin embargo se obtiene una retroalimentación por parte de los usuarios, lo que afecta directamente el punto anteriormente mencionado.

5.3. Implementación de prototipos

El primer prototipo se implementó luego de cumplir el primer ciclo de programación definido en la Figura 4.3 ,después de cumplir 8 sprint, este desarrollo fue gradualmente construido siguiendo las prioridades dadas por la lista del Anexo A.

Para esta primera implementación de prototipo se utilizó el servicio de GCP (Google Cloud Platform), en el cual se levantó una maquina virtual, también se consideró la opción de utilizar las Cloud Function de GCP, pero se descartó debido al tamaño del proyecto (11 MB), lo que obligaba a utilizar Cloud Storage para almacenar los archivos, lo que implica actualizar todas las referencias a archivos, provocando una actualización de código muy extensa. Entonces, se inicia una maquina con requerimientos mínimos corriendo con Windows Server 2012, se configura la entrada y salida http (:80), se descarta https (:443) para el prototipo.

Se instala MySql Server y .Net Core 2.0 para poder correr el proyecto en IIS sin problemas.

La razón de utilizar GCP en vez de AWS o Azure, va relacionado con los 300 USD gratuitos que entrega Google para utilizar su plataforma.

Para facilitar la puesta en marcha del proyecto, se instala en la máquina visual studio code, para que al inicializar el proyecto instale todos los paquetes necesarios con las versiones correspondientes.

Para el primer prototipo se bloquea la opción de registro y se crean cuentas para todos los sujetos de prueba con sus perfiles correspondientes.

Tampoco se requería comprar un dominio, ya que para el ejercicio de testing no era relevante, por lo cual se le facilitó a todos los sujetos de prueba la IP correspondiente al servidor para utilizar el servicio.

5.4. Realización de pruebas

Debido a la metodología aplicada, las pruebas funcionales se realizaron gradualmente en cada sprint, no obstante existe un flujo que claramente debe ser cumplido para poder declarar el sistema listo para ser utilizado, teniendo en cuenta que las pruebas se están realizando en un beta del programa, este flujo representa el objetivo principal de Project-MNG.

El flujo utilizado y que cada perfil tuvo que cumplir, es el siguiente:

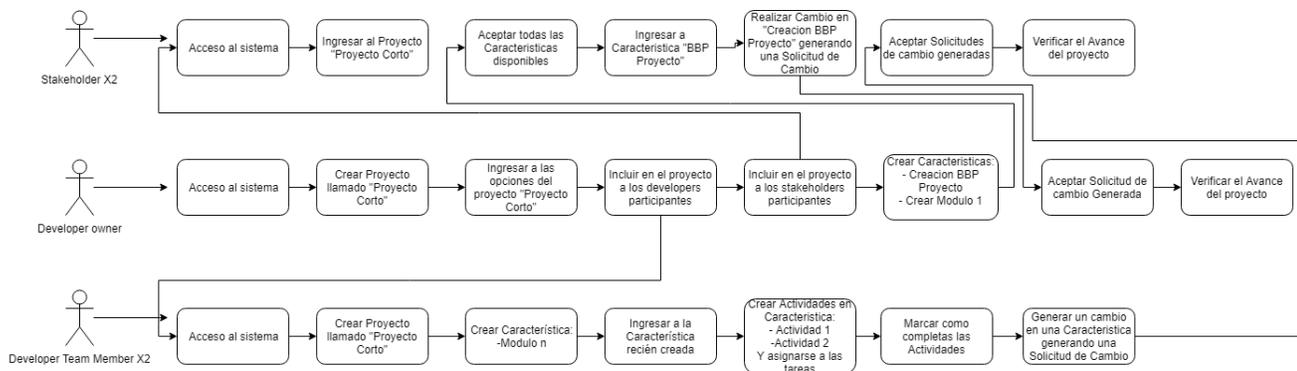


Figura 5.1: Flujo de trabajo

Cada perfil debía cumplir con cada una de sus tareas asignadas, a cada quien se le capacitó personalmente para que pudieran realizar cada una de sus tareas acompañado del tutorial presentado en el Anexo D, en caso de tener dudas podían transmitir sus dudas en persona o vía telefónica, situación que no ocurrió, probablemente debido a que el sistema

posee solo las opciones principales para su funcionamiento, lo cual permite claridad en su funcionamiento a simple vista, además de que cada tarea se explica por sí sola a través de las pantallas. Por medio de estas pruebas se logra probar que se puede crear una interacción fluida por medio de solicitudes y notificaciones de usuario, este canal de comunicación resulta muy efectivo debido a que los pendientes, dudas y cambios no se pierden, siendo incorporados dentro de un flujo de trabajo, todo esto se puede ver con más detalle en la Sección 5.9.

5.5. Rediseño

luego de la evaluación de los usuarios y de la ejecución de las pruebas uno de los grandes cambios realizados en la plataforma, resultado de la evaluación, es el rediseño en la visualización de las características, que podía llegar a convertirse en una lista interminable de ítems, esto es visible en la Figura 4.21.

Por esa razón se plantea crear una nueva forma de visualización resumida que muestre sólo los nombres y utilice menos espacio, como sugieren los usuarios, pero que se respeten los colores de la aplicación:

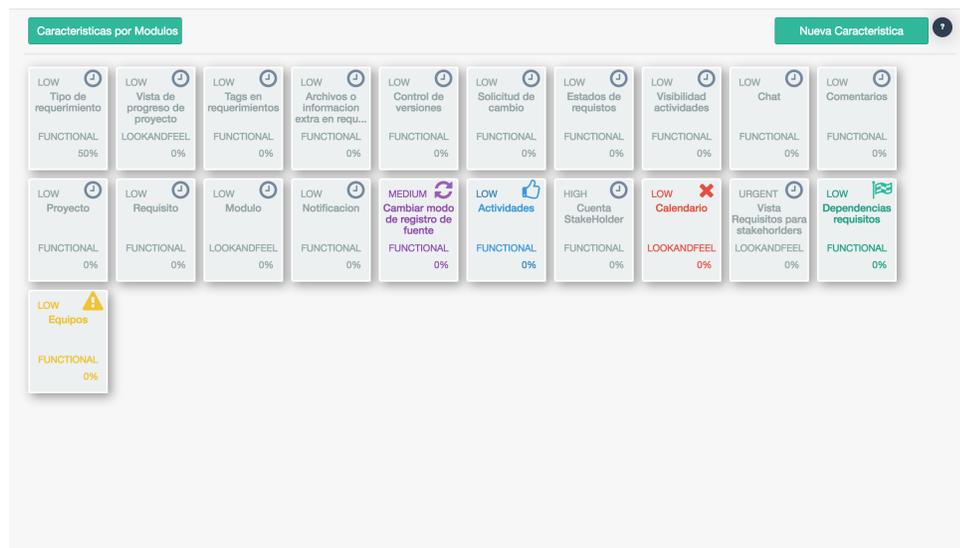


Figura 5.2: Listado de características luego del rediseño

El nuevo diseño de los elementos no sólo optimiza el espacio, sino que es capaz de mostrar más información, es más versátil, logrando el objetivo planteado luego de la evaluación, mejorando notoriamente la experiencia de la plataforma para entornos móviles, un ejemplo de esto está en la Figura A.1:

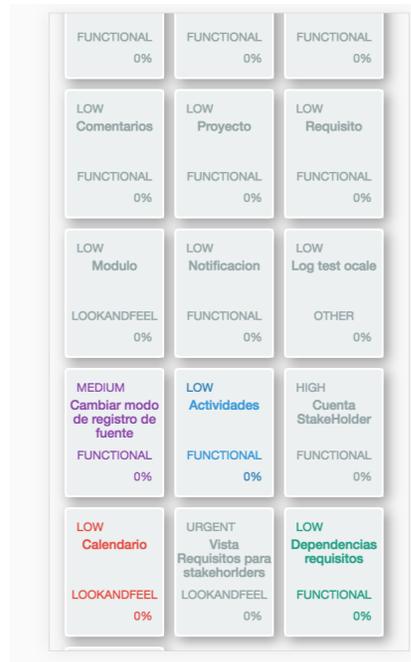


Figura 5.3: Visualización móvil de las características

Las formas de visualización antiguas ya no estarán disponibles, pese a que los usuarios resaltaban que no era necesario reemplazarla, luego de ver la nueva forma de visualizar las características, se concluye que la vista anterior es totalmente innecesaria y redundante.

La nueva forma de visualizar las características genera gran satisfacción en general debido a que suple con creces cualquier problema que presentaba la antigua forma, dándole también otro aspecto al sistema, mucho más homogéneo, lo que no sólo provoca un agrado a nivel funcional, sino que también a nivel visual y como se ve a continuación la nueva versión móvil es notoriamente un triunfo sobre la versión anterior, desde la forma de visualización de características, pudiendo ver a la vez hasta doce características, las cuales contrastan muchísimo en comparación a las dos anteriores.

