**GDEVLINK**

ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE

VERSIÓN 1.0.16



JUAN PABLO MÉNDEZ PERDOMO

DANIEL HERNÁNDEZ GARCÍA

JUAN CARLOS SUÁREZ JAIMES

DIEGO ANDRÉS BURGOS MELO

GABRIEL GÓMEZ CORREDOR

SANTIAGO ANDRÉS CAROPRESE HIDALGO

16 DE OCTUBRE DE 2020

# Historial de cambios

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Versión | Fecha de cambio | Secciones modificadas | Descripción | Responsables |
| 1.0.0 | 06/10/2020 | 1 | Se creó el historial de cambios | Juan Pablo Méndez |
| 1.0.1 | 06/10/2020 | 7.4 | Características del usuario | Juan Pablo Méndez |
| 1.0.2 | 07/10/2020 | 7.5 | Se creó la tabla de requisitos no funcionales | Gabriel Gómez Corredor |
| 1.0.3 | 08/10/2020 | 7.6 | Se listaron las suposiciones y dependencias | Gabriel Gómez Corredor |
| 1.0.4 | 09/10/2020 | 8.3 | Atributos del sistema de software | Juan Pablo Méndez |
| 1.0.5 | 09/10/2020 | 8.2 | Se definió requisitos de desempeño | Diego Burgos |
| 1.0.6 | 09/10/2020 | 9.1 | Planeación | Santiago Caroprese Hidalgo |
| 1.0.7 | 10/10/2020 | 7.4 | Se complementaron las características del usuario | Juan Carlos Suárez |
| 1.0.8 | 11/10/2020 | 9.2 | Se definieron las actividades para el levantamiento de requisitos. | Santiago Caroprese Hidalgo |
| 1.0.9 | 14/10/2020 | 10 | Proceso de verificación y validación. | Juan Pablo Méndez. |
| 1.0.10 | 14/10/2020 | 9.3 | Se hicieron las especificaciones de los requisitos. | Diego Burgos, Gabriel Gómez, Juan Pablo Méndez. |
| 1.0.11 | 14/10/2020 | 8.1 | Características del producto se software. | Juan Pablo Méndez. |
| 1.0.12 | 15/10/2020 | 8.4 | Se definieron requisitos de la base de datos | Diego Andrés Burgos Melo. |
| 1.0.13 | 15/10/2020 | 7.2 | Perspectiva del producto | Daniel Hernández y Juan Carlos Suárez |
| 1.0.14 | 16/10/2020 | 6 | Introducción | Juan Carlos Suárez |
| 1.0.15 | 16/10/2020 | 2 | Resumen | Juan Carlos Suárez |
| 1.0.16 | 16/10/2020 | 7.5 | Se actualizó la tabla de requisitos no funcionales | Gabriel Gómez Corredor |

# Resumen

Este documento describe el proceso de especificación y análisis de requisitos llevado a cabo para el desarrollo de la aplicación web GDevLink, la cual se construye en un ambiente académico, con el objetivo de poner en práctica los fundamentos de la ingeniería de software. El proceso descrito en este documento incluye una descripción global del proyecto, la especificación de los requisitos funcionales y no funcionales, y una descripción del proceso de ingeniería de requisitos. La especificación de estos componentes tiene como objetivo definir formalmente las características que deberá cumplir el producto final. La audiencia esperada de este documento son aquellas personas interesadas en el proceso de especificación de requisitos de un proyecto de software en un ambiente académico.

# Tabla de contenidos

[1 Historial de cambios 2](#_Toc53773668)

[2 Resumen 3](#_Toc53773669)

[3 Tabla de contenidos 3](#_Toc53773670)

[4 Lista de figuras 4](#_Toc53773671)

[5 Lista de tablas 5](#_Toc53773672)

[6 Introducción 5](#_Toc53773673)

[7 Descripción global 6](#_Toc53773674)

[7.1 Modelo de dominio 6](#_Toc53773675)

[7.2 Perspectiva de producto 6](#_Toc53773676)

[7.2.1 Interfaces con el sistema 7](#_Toc53773677)

[7.2.2 Restricciones de memoria 8](#_Toc53773678)

[7.2.3 Operaciones 9](#_Toc53773679)

[7.3 Funciones del producto 9](#_Toc53773680)

[7.4 Características del usuario 9](#_Toc53773681)

[7.5 Requisitos No Funcionales 12](#_Toc53773682)

[7.6 Suposiciones y dependencias 15](#_Toc53773683)

[8 Requisitos específicos 15](#_Toc53773684)

[8.1 Características del producto software 15](#_Toc53773685)

[8.2 Requisitos de desempeño 16](#_Toc53773686)

[8.3 Atributos del sistema software 17](#_Toc53773687)

[8.4 Requisitos de la base de datos 19](#_Toc53773688)

[9 Proceso de ingeniería de requisitos 20](#_Toc53773689)

[9.1 Planeación 20](#_Toc53773690)

[9.2 Levantamiento 21](#_Toc53773691)

[9.3 Especificación 22](#_Toc53773692)

[10 Proceso de verificación y validación 26](#_Toc53773693)

[11 Anexos 29](#_Toc53773694)

[12 Referencias 30](#_Toc53773695)

# Lista de figuras

[Ilustración 1. Diagrama indexacion de dato 20](#_Toc53779764)

[Ilustración 2. Proceso de verificación y validación login 27](#_Toc53779765)

[Ilustración 3. Proceso de verificación y validación Creación Proyecto 27](#_Toc53779766)

[Ilustración 4. Proceso de verificación y validación de errores 28](#_Toc53779767)