En cuanto a otros resultados de las encuestas, muchas de estas se trataban de bugs en la plataforma o desconocimiento del funcionamiento de ésta, para solucionar la segunda, se proporcionaron ítems de ayuda en cada una de las vistas que proporcionan un pequeño

resumen de funcionamiento y permite guiar rápidamente al usuario:

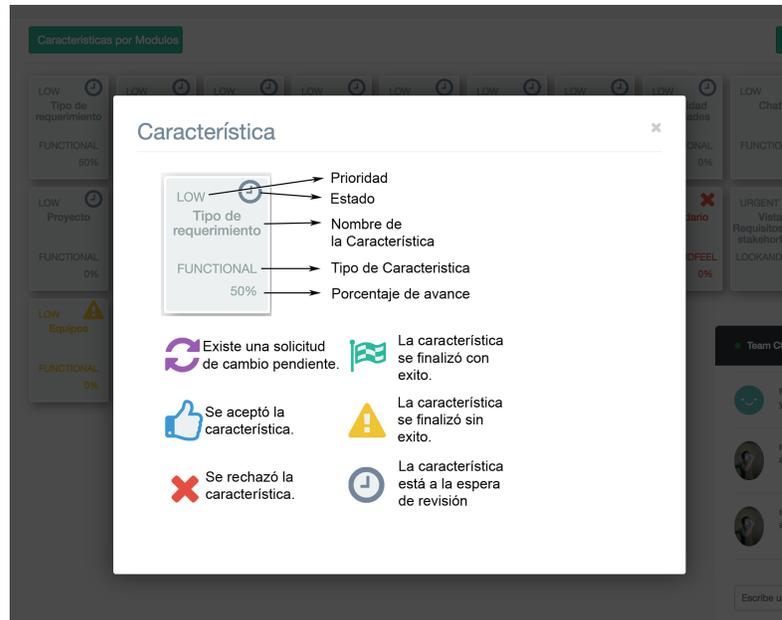


Figura 5.4: Visualización móvil de las características

En cuanto al primer punto, a continuación se muestra el listado de todos los bugs que fueron solucionados:

- Eliminar, reprobado y aprobado característica no genera notificación.
- Agregar y eliminar usuarios a un equipo no genera notificación.
- Se pueden eliminar tareas de cualquier proyecto, sin necesidad de pertenecer a él.
- Agregar patrocinadores en detalle característica no se comporta correctamente.
- Validación de datos en agregar usuarios.
- El campo para agregar miembros a un equipo tiene un correo por defecto.

En su mayoría son bugs de validación a nivel de vista, pero esto no provocó mayores inconvenientes ya que todas las validaciones de ingreso de datos están delimitadas en los controladores, lo cual da mayor robustez y en general es lo que se espera a nivel de seguridad como mínimo, pero no generaba ninguna alerta ni cuando se registraban los datos con éxito ni cuando fallaba, esto provocaba confusión en el usuario dando

la impresión de que el sistema estaba fallando, por tanto se implementan todas las alertas correspondientes como se puede ver a continuación en la Figura 5.5:

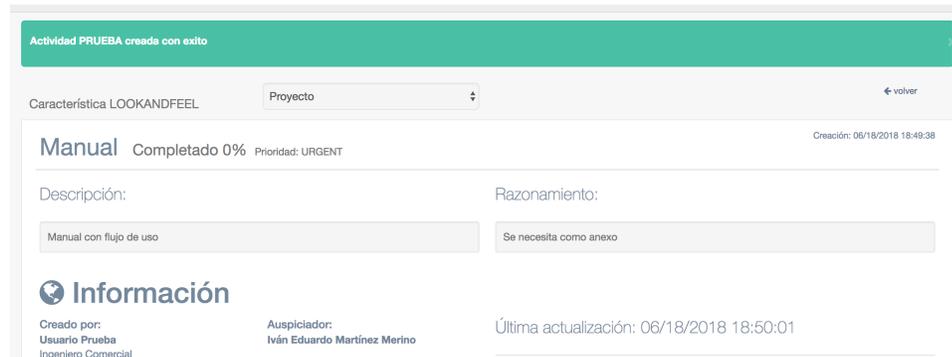


Figura 5.5: Alerta de éxito al crear una actividad

5.6. Implementación del producto o servicio

El producto o servicio no está pensado para ser comercializado, fue desarrollado bajo una clara necesidad identificada en la ejecución de proyectos de software para empresas mientras se trabajaba como freelance, no obstante para utilizar el software se necesita de al menos una maquina n1-standard-1 (1 vCPU, 3,75 GB de memoria), es suficiente para correr IIS con el proyecto y la base de datos MySQL sin problemas, para este release si es necesario la configuración https, así como también queda pendiente para esta puesta en marcha la encriptación de cookies y datos de usuario, que claramente no afectan en nada su funcionalidad, pero si afectan en gran medida la seguridad de la aplicación.

En el proceso de uso del servicio, se incluye una capacitación con el cliente, en la cual se hará una inicio del proyecto y se le presentarán todas las características de éste y cómo interactúa con él, para luego guiarlo en el tiempo por correo hasta que el uso de la herramienta sea frecuente.

5.7. Retroalimentación del usuario

La retroalimentación de los usuarios fue directa por parte de los cinco perfiles mediante la encuesta realizada, cuyos resultados fueron obtenidos de las evaluaciones realizadas a usuarios, cabe recordar que estas pruebas son dirigidas a usuarios seleccionados de acuerdo a qué funcionalidad del sistema utilizarán, tal como se menciona en la Sección 3.2.

El detalle de la encuesta obtenida puede ser vista en el Anexo C.

El rango de evaluación son 0 que significa *muy insatisfecho*, 1 *insatisfecho*, 2 *poco satisfecho*, 3 *neutral*, 4 *satisfecho*, y 5 en que significa *muy satisfecho*.

Lo resultados muestran que la eficacia de los usuarios al usar la plataforma es positiva, en relación a la funcionalidad de las distintas partes del sistema:

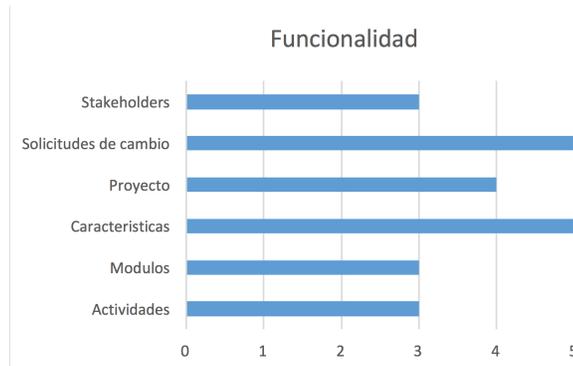


Figura 5.6: Resultados de satisfacción de la plataforma a nivel funcional

En cuanto a los resultados de la interfaz de usuario, que busca medir la satisfacción, se muestran resultados diferentes, aunque no alejados a los de la Figura 5.9:

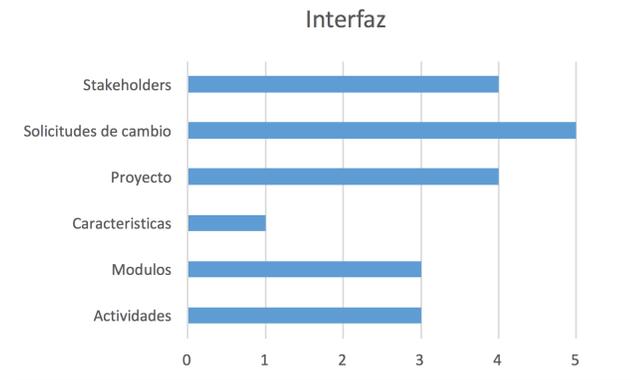


Figura 5.7: Resultados de satisfacción de la plataforma a nivel interfaz

Recordemos que las encuestas no fueron realizadas a personas al azar, con el fin de obtener mejores resultados, por lo tanto, estos resultados promedio, han sido ponderados de acuerdo a la importancia del encuestado en cada sección del sistema.

La visualización de las características es el problema más claro y es recurrente en todos los usuarios, sin importar el rol que cumplan dentro de la plataforma.

5.8. Aplicación

A continuación se enumera y describe la aplicación de cada uno de los pasos de la devolución realizada:

- **Análisis del perfil de usuario:** Al momento de analizar el perfil de los usuarios se consideró su profesión y experiencia en algún rubro, desde un ingeniero eléctrico que trabajó años en la industria del software, hasta una ingeniero comercial con años de experiencia en la consultoría para el sector público.
- **Definición de los objetivo de la usabilidad:** Para este punto se creó un flujo de trabajo en el cual podían interactuar entre ellos, con el fin de lograr varios objetivos durante el proceso. El flujo de trabajo se puede observar en la Figura 5.8. y los objetivos de usabilidad a cumplir son los siguientes:
 - Acceso al sistema.

- Creación de Proyecto.
 - Creación de Característica.
 - Inclusión de Stakeholder al proyecto.
 - Interacción con la Característica.
 - Generar Solicitud de cambio.
 - Interacción de Solicitud de cambio.
- Realización de test: El test está dentro del flujo ya que son encuestas que nacen durante el uso del software, las preguntas realizadas se encuentran en el Anexo B
 - Implementación de prototipos: Se analizó la información proporcionada por el usuario y se planificaron los cambios necesarios.
 - Implementación de producto o servicio: Se implementaron los cambio, esto puede ser visto en más detalle en la Sección 5.6.
 - Retroalimentación del usuario: Una vez realizados los cambios, estos fueron presentados a los usuario con el fin de conocer su opinión y si estos problemas eran solucionados, éste paso podría haber continuado dentro de la iteración que la metodología de evaluación permite, pero para este caso solo se hizo una iteración ya que los mayores comentarios del sistema fueron de forma visual más que funcional.

5.9. Resultados

Luego de que los usuarios encargados de testear el sistema siguieran el siguiente flujo de trabajo 5.8:

Luego de completar el flujo de trabajo los cinco usuarios de estas pruebas fueron libres de realizar lo que consideraran necesario dentro de la plataforma, estas pruebas prolongaron por siete días, en donde todos interactuaron dentro de un proyecto, sin conocimientos el uno del otro más que su foto de perfil.

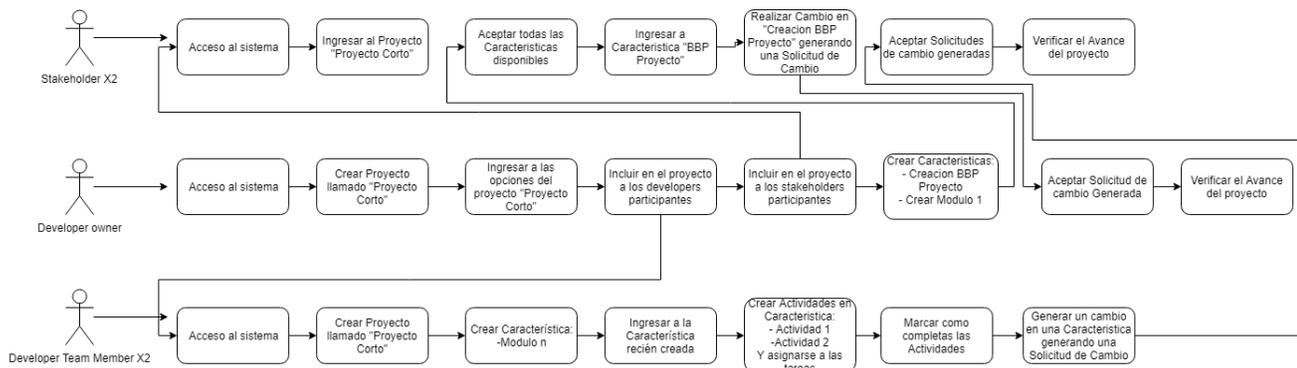


Figura 5.8: Flujo de trabajo

En cuanto a los perfiles de stakeholders, los dos usuarios seleccionados son un ingeniero eléctrico, gerente general de una empresa de tecnología que presta servicios a minería de 55 años y una ingeniero comercial, socia fundadora de una consultora que presta servicios al sector público de 35 años.

Para los perfiles desarrolladores los tres usuarios seleccionados son un ingeniero civil en computación con tres años de experiencia en gestión de proyectos de software, 33 años, el siguiente perfil es de un ingeniero informático con 1 año de experiencia en desarrollo de software, 26 años y por último un memorista de ingeniería civil en computación.

A continuación, se presentan los resultados obtenidos de las evaluaciones realizadas a usuarios, cabe recordar que estas pruebas son dirigidas a usuarios seleccionados de acuerdo a qué funcionalidad del sistema utilizarán, tal como se menciona en la Sección 3.2.

El detalle de la encuesta obtenida puede ser vista en el Anexo C.

El rango de evaluación son 0 que significa *muy insatisfecho*, 1 *insatisfecho*, 2 *poco satisfecho*, 3 *neutral*, 4 *satisfecho*, y 5 en que significa *muy satisfecho*.

Los resultados muestran que la satisfacción de los usuarios al usar la plataforma es positiva, refiriéndonos a la funcionalidad de las distintas partes del sistema:

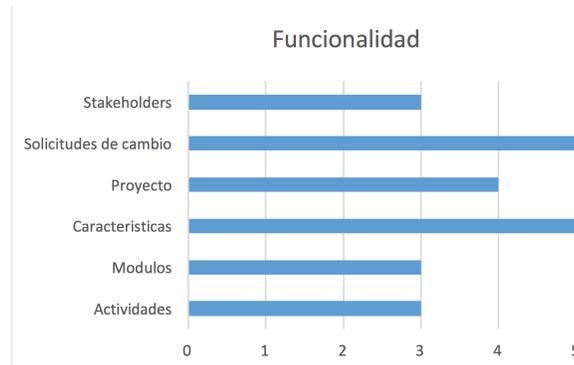


Figura 5.9: Resultados de satisfacción de la plataforma a nivel funcional

En cuanto a los resultados de interfaz de usuario, se muestran resultados diferentes, aunque no alejados a los de la Figura 5.9:

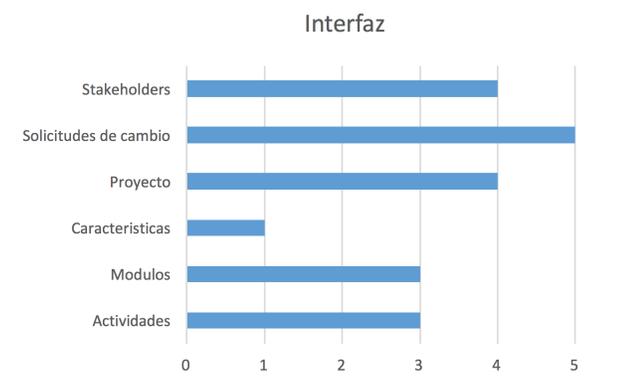


Figura 5.10: Resultados de satisfacción de la plataforma a nivel interfaz

Recordemos que las encuestas no fueron realizadas a personas al azar, con el fin de obtener mejores resultados, por lo tanto, estos resultados promedio, han sido ponderados de acuerdo a la importancia del encuestado en cada sección del sistema.

La visualización de las características es el problema más claro y es recurrente en todos los usuarios, sin importar el rol que cumplan dentro de la plataforma.

Luego de una breve consulta a los usuarios, quedó claro que el problema es la disposición de los elementos, ya que al ser muy grandes, no se logra la visualización de una cantidad útil de elementos a la vez.

Al sugerir a los usuarios reemplazar esta vista por una capaz de mostrar más elementos, estos no se mostraron conformes, puesto que la actual forma de visualizar es útil debido a sus ayudas visuales (Colores e Iconos), aunque en la visualización móvil no logra el objetivo de ninguna manera debido a que no se visualizan más de tres elementos a la vez:



Figura 5.11: Listado de características visto desde un móvil

Como se observa en la Figura 5.11 tres elementos es la máxima cantidad de elementos disponibles, lo que genera un gran problema y pérdida de cualquier visualización general.

Debido a esto, se decide crear una nueva forma de visualización que satisfaga a la necesidad de ver una mayor cantidad elementos en un espacio reducido y que también mejore la visualización móvil de esta sección.

Otros problemas que se visualizan en la Figura 5.9 son: Módulos, Actividades, Stakeholders.

En el caso de Módulos, el resultado es que los usuarios no encontraron relevante esta funcionalidad, debido a que visualizar las antiguas características más esta opción era molesto, lo cual queda solucionado resolviendo el problema de visualizar características.

El problema que encontramos en Actividades, es que no alerta cuando éstas están a punto de expirar, esto sólo es visible en la vista de actividades, pero esta funcionalidad queda para trabajos futuros en la plataforma.

La visualización de Stakeholders, suponía un problema de organización, ya que las características desplegadas en esta vista, no eran coherentes entre su ordenamiento y su visualización (colores), creando confusión a simple vista.

5.10. Objetivos

A continuación se enumerarán los objetivos y se darán detalles de cómo fueron abordados para posteriormente ser cumplidos:

- Canal de comunicación entre desarrolladores y stakeholders: Se logró este objetivo, ya que los sujetos de pruebas lograron interactuar entre ellos a través del sistema, pudiendo incluso crear un flujo de prueba que implicaba estrictamente tener que interactuar para poder avanzar.
- Herramienta simple de especificación y gestión de proyectos: El sistema simple de especificación y gestión de proyectos se logra debido a que no es una invención creativa, sino que se utilizan funcionalidades y formas de interacción de sistemas de especificación y gestión de proyectos ya creadas y exitosas, el objetivo de este proyecto a largo plazo es lograr todas las funcionalidades y capacidades que la competencia ya entrega, integrando las funcionalidades de comunicación que posee ProjectMNG.
- Herramienta para que stakeholder participen en su proyecto: Este objetivo es logrado y claramente visible, ya que el plan de pruebas para los usuarios requiere estrictamente que los stakeholder participen en un proyecto, de no lograrse las pruebas no habrían concluido y no se hubiera tenido la información de los test.
- Sistema web: Todo el proyecto fue testeado a través de un sistema web, que además es presentado en el Anexo D A modo de tutorial.

6. Conclusiones

El desarrollo de aplicaciones mediante metodologías ágiles tan extremas como lo es PXP, conlleva un desafío lleno de retos, realizando continuos análisis, re-planificaciones, modificando definiciones y adaptando constantemente todo el desarrollo con tal de lograr productos que pudieran ser testeados durante el proceso y no perder demasiado tiempo.

A pesar del tiempo limitado y el alcance del proyecto, este logró llegar a buen puerto, todo esto, gracias a seguir rigurosamente la metodología de desarrollo y los resultados de la evaluación, que permitió realizar cambios que eran muy necesarios para mejorar la experiencia del usuario y que al final terminaron siendo cambios críticos que modificaron la forma de ver al sistema.

ProjectMNG como software de planificación de proyectos, aunque es un tema complejo, no necesito de una verificación exhaustiva debido a que los software de planificación de proyectos es algo que hoy en día tienen mucho recorrido, lo que permitió limitar esta sección del desarrollo a imitar características de exponentes más consolidados en este aspecto, logrando cumplir con la funcionalidad esperada.

La comunicación que se busca crear entre los desarrolladores y stakeholders es un mecanismo muy simple de verificación y control de cambio, que es muy difícil de verificar, debido a que la única forma formal de hacer esto, es probarlo en un desarrollo real, lo cual por cuestiones de tiempo es imposible, pero aun así se verifica la satisfacción del usuario en cuanto a funcionalidad e interfaz al utilizar un período de tiempo esta herramienta de comunicación en un ambiente simulado, obteniendo buenos resultados, todo esto debido a que el diseño de la solución está inspirada en experiencias de desarrollos reales.

6.1. Trabajo futuro

ProjectMNG tiene mucho potencial de crecimiento, su diseño permite incluir nuevas funcionalidades con relativa facilidad, tomando en cuenta que todas sus características y atributos fueron construidas para crecer en un futuro, se puede ver en el desarrollo del sistema una gran cantidad de definiciones que no son utilizadas, esto se debe a que están ideadas para funcionalidades futuras, como: distintas formas de filtrar información, capacidad de guardar Características en Favoritos, calendarios interactivos, modo *daltónico*, etc.