[Ilustración 5. Proceso de verificación y validación sistema operativo 28](#_Toc53779768)

[Ilustración 6. Proceso de verificación y validación repositorio 29](#_Toc53779769)

# Lista de tablas

[Tabla 1. Interfaces con el sistema. 7](#_Toc53779750)

[Tabla 2. Requisitos de cómputo por sistema operativo 8](#_Toc53779751)

[Tabla 3. Requisitos de cómputo para el servidor Web 8](#_Toc53779752)

[Tabla 4. Características del usuario no registrado 10](#_Toc53779753)

[Tabla 5. Características del usuario registrado 10](#_Toc53779754)

[Tabla 6. Características del administrador 11](#_Toc53779755)

[Tabla 7. Características del administrador maestro 12](#_Toc53779756)

[Tabla 8. Clasificación de requisitos no funcionales según modelo FURPS+. 13](#_Toc53779757)

[Tabla 9. Atributos del Sistema Disponibilidad 17](#_Toc53779758)

[Tabla 10. Atributos del Sistema Confiabilidad 17](#_Toc53779759)

[Tabla 11. Atributos del Sistema Mantenibilidad 18](#_Toc53779760)

[Tabla 12. Atributos del Sistema Portabilidad 18](#_Toc53779761)

[Tabla 13. Atributos del Sistema Seguridad 18](#_Toc53779762)

[Tabla 14 Encargados procesos de requisitos 21](#_Toc53779763)

# Introducción

El propósito de este documento es identificar y analizar los requisitos del producto de software a ser desarrollado. Esto implica definir las condiciones que debe cumplir el sistema a partir de las necesidades del cliente. Este documento resulta importante para el desarrollo de un producto de software, dado que ayuda al cliente a describir claramente lo que desea obtener y ayuda a los desarrolladores a entender lo que quiere el cliente [1]. En el caso de particular de este proyecto, el equipo de desarrollo cumple en parte el rol de cliente, por lo que elaborar este documento le permite al equipo definir formalmente todas las características y condiciones con las que el producto final deberá cumplir.

El producto cuyo desarrollo motiva la elaboración del presente documento se llama GDevLink, una aplicación web que busca facilitar el contacto entre personas interesadas en el desarrollo de videojuegos como hobby. Las principales funciones de la aplicación incluyen la creación y gestión de perfiles de usuarios, la creación y gestión de proyectos, la creación y gestión de posiciones vacantes por parte de los proyectos, a las cuales pueden aplicar los usuarios, la comunicación entre usuarios y una sección de preguntas en la cual los usuarios pueden crear preguntas y responder a las inquietudes de otros usuarios.

El desarrollo de esta aplicación es motivado por la falta de una plataforma común para la comunicación entre los miembros de la comunidad de desarrollo de videojuegos. Actualmente existe en internet una gran cantidad de personas que en su tiempo libre colaboran en el desarrollo de proyectos ambiciosos de desarrollo de videojuegos, pero la falta de un medio de comunicación especializado provoca que entrar en contacto con personas con intereses similares sea difícil. GDevLink tiene como objetivo solucionar esta problemática, facilitando el contacto entre los miembros de esta comunidad e incentivando la ayuda mutua entre ellos. Se considera que una aplicación de este tipo podría tener un impacto positivo sobre esta comunidad y reducir la barrera de entrada a ella, impulsando el poder creativo de aquellas personas apasionadas por el desarrollo de videojuegos.

# Descripción global

## Modelo de dominio

En esta sección presentamos el modelo de dominio. Este representa una visualización y descripción de conceptos abstractos del mundo real, que están en el sistema [2]. Su representación visual, en donde se muestra cada elemento se muestra en el anexo 1, y la descripción e información de cada elemento, se detalla en el anexo 2.

## Perspectiva de producto

GDevLink es un producto totalmente nuevo que es desarrollado para crear un canal de comunicación dedicado para los desarrolladores de videojuegos. Esta aplicación busca fomentar la interacción y la comunicación entre los miembros de la comunidad de desarrollo de videojuegos que actualmente existe en internet. Se evidencia que actualmente se utilizan diversos medios para la comunicación entre los miembros de esta comunidad, lo que puede dificultar entrar en contacto con personas con intereses similares. Por este motivo, se considera que el establecer un medio de comunicación estándar y especializado para los miembros de esta comunidad puede beneficiar en gran medida a los usuarios finales, facilitando en gran medida la comunicación y el contacto con otras personas interesadas en el desarrollo de videojuegos.

Teniendo en cuenta la oportunidad que representa GDevLink para mejorar la comunicación entre los miembros de la comunidad de desarrollo de videojuegos, se identificaron dos aspectos fundamentales de esta interacción que se podrían ver especialmente beneficiados por una plataforma como la planteada.

En primer lugar, GDevLink permitiría facilitar en gran medida el proceso de búsqueda de proyectos en los cuales trabajar, por parte de los desarrolladores de videojuegos, y el proceso de búsqueda de desarrolladores, por parte de los proyectos de desarrollo, beneficiando a ambas partes involucradas. Se identificó que los medios más utilizados actualmente carecen de una manera fácil de buscar proyectos de desarrollo, por lo que una plataforma que resuelva este problema de manera integral podría ser favorable para la comunidad. Esto podría resultar especialmente beneficioso para aquellas personas con poca experiencia que buscan empezar a trabajar en proyectos para armar un portafolio, dado que una plataforma como esta podría permitirles encontrar con mayor facilidad personas con los mismos intereses con las cuales puedan colaborar.