El crecimiento de ProjectMNG, va de la mano con la inclusión de nuevas y mejores formas de comunicación entre desarrolladores y stakeholders, las que incluyen straming, pizarras interactivas, mejoras en los chats, crear nuevas alertas, etc. Elementos que podrían mejorar la experiencia de ambas partes, y no sea necesario el uso de herramientas externas.

Glosario

Stakeholder: Se refiere a cualquier persona que tenga algún interés en el proyecto, en este caso particular llamaremos stakeholder a cualquier persona dentro del proyecto, que no sea desarrollador, en ese sentido esto incluye clientes, usuarios, key users, etc.

Desarrollador: Nos referiremos como desarrollador a cualquier persona a cargo de llevar a cabo un proyecto.

Ingeniería de software: Es la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software y el estudio de estos enfoques, es decir, el estudio de las aplicaciones de la ingeniería al software.

Extreme programing: Es una metodología de desarrollo de la ingeniería de software formulada por Kent Beck, autor del primer libro sobre la materia, Extreme Programming Explained: Embrace Change [3]. Es el más destacado de los procesos ágiles de desarrollo de software.

PXP: Es una metodología de desarrollo basada en Extreme Programing pero modificado para ser utilizado por una sola persona.

Framework: Entorno de trabajo o marco de trabajo. Es un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.

Proyecto: Idea de algo que se desea realizar y en la cual se establece un modo determinado y un conjunto de medios necesarios.

Planificar: Elaborar o establecer el plan conforme al que se ha de desarrollar algo, por ejemplo un proyecto.

Solicitud de cambio: Es la petición que se hace para poder llevar a cabo un cambio.

Control de cambio: Es realizar el control integrado de cambios de proyectos, pasa por el proceso de supervisar y controlar los trabajos realizados y llevar un registro de ello.

Responsivo: Hace referencia al diseño web adaptable cuyo objetivo es adaptar la apariencia de las páginas web al dispositivo que se esté utilizando para visitarlas.

Streaming: Es la distribución digital de contenido multimedia a través de una red.

DBA: El DataBase Administrator es el responsable de los aspectos técnicos, tecnológicos, científicos, inteligencia de negocios y legales de bases de datos, y de la calidad de datos.

Característica: Es el nombre que se le da a las definiciones atómicas, que se pueden hacer de un proyecto dentro de la plataforma ProjectMNG.

Actividad: Es una acción que debe ser llevada cabo, para concretar una característica.

Comunicación bilateral: Es el acto comunicativo en donde el emisor emite un mensaje, el receptor lo decodifica, y responde ahora convirtiéndose ahora en emisor.

Comunicación directa: Es cuando la comunicación no tiene intermediarios innecesarios de por medio.

Bibliografía

- [1] Ravikant Agarwal and David Umphress. Extreme programming for a single person team. *Proceedings of the 46th Annual Southeast Regional Conference on XX - ACM-SE 46*, (January 2008):82, 2008.
- [2] J.A. Aloy, M.R. Delgado, and X.R. Perez. Editorial UOC, 2014.
- [3] Kent Beck and Cynthia Andres. *Praise for Extreme Programming Explained*. 1999.
- [4] Perurena Cancio. Revista Cubana de Información en Ciencias de Usabilidad de los sitios Web , los métodos y Usability of Web sites , methods and evaluation techniques. 24(2013):24, 2018.
- [5] Marcos Catalán . Metodologías de evaluación de interfaces gráficas de usuario. *EPrints in Library and Information Science*, page 138, 2000.
- [6] Justine Desroches, Pier Alexandre Champagne, Yasmine Benhassine, Jean-François Paquin, Michael Lukacs, and Dipasis Bhadra. Choosing between .NET Core and .NET Framework. *Org. Biomol. Chem.*, 13(8):2243–2246, 2015.
- [7] Keil Mark. Pulling the Plug: Software Project Management and the Problem of Project Escalation. *MIS Quarterly*, 19:27, 1995.
- [8] Oracle. Top 10 Reasons to Choose MySQL for Next Generation Web Applications: A MySQL Whitepaper. (August):3, 2016.
- [9] Roger Pressman. *Ingeniería del software. Un enfoque práctico*. Editorial McGraw-Hill, septima edition, 1998.

ANEXOS

A. Historias de usuarios

■ DESARROLLADOR

- Como desarrollador quiero poder crear proyectos.
- Como desarrollador quiero administrar los requisitos y/o historias de usuario de mis proyectos.
- Como desarrollador quiero poder crear actividades relacionadas a estos requisitos y/o historias de usuario para el cumplimiento de estas.
- Como desarrollador quiero un calendario donde visualizar estas actividades.
- Como desarrollador quiero poder agregar gente a mi equipo de desarrollo de un proyecto.
- Como desarrollador quiero poder organizar mis requisitos y/o historias de usuario en módulos.
- Como desarrollador quiero poder obtener retroalimentación por parte de los stakeholders sobre el proceso de especificación y desarrollo de un proyecto.
- Como desarrollador quiero un chat para comunicarse en tiempo real con el o los stakeholders y un chat para el equipo.
- Como desarrollador quiero recibir notificaciones sobre los cambios así como también de las actividades que están asociadas a mí.
- Como desarrollador quiero poder crear una cuenta para stakeholders para que puedan estar al tanto del proyecto sin que puedan interferir directamente en él.

■ STAKEHOLDER

- Como stakeholder quiero poder ver el avance de mi proyecto
- Como stakeholder quiero poder hacer cambios durante la especificación y el desarrollo
- Como stakeholder quiero poder aprobar o desaprobar los avances.
- Como stakeholder quiero un chat directo con los desarrolladores.
- Como stakeholder quiero que se me notifique cuando algo pueda ser revisado.

A.1. Listado funcionalidades

■ Proyectos

- Visualización de avance
- Listar proyectos
- Tienen equipos de usuarios
- Historial de cambios

■ Características

- Listar para usuario o para stakeholder
- Solicitud de cambio
- Estados en Características
- Con dependencias
- Historial de cambios
- Comentarios entre el equipo de desarrollo

■ Usuarios

- Poder registrarse como usuario
- Poder crear cuentas temporales para stakeholders
- Stakeholder
- Vista de proyecto resumido
- Realizar solicitudes de cambio

- Solicitud de creación de requisitos

■ **Actividades**

- Asociar a una característica.
- Con tiempo de duración
- Visualizar en un calendario
- Notificaciones
- Notificaciones cuando seas agregado a un equipo.
- Notificaciones cuando alguien se integre a tu equipo.
- Notificaciones cuando exista una solicitud de cambio.
- Notificaciones cuando una actividad está por finalizar o retrasada.
- Notificaciones para stakeholder cuando algo está para ser revisado

A.2. Prioridad final de construcción con pesos

Características	-
CRUD	2
Listar para usuario o para stakeholder	6
Solicitud de cambio	8
Estados en Características	4
Con dependencias	4
Subir archivos	2
Historial de cambios	6
Comentarios entre el equipo de desarrollo	4
Usuarios	-
CRUD	2
Poder registrarse como usuario	4
Poder crear cuentas temporales para stakeholders	2
Proyectos	-
CRUD	2
visualización de avance	4
Listar proyectos	4
Tienen equipos de usuarios	4
historial de cambios	6
Stakeholder	-
Vista de proyecto resumido	4
Realizar solicitudes de cambio	4
Solicitud de creación de requisitos	6
Actividades	-
CRUD	2
Asociar a un requisitos	4
Con tiempo de duración	4
visualizar en un calendario	2
Notificaciones	-
Notificaciones cuando seas agregado a un equipo.	4
Notificaciones cuando alguien se integre a tu equipo.	4
Notificaciones cuando exista una solicitud de cambio.	4
Notificaciones cuando una actividad está por finalizar o retrasada.	4
Notificaciones para stakeholder cuando algo está para ser revisado	4

Figura A.1: Prioridad final de construcción con pesos

B. Encuestas

B.1. preguntas

- **Activity:**

- La ubicación de la información es correcta
- Logra interactuar con sus actividades de buena manera
- Logra visualizar sus actividades claramente
 - Cantidad de información
 - Colores
- Es suficiente la forma de filtrado de Actividades
 - Colores
 - Tipos de filtro
 - Cantidad de filtros

- **Module:**

- Logra visualizar todo lo necesario
- La ubicación de la información es correcta
- Logra interactuar con los elementos con facilidad
- Logra visualizar sus módulos claramente
- Logra organizar de buena manera las características
 - Colores

- Tamaño de las características
- Cantidad de información.

■ **Project:**

- La ubicación de la información es correcta
- Logra interactuar con los elementos con facilidad
- Logra visualizar sus proyectos claramente
- logra visualizar toda la información relevante a simple vista:
 - Tamaño de los miembros del equipo
 - espacio descripción del proyecto
 - disposición de las solicitudes de cambio
- Logra agregar fácilmente un miembro a su equipo
 - Esta opción es de fácil acceso
 - La forma de agregar un miembro
 - ninguna

■ **Characteristic:**

- Logra visualizar todo lo necesario
- La ubicación de la información es correcta
- Logra interactuar con los elementos con facilidad
- Logra visualizar sus características claramente
- Muestra las características relevantes a primera vista:
 - Organizar por las que falta validar
 - Organizar por prioridad
 - ninguna

■ **ChangeRequest:**

- Logra Visualizar el cambio:
- La ubicación de la información es correcta

- Logra interactuar con los elementos con facilidad
- Logra visualizar sus solicitudes de cambio claramente
 - Colores
 - información desplegada
 - ninguna

■ **Stakeholder:**

- La ubicación de la información es correcta
- Logra interactuar con los elementos con facilidad
- Logra visualizar sus características claramente
- logra Visualizar toda la información relevante a simple vista:

C. Resultados evaluación

A continuación, en la Tabla C.1 se muestran los resultados obtenidas de la evaluación, donde T1, T2, etc. Hacen referencia a las personas que realizaron las encuestas dentro del sistema.

En la columna final se muestra un promedio del resultado, que es lo que fue utilizado para los fines prácticos.

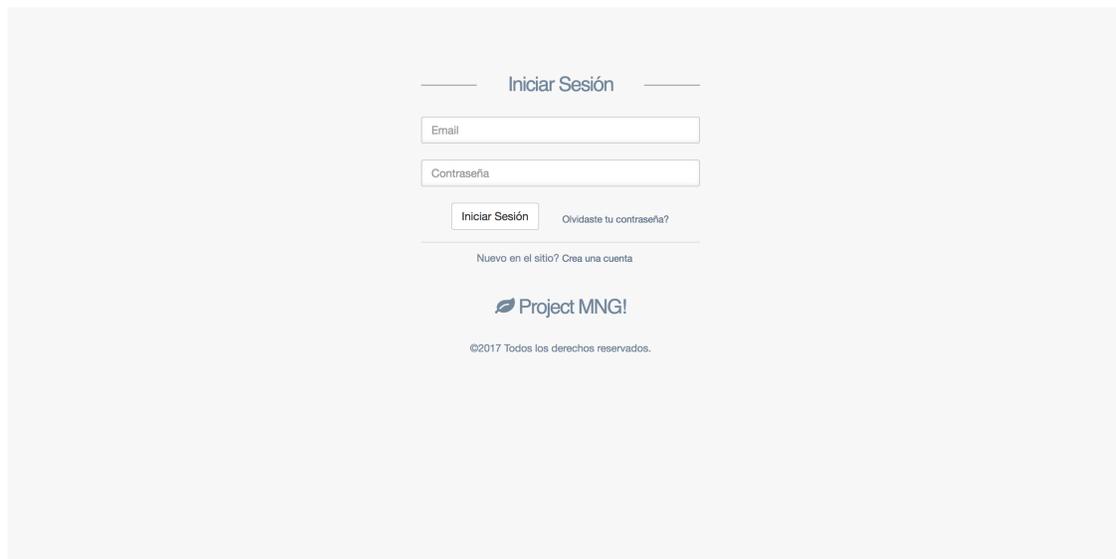
Pregunta	T1	T2	T3	T4	T5	Prom.
Activity						
La ubicación de la información es correcta	4	5	3	4	4	4
Logra interactuar con sus actividades de buena manera	4	3	2	3	3	3
Logra visualizar sus actividades claramente	4	3	4	4	4	3,8
Es suficiente la forma de filtrado de Actividades	3	3	2	2	3	2,6
Module						
Logra visualizar todo lo necesario	2	1	2	1	2	2
La ubicación de la información es correcta	4	3	4	4	4	3,8
Logra interactuar con los elementos con facilidad	4	4	4	4	4	4
Logra visualizar sus módulos claramente	4	4	5	4	4	4,2
Logra organizar de buena manera las características	3	2	4	3	3	3
Project						
La ubicación de la información es correcta	5	4	4	3	5	4,2
Logra interactuar con los elementos con facilidad	5	5	4	4	4	4,4
Logra visualizar sus actividades claramente	4	5	4	4	4	4,2
logra Visualizar toda la información relevante a simple vista	5	5	5	5	5	5
Logra agregar fácilmente un miembro a su equipo	4	4	3	4	4	3,8
Characteristic						
Logra visualizar todo lo necesario	1	1	2	2	1	1,4
La ubicación de la información es correcta	5	5	5	5	5	5
Logra interactuar con los elementos con facilidad	5	5	5	4	5	4,8
Logra visualizar sus actividades claramente	5	4	5	5	5	4,8
Muestra las características relevantes a primera vista	4	5	5	5	4	4,6
ChangeRequest						
Logra Visualizar el cambio	5	4	5	5	5	4,8
La ubicación de la información es correcta	5	4	5	5	4	4,6
Logra interactuar con los elementos con facilidad	5	4	5	5	5	4,8
Logra visualizar sus actividades claramente	5	5	4	4	5	4,6
Stakeholder						
La ubicación de la información es correcta	5	5	5	4	4	4,6
Logra interactuar con los elementos con facilidad	4	3	3	3	4	3,4
Logra visualizar sus actividades claramente	3	4	4	3	4	3,6
logra Visualizar toda la información relevante a simple vista	4	4	5	5	4	4,4

Cuadro C.1: Detalle de resultados de la encuesta de evaluación

D. Guía rápida

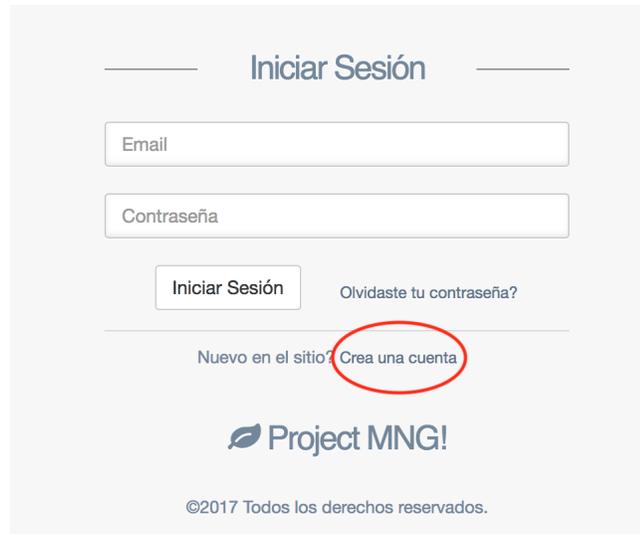
- **Creación de usuario:**

Al ingresar al sistema web ProjectMNG, usted encontrará la siguiente pantalla de inicio:



The screenshot shows a login page for Project MNG!. At the top, the text "Iniciar Sesión" is centered between two horizontal lines. Below this, there are two input fields: "Email" and "Contraseña". Under the "Contraseña" field, there is a button labeled "Iniciar Sesión" and a link that says "Olvidaste tu contraseña?". Below the input fields, there is a link that says "Nuevo en el sitio? Crea una cuenta". At the bottom, there is the Project MNG! logo and a copyright notice: "©2017 Todos los derechos reservados."

Para continuar, usted deberá presionar el botón *Crear una cuenta*:



A continuación, como primer paso, se le solicitará ingresar algunos de sus datos personales para comenzar con la creación de su cuenta de usuario:



El segundo paso en la creación de una nueva cuenta de usuario, solicita al cliente el ingreso de datos de seguridad de la cuenta:

1 Paso 1 Info. Personal

2 Paso 2 Info. Cuenta

3 Paso 3 Info. Profesional

E-mail * correo@falso.cl

Contraseña *

Repita la contraseña *

Finalizar **Siguiente** Anterior

El proceso de creación de una nueva cuenta, finaliza con el paso numero tres, donde se solicita al usuario el ingreso de datos acerca de su formación profesional:

1 Paso 1 Info. Personal

2 Paso 2 Info. Cuenta

3 Paso 3 Info. Profesional

Sitio Web * sitio.com

Compañía * company

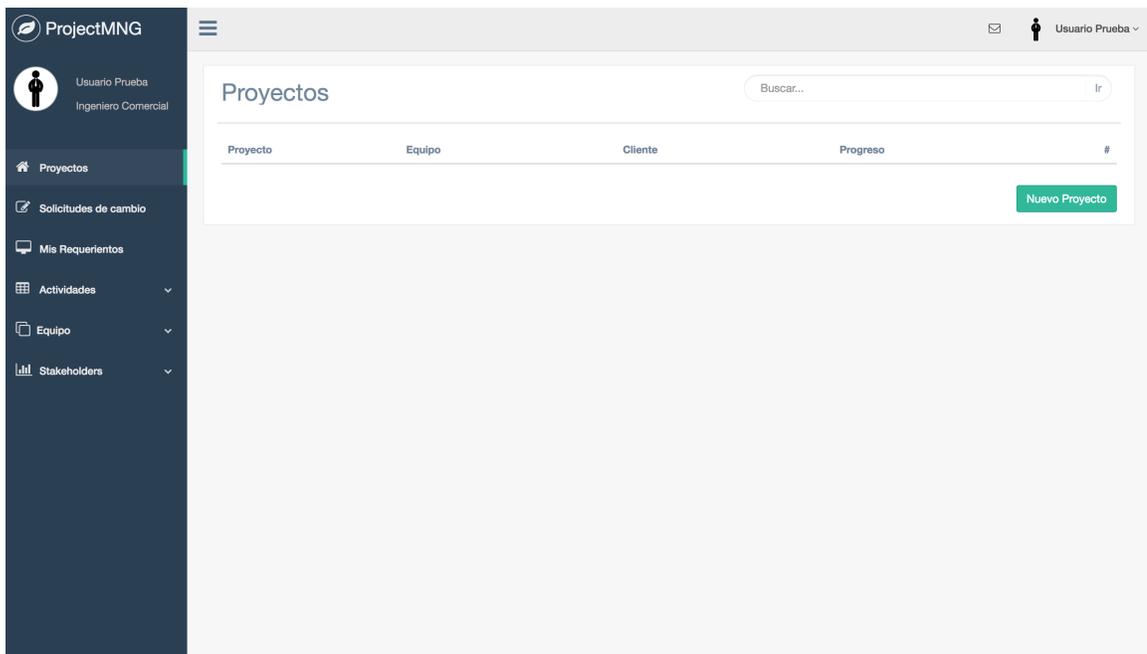
Título * Ingeniero Comercial

Finalizar **Siguiente** Anterior

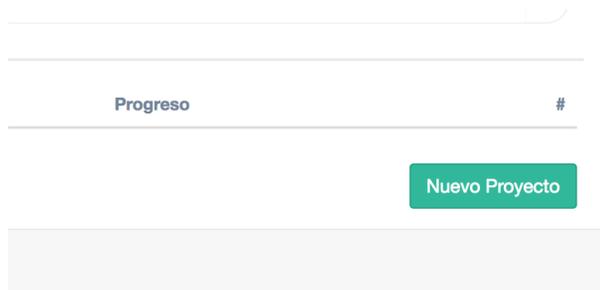
- **Creación de proyectos:**

Una vez finalizado con éxito el proceso previo, de creación de una nueva cuenta, el usuario tendrá acceso a una nueva pantalla, la cual contiene sus proyectos.