En segundo lugar, se identificó que GDevLink también resultaría especialmente útil para el intercambio de conocimiento entre desarrolladores de videojuegos, dado que, al contar con usuarios con diversos niveles de experiencia, se presenta una oportunidad valiosa de ayuda mutua entre los miembros de la comunidad, que permitiría fortalecer la misma y contribuir a la formación de sus miembros.

### Interfaces con el sistema

En la siguiente tabla se describen las interfaces que utilizará el sistema para comunicarse con el usuario y otros sistemas.

Tabla 1. Interfaces con el sistema.

|  |  |
| --- | --- |
| Interfaz | Descripción |
| Interfaz de usuario | Conjunto de archivos HTML, CSS y JavaScript que son utilizados por el navegador para desplegar páginas del sitio web dentro de una ventana. El código HTML define el contenido de la página, el código CSS define su estilo y el código JavaScript define su comportamiento. Esta interfaz de usuario registra la entrada del usuario a través del navegador, permitiendo obtener información acerca eventos relacionados con las acciones realizadas a través del mouse, en relación con la ventana, incluyendo los clicks del usuario sobre la ventana y el movimiento del mouse sobre la misma. En los dispositivos móviles, el mouse es reemplazado por una pantalla táctil, la cual cumple las mismas funciones. Asimismo, el navegador web permite obtener la entrada del teclado, haciendo posible registrar automáticamente la entrada del usuario al llenar los campos de un formulario. Todas las interacciones con el usuario se llevarán a cabo a través del navegador, por lo que este resulta un componente fundamental para la interfaz con el usuario. La interfaz utilizada para esta interacción está definida por los estándares del W3C, los cuales buscan brindar una experiencia interactiva y consistente a través de los diversos navegadores y dispositivos [3]. Por lo tanto, para este componente, se asume que el navegador web utilizado por el usuario cumple con estos estándares. |
| Autenticación con credenciales de Google | Al permitir autenticación a través de Google, la aplicación deberá utilizar la API de Google. Esta API utiliza el protocolo OAuth 2.0 para la autenticación y la autorización de usuarios. Para poder utilizar este servicio, es necesario que la aplicación solicite un token de acceso al servidor de autorización de Google, el cual se le enviará a la API [4]. A través de la API es posible obtener información del perfil de Google del usuario, lo que permite registrarlo y autenticarlo. A través de la API también es posible cerrar la sesión del usuario en el sitio web sin necesidad de cerrar su sesión en Google [5]. |

### Restricciones de memoria

Ya que el producto que se está realizando es un servicio que se presenta como página web, se vuelve evidente la necesidad de un navegador web. En el mercado existen diversas opciones, y cada una utiliza los recursos del computador en distintas medidas, así que las necesidades de la capacidad de cómputo están dadas por el tipo de navegador. En este sentido, se puede decir que cualquier dispositivo capaz de utilizar un navegador de internet es capaz de utilizar el producto. Con base en la información encontrada, el navegador que más recursos utiliza es Google Chrome [6]. De ese modo, para especificar los requisitos mínimos se tendrá en cuenta los requisitos de Google Chrome para diferentes dispositivos y sistemas operativos [7, 8]. Los campos no especificados en la tabla se deben ya sea a que no tienen un requisito mínimo o que no se especifica por los creadores del sistema operativo.

Tabla 2. Requisitos de cómputo por sistema operativo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sistema Operativo | Procesador | RAM | Almacenamiento | Tarjeta Grafica |
| Windows 7, Windows 8, Windows 8.1, Windows 10 [9, 10] | Procesador Intel Pentium 4 o posterior compatible con SSE2 | 1 GB de RAM (32 bits) o 2 GB de RAM (64 bits) | 16 GB (32 bits) o 20 GB (64 bits) | DirectX 9 con controlador WDDM 1.0 o superior |
| OS X Yosemite 10.10 [11] | - | 2 GB | 8 GB | - |
| 64 bits de Ubuntu 14.04, Debian 8, openSUSE 13.3, Fedora Linux 24 [12] | Procesador Intel Pentium 4 o posterior compatible con SSE2 | 512 Mb | 5 GB | Tarjeta gráfica y monitor capaz de soportar una resolución de 1024×768 |
| iOS 12 o posterior | - | - | - | - |
| Android 5.0 (Lollipop) [13] | Quad Core | 1Gb | 8Gb | - |

Para especificar las necesidades que requiere para mantener el funcionamiento del servidor web, se buscaron diferentes servicios de web hosting y se llegó a los requisitos de la tabla 3 [14, 15]:

Tabla 3. Requisitos de cómputo para el servidor Web

|  |  |  |  |
| --- | --- | --- | --- |
| Espacio en Disco | Base de datos | Núcleos CPU | RAM |
| 10 GB | 3GB | 1 | 256 MB |

### Operaciones

El producto posee dos modos de uso: el modo administrador y el modo usuario. El modo administrador este reservado para aquellas personas encargadas de administrar la información almacenada en la base de datos. Este modo permite agregar y eliminar datos y está pensado principalmente para realizar resoluciones de conflictos que se puedan presentar. El modo usuario corresponde al uso normal del servicio, especificados en la sección 7.4.

Por sus requisitos (Sección 7.5) el sistema debe permanecer activo el mayor tiempo posible, pero es necesario realizar actividades administrativas o de mantenimiento, por ello se establece que el último día de cada mes, entre las 3 a.m. y las 5 a.m. de acuerdo con UTC -5, se presentara un periodo de inactividad para realizar dichas actividades. Además, cada cuatro meses este periodo de inactividad se extenderá desde la 1 a.m. hasta las 5 a.m. para realizar operaciones de respaldo y recuperación de información.

## Funciones del producto

Se anexa el diagrama de casos de uso, en el cual se incluyen los casos de uso del proyecto, relacionados a los actores que los pueden iniciar. En este diagrama se puede observar la herencia de los tipos de usuario, descrita en la sección 7.4. También se anexa un documento con la especificación de los casos y los mockups de las páginas de cada uno.

## Características del usuario

Nuestra aplicación web está dirigida a todas las personas que estén interesadas en el desarrollo de videojuegos y que quieran entrar en contacto con otras personas con intereses similares para colaborar en proyectos de desarrollo de videojuegos o que quieran participar en un intercambio de conocimiento con los demás miembros de la comunidad de desarrollo de videojuegos. La página web será diseñada con una interfaz gráfica intuitiva y amigable para el usuario, con el objetivo de que sea fácil de utilizar, incluso para personas sin demasiada experiencia en la interacción con aplicaciones web. Se pueden identificar 4 tipos de usuarios dentro de la aplicación: usuarios no registrado, usuario registrado, administrador y administrador maestro.

Cabe mencionar que se puede establecer una jerarquía de herencia a partir de estos tipos de usuario, donde un tipo de usuario determinado hereda todos los privilegios del tipo usuario del que hereda, incluyendo además privilegios adicionales. Como se puede observar en el diagrama de casos de uso, el administrador maestro hereda del administrador, el administrador hereda del usuario registrado y el usuario registrado hereda del usuario no registrado. También resulta importante aclarar que los roles de administrador y de administrador maestro son relativos a los proyectos, por lo que el hecho de que un usuario sea administrador o administrador maestro de un determinado proyecto no implica que también lo sea para los demás proyectos de la aplicación. A continuación, se especifican las características de cada uno de los tipos de usuario mencionados.

Tabla 4. Características del usuario no registrado

|  |  |
| --- | --- |
| Usuario no registrado | Descripción |
| Nivel de seguridad o de privilegios | Estos usuarios no son autenticados. Pueden buscar y visualizar páginas de proyectos, perfiles de usuarios registrados, posiciones vacantes disponibles y también preguntas, pero no pueden crear ni seguir proyectos, realizar ni responder preguntas, enviarles mensajes a otros usuarios, ni postularse a una posición vacante. En la página principal, estos usuarios pueden visualizar la información principal de los proyectos con mayor número de seguidores |
| Roles | Visitante: Puede visualizar parte de la información de la aplicación, pero no puede aplicar ningún cambio sobre esta. |
| Nivel de estudios o experiencia técnica | Se espera que aproximadamente el 50% de los usuarios no registrados tenga algo de experiencia en el desarrollo de videojuegos y que el 95% tenga por lo menos algo de experiencia en uso de aplicaciones web con una navegación y características similares. |
| Frecuencia de uso | Se espera que para los usuarios no registrados la frecuencia de uso sea como máximo 2 veces por semana, dado que se espera que un usuario que le dé un uso mayor consideraría en registrarse para acceder a todas sus funcionalidades. |

Tabla 5. Características del usuario registrado

|  |  |
| --- | --- |
| Usuario registrado | Descripción |
| Nivel de seguridad o de privilegios | Estos usuarios deben ser autenticados. Pueden visualizar la misma información disponible para los usuarios no registrados, pero además pueden crear y seguir proyectos, realizar y responder preguntas, enviarles mensajes a otros usuarios y postularse a posiciones vacantes. Este usuario tiene también acceso a su perfil, el cual puede editar. Estos usuarios pueden visualizar en la página principal las actualizaciones de progreso de sus proyectos seguidos y las posiciones vacantes que podrían interesarles. |
| Roles | (Hereda roles de usuario no registrado)  Dueño de cuenta: Al usuario le pertenece una cuenta, cuya información puede gestionar  Contribuidor: El usuario puede crear nuevos objetos dentro de la base de datos la aplicación, los cuales se encuentran asociados a su cuenta.  Suscriptor: El usuario puede seguir proyectos para visualizar en la página principal las publicaciones de este. |
| Nivel de estudios o experiencia técnica | Se espera que aproximadamente el 80% de los usuarios registrados tengan alguna experiencia en el desarrollo de videojuegos, entre los cuales se encontrarán desarrolladores experimentados y novatos que desean aprender más. |
| Frecuencia de uso | Se espera que el 60% de los usuarios utilice la aplicación más de 2 veces por semana. |

Tabla 6. Características del administrador

|  |  |
| --- | --- |
| Administrador | Descripción |
| Nivel de seguridad o de privilegios | Debe ser autenticado. Tiene todos los privilegios de un usuario registrado, pero adicionalmente puede realizar las siguientes acciones en aquellos proyectos en los en los que tiene este rol: editar la página del proyecto, gestionar sus miembros, agregar nuevas actualizaciones de progreso y crear y gestionar posiciones vacantes. |
| Roles | (Hereda roles de usuario registrado)  Administrador: El usuario puede gestionar la información de los proyectos en los que cumple este rol. También puede crear objetos dentro de la base de datos de la aplicación, los cuales se encuentran asociados al proyecto. |
| Nivel de estudios o experiencia técnica | Se espera que los administradores ya tengan experiencia en el desarrollo de videojuegos. Sin embargo, es probable que todavía no hayan administrado un proyecto de este tipo. |
| Frecuencia de uso | Se espera que estos usuarios utilicen la aplicación por lo menos 4 veces por semana. |