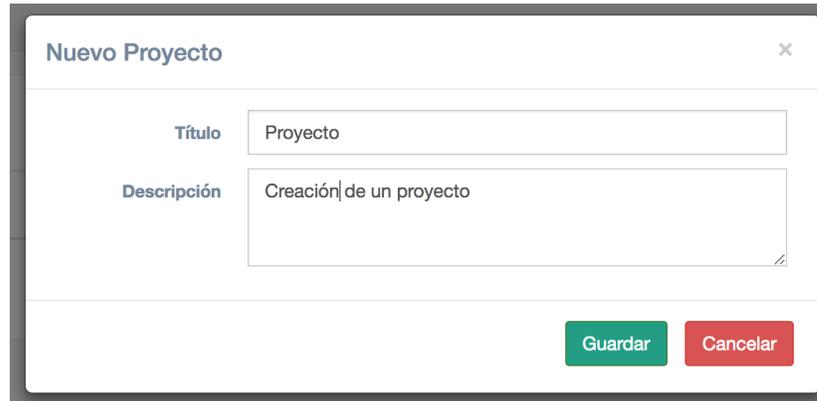
En la imagen a continuación, el usuario de prueba aún no cuenta con proyectos, ya que acaba de crear su nueva cuenta de acceso:



Para crear un nuevo proyecto, el usuario debe presionar el botón *Nuevo Proyecto*:

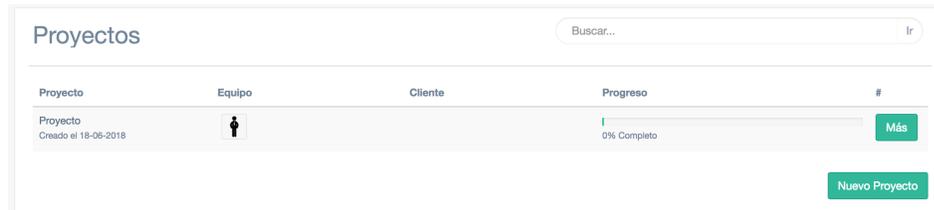


Al presionar el botón crear nuevo proyecto, el sistema solicita el ingreso de datos referentes al nuevo proyecto:



Formulario de creación de un nuevo proyecto. El formulario tiene un título "Nuevo Proyecto" y un botón de cerrar "x". Contiene dos campos de texto: "Título" con el valor "Proyecto" y "Descripción" con el valor "Creación de un proyecto". En la parte inferior derecha hay dos botones: "Guardar" (verde) y "Cancelar" (rojo).

Una vez realizado el ingreso de los datos solicitados por la pantalla anterior, el proyecto es creado y es posible visualizarlo en la ventana *Proyectos*:



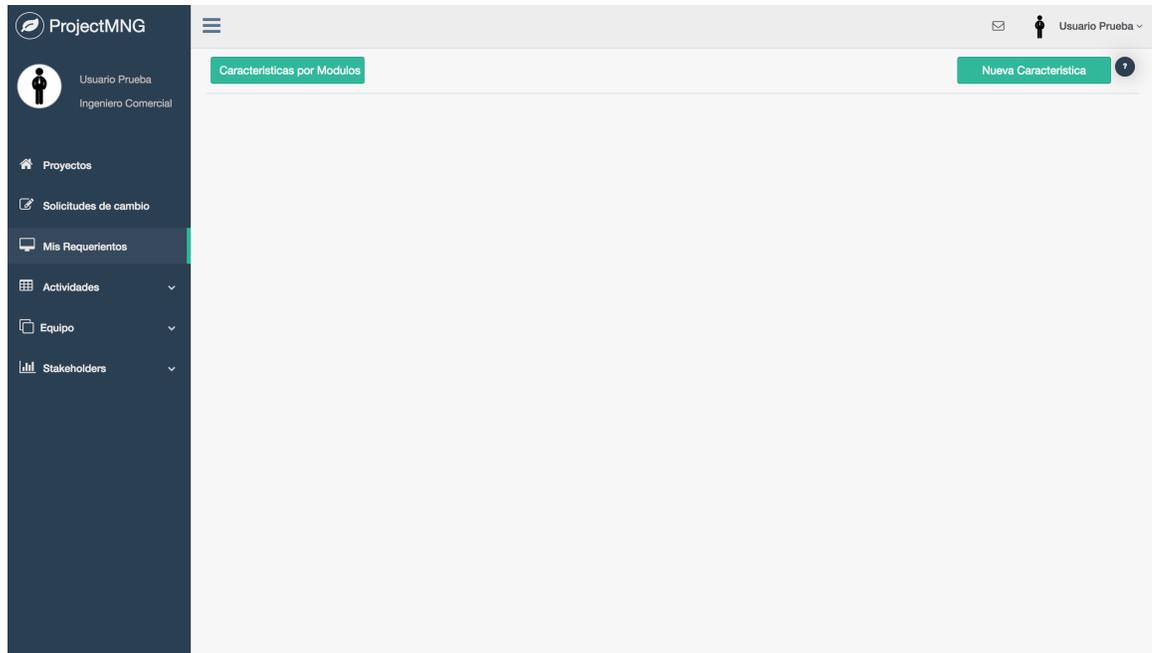
Vista de la lista de proyectos en la ventana "Proyectos". Incluye un campo de búsqueda "Buscar..." y un botón "Ir". La lista muestra un proyecto con los siguientes datos:

Proyecto	Equipo	Ciente	Progreso	#
Proyecto Creado el 18-06-2018			<div style="width: 10px; height: 10px; background-color: #ccc; display: inline-block;"></div> 0% Completo	Más

En la parte inferior derecha hay un botón "Nuevo Proyecto".

■ Creación de características:

Al hacer click en alguno de los proyectos, es posible ingresar a la sección *Características* de este. En este caso, el usuario de prueba aun no cuenta con características:



Es posible agregar características a los proyectos, mediante el botón *Nueva Característica*, donde se solicitara al usuario ingresar los datos de la característica que se desea crear:

Crear video Creacion: 05/19/2018 04:25:11

Descripcion:

Razonamiento:

Informacion

Auspiciador:

Creado por:
 CISO
 a.com
 Email: isaoku_saope@hotmail.com

Requerimiento:

Prioridad:

Modulo:

Pre-Requisitos:

Requisito	Descripcion	Progreso	Pre-requisito
Característica	Características de prueba	0	<input type="checkbox"/>
Manual	Manual con flujo de uso	0	<input checked="" type="checkbox"/>

Sonyvegas x video x presentación x add a tag

Una vez creadas las características, es posible visualizarlas de la siguiente forma:

Características por Modulos Nueva Característica ?

HIGH

Crear video

URGENT

Manual

OPERATIONAL

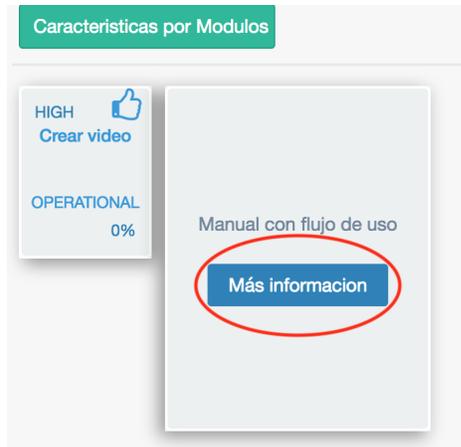
0%

LOOKANDFEEL

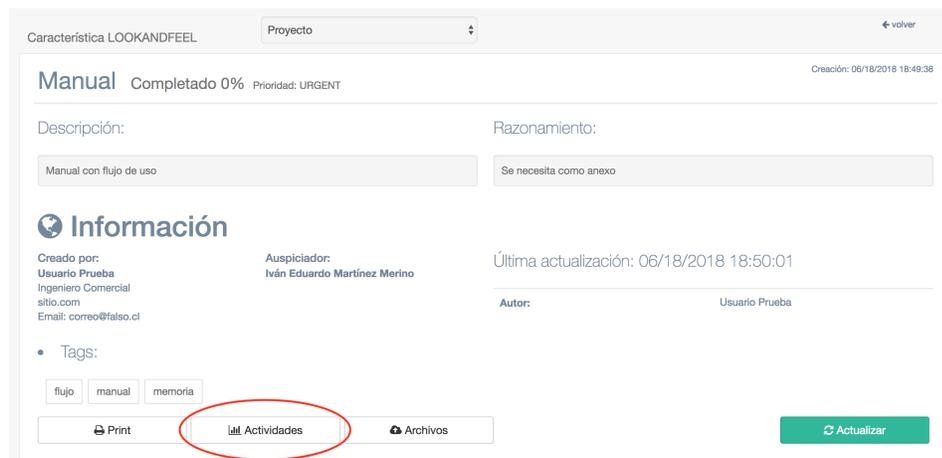
0%

■ Creación de actividades:

Para la creación de actividades, el usuario primero debe ingresar en características del proyecto, para posteriormente acceder al botón *Más información*, que como su nombre indica, otorga mayor información acerca de la característica:



Una vez realizado el ingreso a la característica, el usuario debe dirigirse al botón *Actividades*:



Una vez presionado el botón *Actividades*, el usuario tendrá acceso a las actividades de la característica que está revisando. En este caso, el usuario de prueba no cuenta con actividades asociadas, pero puede crear una presionando el botón *Crear Actividad*:



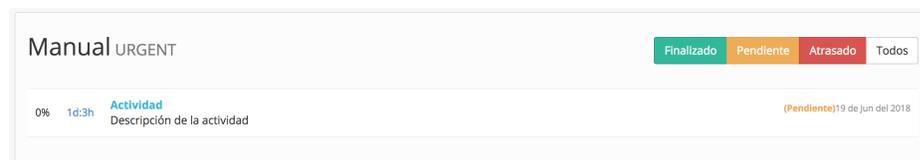
Una vez presionado el botón *Crear Actividad*, el usuario deberá ingresar la actividad que desee y seleccionar al o los usuarios que estarán encargados de llevarla a cabo:

 This screenshot shows a form titled 'Crea una actividad para Manual'. It contains the following fields:

- 'Actividad *': A text input field containing the word 'Actividad'.
- 'Descripción *': A text input field containing 'Descripción de la actividad'.
- 'Duración *': A date range selector showing '06/19/2018 - 06/21/2018'.
- 'Encargados de esta tarea *': A section with two user avatars: 'Usuario Prueba' (a black silhouette) and 'Iván Martínez' (a green smiley face). Below each avatar is a radio button. The 'Usuario Prueba' radio button is selected.

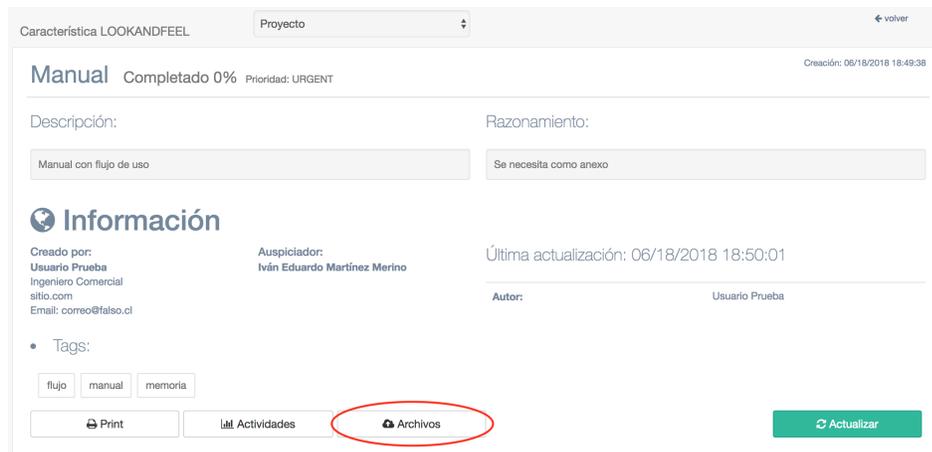
 At the bottom right of the form are two buttons: 'Cancelar' (blue) and 'Crear Actividad' (green).

Las actividades creadas, pueden ser visualizadas en la sección *Actividades* de las características:

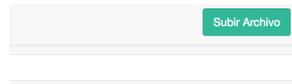


■ Agregar archivos a una característica:

Las características pueden contar con las actividades y archivos, estos últimos pueden ser ingresados presionando el botón *Archivos* dentro de la pantalla *Más información*:



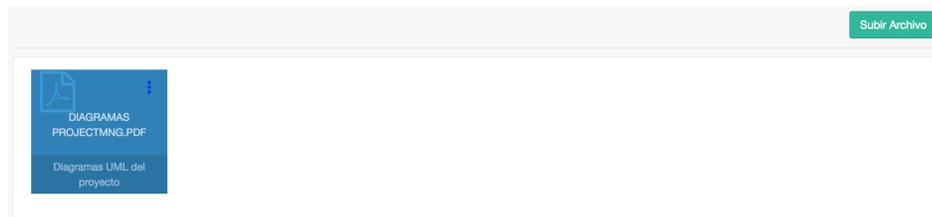
Para ingresar el archivo, sólo se necesita cargar el archivo desde el computador o dispositivo utilizado y presionar el botón *Subir Archivo*:



Una vez presionado el botón *Subir Archivo*, se desplegará un formulario, el cual debe ser completado y seleccionarse el archivo que se quiere cargar a la plataforma:

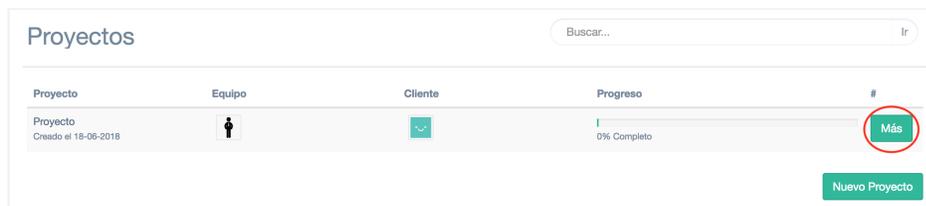


Los archivos cargados en el sistema, estarán disponibles para su visualización en la sección *Archivos*, en el detalle de las características:

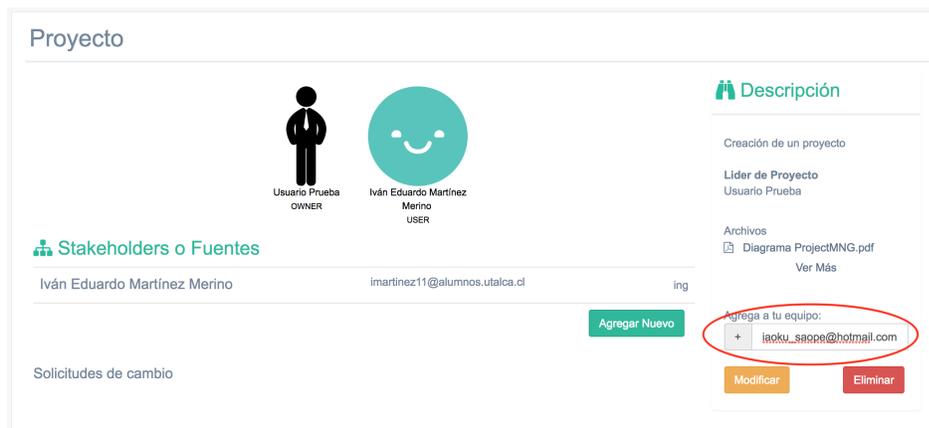


- **Incluir un miembro en el equipo:**

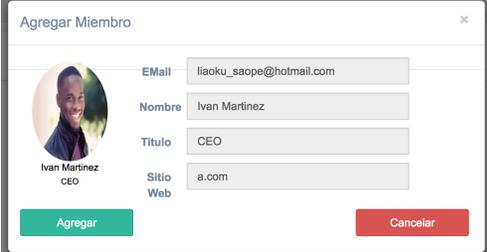
Para incluir a un nuevo miembro en un proyecto, se debe seleccionar el botón *Más* de alguno de los proyectos disponibles.



Para agregar a un nuevo miembro de equipo, este debe ser invitado mediante correo electrónico, utilizando un botón que el usuario encontrara dentro de la información del proyecto:



Al presionar el botón + aparecerá una ventana con la información relacionada al correo electrónico ingresado:

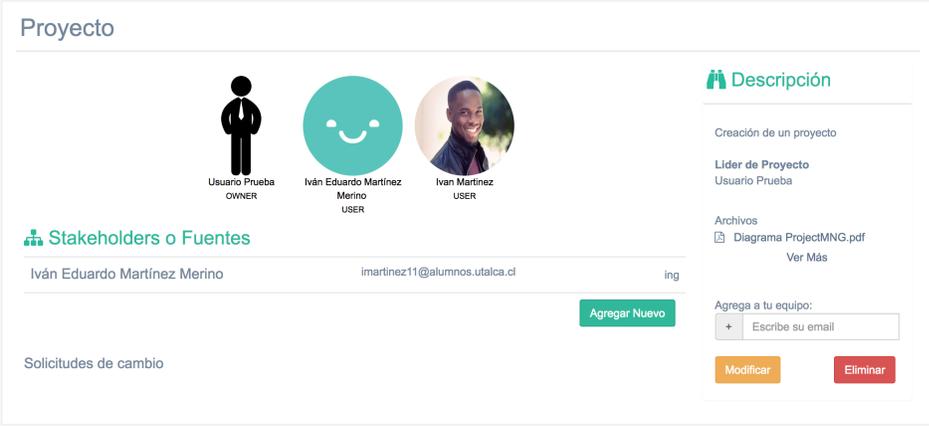


Formulario "Agregar Miembro" con los siguientes campos:

- EMail: liaoku_saope@hotmail.com
- Nombre: Ivan Martinez
- Título: CEO
- Sitio Web: a.com

Botones: Agregar (verde), Cancelar (rojo).