Tabla 7. Características del administrador maestro

|  |  |
| --- | --- |
| Administrador maestro | Descripción |
| Nivel de seguridad o de privilegios | Debe ser autenticado. Tiene todos los privilegios del administrador, pero al ser el creador del proyecto, puede otorgar y revocar permisos de administrador a los miembros del proyecto creado. |
| Roles | (Hereda roles de administrador)  Súper administrador: Puede otorgar y revocar permisos de administrador dentro del proyecto en el que cumple este rol. |
| Nivel de estudios o experiencia técnica | Se espera que aproximadamente el 60% de los usuarios con este rol tenga experiencia administrando procesos de desarrollo de videojuegos. |
| Frecuencia de uso | Se espera que estos usuarios utilicen la aplicación por lo menos 4 veces por semana. |

## Requisitos No Funcionales

En cuanto a las características generales de la aplicación se tuvo en cuenta toda la lógica de negocio que va a establecer el software. Para ello, se analizaron adecuadamente todas las características que están presentes en las funciones que están al servicio del usuario.

Antes de presentar y clasificar los requisitos no funcionales según el modelo FURPS+, resulta importante resaltar cómo se realizó el análisis correspondiente a las restricciones del sistema.

* **Características y restricciones generales:** Las características y restricciones generales de la aplicación van a regir entorno de la lógica de negocio del software [16]. Se plantearon primero todas las funcionalidades que el software debe a brindarle al usuario y basándose en ellas, se establecieron los posibles escenarios no deseables que se pueden presentar al momento de hacer uso de cada una de estas funcionalidades. Estos posibles escenarios pueden ser dados por errores del usuario, falta de comprensión del sistema y sus características, etc. De acuerdo con los posibles escenarios identificados, el software establecerá una serie de restricciones que ayudarán a reducir la probabilidad de que se presenten. También se incluirá la tolerancia a fallos, ya sean clasificados como fallos transitorios, permanentes o intermitentes, esto permitirá que el sistema sea confiable. Por último, en esta categoría también se incluyen los idiomas soportados por el software.
* **Restricciones de Software:** Las restricciones del software van a estar orientadas hacia las licencias adquiridas para el desarrollo adecuado del proyecto. Dentro de estas restricciones también se encontrarán las API’s con las cuales el proyecto interactuará y los lenguajes de programación usados para la implementación. Con respecto a las restricciones lenguajes de programación, estas estarán enfocadas la compatibilidad que tienen estos tanto como con el framework a utilizar (Django) y la base de datos (Postgres). Esta compatibilidad se basa en la medida en que las funciones que brindan las librerías de estos lenguajes tengan afinidad con el framework y la base de datos. También se analizó que estos lenguajes de programación brinden funcionalidades de autenticación y seguridad de datos.
* **Restricciones de Hardware:** Las restricciones de hardware van a estar enfocadas en las características de hardware que debe tener la máquina que ejecutará el software implementado. Estas restricciones son de vital importancia ya que es importante establecer unos requisitos mínimos para el buen desempeño de la aplicación, la velocidad de respuesta, la resolución, la tarjeta de red, etc.

**NOTA:** En la sección de usabilidad, los requisitos no funcionales hacen referencia a las notas de ayuda que se le brindará al usuario para que su navegación por la aplicación sea más cómoda y fácil de usar, y así evitar errores que el usuario pueda cometer [17].

Tabla 8. Clasificación de requisitos no funcionales según modelo FURPS+.

|  |  |
| --- | --- |
| **Usabilidad** | **Desempeño (Performance)** |
| * U-1: Se mostrarán mensajes que les indiquen a los usuarios las acciones que pueden realizar en cada una de las páginas y como acceder a ellas. * U-2: En las páginas de registro, creación de proyecto, creación de posición vacante y creación de actualización de progreso se le indicarán al usuario cuáles son los campos obligatorios. * U-3: En caso de que ocurra un problema en el registro o en la creación de un proyecto, como resultado de las reglas de negocio establecidas, se le indicará al usuario la razón del problema (por ejemplo, indicar que el usuario ingresado ya existe). * U-4: En los dispositivos móviles, los elementos de la página deben ajustarse para ser visibles de manera clara y organizada. | * P-1: El servidor estará disponible al menos 24/6, el usuario podrá acceder a la aplicación en cualquier momento del día [18]. * P-2: El sistema debe soportar hasta 10 millones de Hits (imágenes, texto, archivos, etc.) debido al buen desempeño del servidor [18] [19]. * P-3: El sistema debe proveer funciones request/response que establezcan una conexión de respuesta de 3 segundos promedio con el servidor [32]. |
| **Confiabilidad (Reliability)** | **Soporte** |
| * R-1: El sistema deberá ofrecer un tiempo de holgura de 10 segundos aproximadamente en el peor caso de peticiones de request/response donde estas tengan un tiempo de respuesta demorado para la renderización de páginas de la aplicación, ya que Django brinda esta tolerancia y lo hace por defecto. * R-2: El sistema deberá bloquear las acciones del usuario si este está saturando las solicitudes al servidor. | * S-1: Los archivos del proyecto estarán en un repositorio compartido con todos los miembros de desarrollo para el fácil mantenimiento colectivo de la aplicación. * S-2: Se dividirá la aplicación principal en aplicaciones más pequeñas (main, proyectos, usuarios), estos componentes permitirán al proyecto adaptarse al cambio fácilmente debido a su modulación. * S-3: Se documentará el proyecto adecuadamente de una forma clara y precisa para que el momento de internacionalizar la aplicación facilite el acoplamiento de entidades externas al proyecto. * S-4: La arquitectura del proyecto debe ser capaz de soportar cambios a largo plazo que impliquen la extensión del software debido a la demanda, eso hace referencia a la extensibilidad. |
| **+** | |
| * O-1: Todos los archivos que componen el software se empaquetarán en un contenedor Docker para la adecuada distribución del software [20]. * O-2: El sistema deberá utilizar un orquestador de contenedores llamado Kubernetes para el uso correcto del contenedor Docker [21]. * O-3: El sistema implementará cross site scripting (XSS) protection [22]. * O-4: El sistema implementará cross site request forgery (CSRF) protection [22]. * O-5: El sistema implementará SQL injection protection [22]. * O-6: El sistema implementará clickjacking protection [22]. * O-7: El sistema implementará Host Header Validation [22]. * O-8: La máquina que ejecutará la aplicación deberá tener una tarjeta de red de buen desempeño para la rápida respuesta que ofrece el servidor. * O-9: La máquina deberá tener mínimo un procesador Intel Core I3. * O-10: Se podrá acceder a la aplicación en dispositivos móviles y de escritorio. * O-11 Se podrá acceder a la aplicación desde cualquier sistema operativo. * O-12: Se podrá acceder a la aplicación desde cualquier navegador W3C-compliant. | |

## Suposiciones y dependencias

En la siguiente sección, se listarán todos los factores que pueden llegar a afectar tanto como los requisitos funcionales como los no funcionales.

**Suposiciones**:

* La máquina del usuario que ejecuta la aplicación deberá tener una conexión de red estable junto con la conexión al servidor.
* El cliente accederá a la aplicación desde un navegador web “W3C-compliant”.
* La topología de la red del usuario que hace uso de la aplicación deberá ser estable y conectada a internet
* Las librerías de los lenguajes de programación utilizados deberán seguir vigentes a lo largo del desarrollo del software.
* Los requisitos del sistema no cambiarán durante el desarrollo del proyecto.
* El usuario que hará uso de la aplicación deberá seguir las restricciones que rige el software al momento de crear un perfil
* El sistema de control de versiones (Github), se mantendrá en funcionamiento durante el desarrollo del proyecto
* La conectividad del usuario que está haciendo uso de la aplicación, deberá tener una conexión a internet estable durante el uso de la página web.
* El usuario deberá leer los mensajes de campos obligatorios que la aplicación le brinda para evitar cometer errores al momento de crear proyecto o crear cuenta.

**Dependencias**:

* La velocidad de red que tenga el usuario que hace uso de la aplicación
* El buen desempeño de la máquina del usuario que hace uso de la aplicación
* La disposición y el aporte de todos los miembros del equipo de desarrollo
* La conexión a internet brindada por el proveedor de comunicaciones
* El correcto funcionamiento del servidor del framework (Django)
* El correcto funcionamiento del sistema de control de versiones (Github)

# Requisitos específicos

## Características del producto software

Un requisito funcional es lo que define que hace el sistema, en pocas palabras las funciones del sistema. En nuestro caso identificamos 26 requisitos funcionales con los cuales nuestro software será exitoso y viable.

En esta sección hablaremos sobre todos los requisitos funcionales de nuestra página web, para esto describiremos todos los requisitos en tablas las cuales contienen como ítems descripción, razón, autor, criterio de medición, prioridad, versión, modulo asociado, fecha, numero del requisito, tipo de requisito y caso de uso asociado. Para esto tuvimos que identificar todos los requisitos funcionales de nuestro sistema el cual recibe entradas y produce unas salidas. Para ver esto con más detalle mirar el anexo incluido en esta parte llamado “8.1 Anexo Características del producto de software”.

## Requisitos de desempeño

En cuanto a los requisitos se tendrán en cuenta los requerimientos que el dueño del problema sugirió al scrum máster junto con algunas pautas que la aplicación web debería tener y soportar en todo momento, en cuanto al desempeño de esta a partir del despliegue. Todo esto con el fin de que el equipo scrum trabaje de manera conjunta con el líder de arquitectura de software, para tomar decisiones que se vean reflejadas en los atributos de calidad.

* **Latencia**: En cuanto al tiempo que va a tomar el evento ocasionado por un usuario desde el momento de su llegada hasta el momento de la misma resolución de los eventos individuales se tendrá encuentra menos de 3 segundos y para los eventos más complejos como la renderización de una página entera un máximo de 10 segundos. para ello se tendrán en cuenta decisión arquitecturales como la limitación del tiempo en los procesos sin importar si esta petición resulto exitosa o no, priorización de eventos, limitación de la cantidad de estilos mostrados por parte del css, compresión y uso correcto de formatos en las imágenes usadas.
* **Tasa de procesamiento:** En cuanto a la cantidad respuesta a las n peticiones creadas por los n usuarios en un determinado intervalo de tiempo, se tendrá en cuenta un factor muy importante el cual es buffering de cada navegador utilizado por el usuario, ya que estos atributos son propios de cada navegador nos puede ayudar a alivianar procesamiento reutilizando archivos, ya sean imágenes elementos css que puedan ayudar a que las peticiones sean más rápidas. Por otra parte, la cantidad de peticiones y como estas afectan a nuestra tasa de procesamiento serán monitoreadas por la herramienta de desarrolladores de Google en el apartado Network, la cual nos ayudara a medir la cantidad, que tipo de petición fue, su peso y su tiempo de respuesta para cada uno. Con lo cual podremos tener en cuantas vistas que estén cargando muchos elementos de manera errónea y reduzcan la tasa de procesamiento, así tomando correctivos arquitecturales como la reducción de elementos según página renderizada.

\*Nota: las pruebas de rendimiento serán evaluadas y corregidas con la medición que nos va a proporcionar la herramienta proporcionada por Google llamada “Google PageSpeed Insights”, mostrando varias pruebas como velocidad y rendimiento entre otras. Estas pruebas serán apoyo a los criterios singulares especificados en otras secciones [23].