Una vez incluido el nuevo miembro del equipo, aparecerá en una miniatura con su foto:

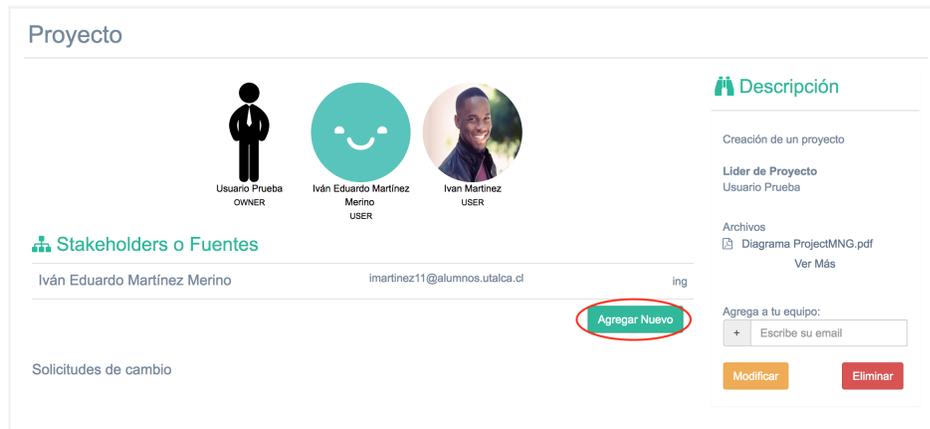


Vista de un proyecto con los siguientes elementos:

- Proyecto**
- Descripción**
 - Creación de un proyecto
 - Líder de Proyecto: Usuario Prueba
 - Archivos: Diagrama ProjectMNG.pdf (Ver Más)
 - Agrega a tu equipo: + Escribe su email
 - Botones: Modificar (naranja), Eliminar (rojo)
- Stakeholders o Fuentes**
 - Iván Eduardo Martínez Merino (imartinez11@alumnos.utalca.cl) ing
 - Botón: Agregar Nuevo (verde)
- Equipo**
 - Usuario Prueba (OWNER)
 - Iván Eduardo Martínez Merino (USER)
 - Ivan Martinez (USER)
- Solicitudes de cambio

- **Crear cuenta para stakeholder:**

Para la creación de la nueva cuenta de stakeholder, en el detalle del proyecto, presionamos donde se indica *Stakeholders o fuentes*, el botón *Agregar Nuevo*:



Al presionar *Agregar Nuevo*, se despliega una ventana donde el usuario debe ingresar los datos del stakeholder y presionar *Agregar*:

The dialog box is titled 'Agregar Nueva Fuente' and has a close button (X) in the top right corner. It contains three input fields: 'E-Mail' with the value 'stake@stake.com', 'Nombre' with the value 'Stakeholder', and 'Cargo' with the value 'ayudante'. At the bottom right, there are two buttons: 'Agregar' (green) and 'Cancelar' (red).

Luego de agregar a un stakeholder, es posible ver su usuario dentro de la pantalla *detalle del proyecto*:

The screenshot shows a 'Proyecto' detail page. At the top, there are four user icons: 'Usuario Prueba OWNER', 'Iván Eduardo Martínez Merino USER', 'Ivan Martínez USER', and 'Stakeholder USER'. Below this is a 'Stakeholders o Fuentes' section with a table:

Stakeholder	Usuario	Rol
Iván Eduardo Martínez Merino	imartinez11@alumnos.utalca.cl	ing
Stakeholder	stake@stake.com	ayudante

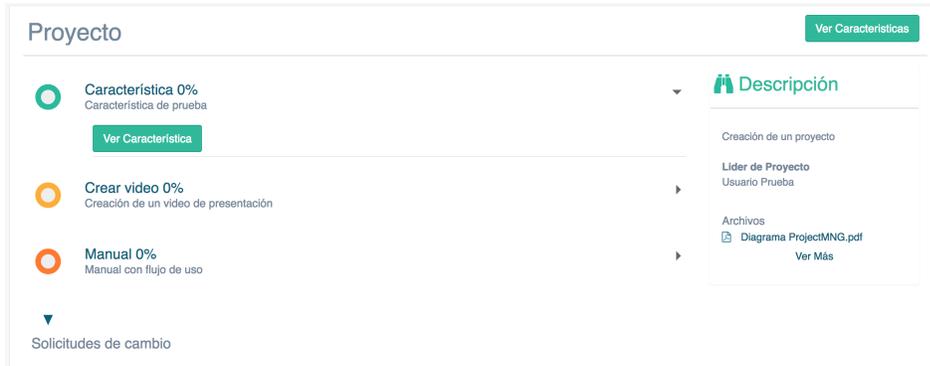
Below the table is a 'Solicitudes de cambio' section. To the right is a 'Descripción' section with a list of files: 'Functional-requirements.docx', 'UAT.pdf', 'Email-from-flatbal.min', 'Logo.png', 'Contract-10_12_2014.docx', and 'Contract-10_12_2014.docx'. There is also a section 'Agrega a tu equipo:' with a search box and 'Modificar' and 'Eliminar' buttons.

■ **Aceptar característica siendo stakeholder:**

Luego de iniciar sesión como stakeholder, podremos ver una lista de proyectos:

The screenshot shows a 'Proyectos' list page. At the top is a search bar with 'Buscar...' and 'Ir'. Below is a table with columns: 'Proyecto', 'Equipo', 'Cliente', and 'Progreso'. The first row shows a project created on 18-Jun-2018, with two team members, two clients, and 0% completion. A 'Nuevo Proyecto' button is at the bottom right.

Al presionar un proyecto, entraremos a su detalle y a una lista de características:



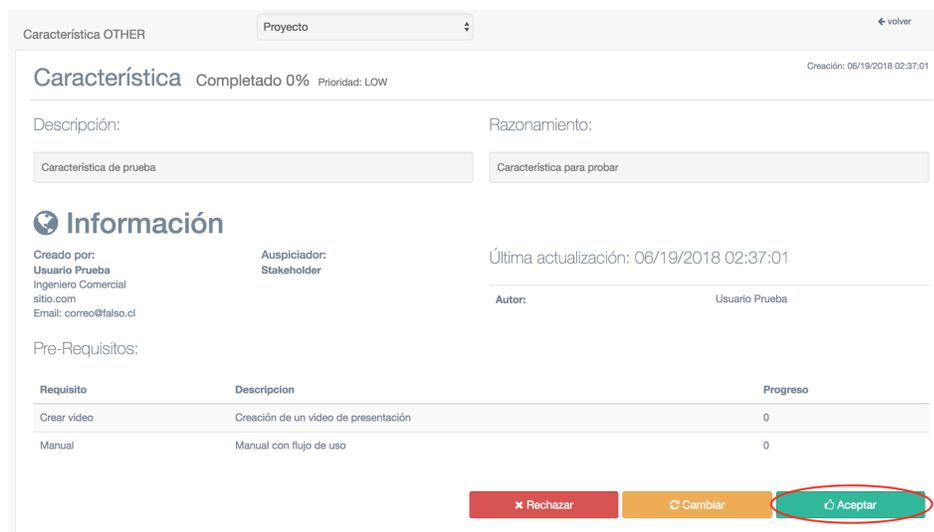
The screenshot shows a project detail page titled "Proyecto". At the top right, there is a green button labeled "Ver Características". Below the title, there is a list of characteristics:

- Característica 0%** (Característica de prueba) with a green circle icon and a "Ver Característica" button.
- Crear video 0%** (Creación de un video de presentación) with an orange circle icon.
- Manual 0%** (Manual con flujo de uso) with a red circle icon.

Below the list, there is a section for "Solicitudes de cambio". To the right, there is a "Descripción" panel with a green icon, containing the following text:

- Creación de un proyecto
- Líder de Proyecto: Usuario Prueba
- Archivos: Diagrama ProjectMNG.pdf
- Ver Más

Presione una característica, para acceder a su detalle y poder *Aceptar* su definición:



The screenshot shows a characteristic detail page titled "Característica OTHER". At the top, there is a dropdown menu set to "Proyecto" and a "volver" button. The characteristic is "Característica" with "Completado 0%" and "Prioridad: LOW". The creation date is "06/19/2018 02:37:01".

The page is divided into two columns:

- Descripción:** Característica de prueba
- Razonamiento:** Característica para probar

Información:

- Creado por: Usuario Prueba (Ingeniero Comercial, sitio.com, Email: correo@falso.cl)
- Auspiciador: Stakeholder
- Última actualización: 06/19/2018 02:37:01
- Autor: Usuario Prueba

Pre-Requisitos:

Requisito	Descripcion	Progreso
Crear video	Creación de un video de presentación	0
Manual	Manual con flujo de uso	0

At the bottom, there are three buttons: "Rechazar" (red), "Cambiar" (orange), and "Aceptar" (green, highlighted with a red circle).

E. Flujo Metodología Desarrollo

Fase Planning tiene el siguiente flujo de trabajo:

- . P1. Produce requirements statement.
- . A. Write System Metaphor.
- . B. Write User Stories
- . P2. FOR all new User Stories
- . A. Pick a User Story and break it into features.
- . END
- . P3. Create domain-driven design to reflect the domain model in your design.
- . P4. Group features into meaningful groups (FeatureSets).
- . P5. Write acceptance tests for FeatureSets
- . P5. Estimate FeatureSet size and required development time using ideal days.
- . P7. Sort FeatureSet according to priority to create FeatureSetPriorityList.
- . P8. Establish Iteration schedule.

En la fase de Development:

- . D1. FOR all FeatureSets in the FeatureSetPriorityList
- . A. Pick a FeatureSet to implement from top of the FeatureSetPriorityList.
- . B. IF (change introduced in this FeatureSet == YES)
- . 1. Update the FeatureSet.
- . 2. Re-sort the FeatureSetPriorityList.
- . C. ELSE
- . 1. Update design and create an iteration plan for the current FeatureSet.
- . 2. FOR all Features in the FeatureSet
- . a. Pick a feature from the FeatureSet and break it into tasks.
- . b. Sort tasks according to their priority to create a TaskPriority-

Fase Post Mortem:

- . M1. Complete acceptance testing for code in Production baseline.

Fase Exit:

- . X1. Tested program