* **Capacidad:** para mantener una buena capacidad de respuesta antes que los eventos sobrepasen la latencia, debemos asegurar un flujo de solicitudes desde el navegador hasta el servidor de la página en las otras aplicaciones como PostgreSQL Django para que allá se procesen y sean devueltos los resultados, mostrándolos en el navegador. Para esto debemos tomas decisiones y asegurar la optimización en el código, limitar errores y aumentar el cache del servidor. Así mismo que los servidores tengan ubicación cercana ganando una latencia baja y por último el buen tamaño de datos pesados como imágenes y videos.
* **Disponibilidad:** en cuanto a la cantidad de tiempo que nuestra aplicación web va a estar disponible para la entrada y la realización de peticiones por parte de los usuarios se intentará mantener en el 98% de disponibilidad semestral (teniendo en cuenta el tiempo de desconexión por mantenimiento), este porcentaje de disponibilidad se calcula teniendo en cuenta la capacidad de soportar eventos como mantenimientos, interrupciones del servidor, errores en código y aumento en el tráfico. Este 98% anual se basa en que Django y nuestra empresa proveedora del hosting tienen una alta confiabilidad como empresa por su infraestructura la cual brinda más del 99% de disponibilidad al año.

\*Nota: la disponibilidad será medida mediante la página gratuita llamada “UptimeRobot” la cual nos ayudará a calcular el porcentaje de disponibilidad que posee nuestra aplicación web y notificar los eventos como las caídas de la página entre otros. Por otra parte, esta también nos ayudara a evidenciar la capacidad de nuestra página web. Estas pruebas serán apoyo a los criterios singulares especificados en otras secciones [24].

## Atributos del sistema software

**Disponibilidad:**

Tabla 9. Atributos del Sistema Disponibilidad

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ***# Requisito*** | ***P-1*** | ***Tipo de requisito*** | | ***No Funcional*** | ***Casos de uso asociados*** | |  |
| ***Descripción*** | ***El servidor de la aplicación estará disponible las 24 horas del día 6 días a la semana.*** | | | | | | |
| ***Razón*** | ***Para que el usuario pueda ingresar a la página web en cualquier momento del día.*** | | | | | | |
| ***Autor*** | ***Todos los integrantes del grupo.*** | | | | | | |
| ***Criterio de medición*** | ***Ingresando a la página web a cualquier hora del día durante los 6 días hábiles.*** | | | | | | |
| ***Prioridad*** | ***Alta.*** | | ***Módulo asociado*** | | |  | |
| ***Versión*** | ***1.0.0*** | | ***Fecha*** | | | ***08/10/2020*** | |

**Confiablidad:**

Tabla 10. Atributos del Sistema Confiabilidad

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ***# Requisito*** | ***R-1*** | ***Tipo de requisito*** | | ***No Funcional*** | ***Casos de uso asociados*** | |  |
| ***Descripción*** | ***Si las peticiones request/response tardan demasiado en dar respuesta, el sistema deberá ofrecer un tiempo de tolerancia a dicha respuesta.*** | | | | | | |
| ***Razón*** | ***Para poder ofrecer un tiempo de holgura a los retrasos de conexión por parte del servidor.*** | | | | | | |
| ***Autor*** | ***Todos los integrantes del grupo.*** | | | | | | |
| ***Criterio de medición*** | ***Teniendo una velocidad baja de red e intentar cargar la página.*** | | | | | | |
| ***Prioridad*** | ***Alta.*** | | ***Módulo asociado*** | | |  | |
| ***Versión*** | ***1.0.0*** | | ***Fecha*** | | | ***08/10/2020*** | |

**Mantenibilidad:**

Tabla 11. Atributos del Sistema Mantenibilidad

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ***# Requisito*** | ***S-1*** | ***Tipo de requisito*** | | ***No Funcional*** | ***Casos de uso asociados*** | |  |
| ***Descripción*** | ***Los archivos del software estarán en un sistema de control de versiones compartida con todos los miembros de desarrollo.*** | | | | | | |
| ***Razón*** | ***Para el adecuado mantenimiento de los archivos fuente que componen el software, el sistema de control de versiones permite realizar un mantenimiento colectivo (scrum team).*** | | | | | | |
| ***Autor*** | ***Todos los integrantes del grupo.*** | | | | | | |
| ***Criterio de medición*** | ***Si la máquina de cada uno de los miembros adquiere correctamente las últimas actualizaciones de los archivos compartidos.*** | | | | | | |
| ***Prioridad*** | ***Alta.*** | | ***Módulo asociado*** | | |  | |
| ***Versión*** | ***1.0.0*** | | ***Fecha*** | | | ***08/10/2020*** | |

**Portabilidad:**

Tabla 12. Atributos del Sistema Portabilidad

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ***# Requisito*** | ***O-1*** | ***Tipo de requisito*** | | ***No Funcional*** | ***Casos de uso asociados*** | |  |
| ***Descripción*** | ***Todos los archivos que constituyen el software se empaquetaran en un contenedor Docker.*** | | | | | | |
| ***Razón*** | ***Para distribuir o transportar adecuadamente el software.*** | | | | | | |
| ***Autor*** | ***Todos los integrantes del grupo.*** | | | | | | |
| ***Criterio de medición*** | ***En una maquina ajena se utilizará el orquestador de contenedores llamado Kubernetes para poner en funcionamiento el software que contiene el Docker.*** | | | | | | |
| ***Prioridad*** | ***Alta.*** | | ***Módulo asociado*** | | |  | |
| ***Versión*** | ***1.0.0*** | | ***Fecha*** | | | ***08/10/2020*** | |

**Seguridad:**

Tabla 13. Atributos del Sistema Seguridad

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ***# Requisito*** | ***[O-3, O-7]*** | ***Tipo de requisito*** | | ***No Funcional*** | ***Casos de uso asociados*** | |  |
| ***Descripción*** | ***Al usar el framework Django se hará uso de las funciones que garantizan la seguridad del sistema (revisar sección 7.5).* Django utiliza el algoritmo PBKDF2 con un hash SHA256.** | | | | | | |
| ***Razón*** | ***Para proteger la integridad de los datos de los usuarios.*** | | | | | | |
| ***Autor*** | ***Todos los integrantes del grupo.*** | | | | | | |
| ***Criterio de medición*** | ***Deltasoft no cuenta con los recursos necesarios para medir este requisito.*** | | | | | | |
| ***Prioridad*** | ***Alta.*** | | ***Módulo asociado*** | | |  | |
| ***Versión*** | ***1.0.0*** | | ***Fecha*** | | | ***08/10/2020*** | |

## Requisitos de la base de datos

La base de datos que va a ser usada por GDEvlink será postgress la cual se va a conectar con el framework de Django. Se cuenta con una herramienta propia del framework donde es necesario crean un súper usuario para poder ingresar a al apartado “/admin”, el cual nos va a ayudar a gestionar de manera gráfica nuestros modelos y datos correspondientes a cada aplicación. A continuación, veremos los tipos de datos que vamos a usar y características de cada modelo de nuestro proyecto.

* **ArrayField**: es un tipo de dato que permite crear y almacenar una lista de datos la cual se puede especificar el tipo de datos que se van a almacenar y su tamaño [22].
* **CharField**: es un tipo de dato el cual almacena una cadena corta de caracteres, en el cual se puede especificar su tamaño y otras opciones [22].
* **ImagenField**: es un tipo de dato el cual es capaz de almacenar imágenes de tamaño grande el cual tiene como parámetros, el ancho, el alto, la referencia y el tamaño de la imagen a ingresar [22].
* **DateTimeField**: es un tipo de dato el cual permite almacenarla fecha. Esta puede ser con dos atributos “Auto\_now\_add” que corresponde a la fecha de la creación del objeto y “Auto\_now” que corresponde a la fecha cada vez que se guarde el campo [22].
* **ForeingKey**: es un tipo de dato el cual es una referencia a otro modelo donde tiene como parámetros dos campos el primero va a ser el nombre de este atributo y el segundo es si esta llave foránea posee integridad referencial [22].

Nuestra aplicación web tendrá un flujo alto de consultas debido a su constante interacción e intercambio de información entre usuarios, por esto GDevLink contará con una frecuencia bastante elevada de acceso a la base de datos. La base de datos alojada en PostgreSQL planea tener un máximo de 30 conexiones abiertas sin que el rendimiento de la base de datos se vea afectado, en estas conexiones los principales tipos de consultas van a ser las de actualización, modificación y creación.

La forma en la cual se va a almacenar los datos de nuestra aplicación será mediante la creación de modelos en el framework de Django, que automáticamente se traduce en una ejecución de código sql automático por parte del framework creando las tablas en postgres. Cuando tenemos estos modelos creados es posible realizar consultas teniendo como parámetro la información ingresada por el usuario pasando por las capas del framework el cual cuenta con arquitectura MVC. este flujo se da del usuario al controlador del controlador al modelo donde este ejecuta búsquedas en la técnica de programación ORM para que sean traducidas al sql y se devuelva el resultado hasta la vista y el controlador lo pueda mostrar al usuario.

En la ilustración 1 se pueden ver las relaciones básicas que tendrán las entidades en la base de datos.

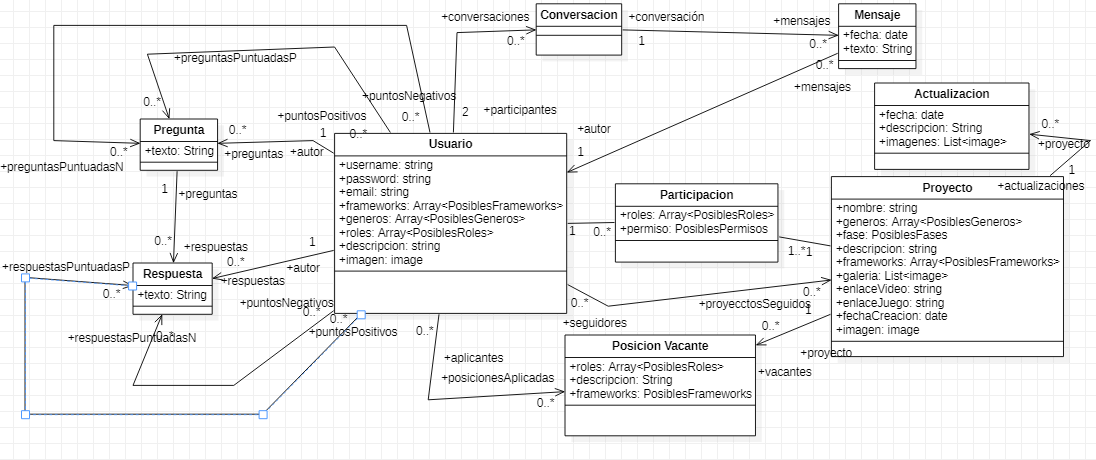


Ilustración Diagrama indexación de dato

# Proceso de ingeniería de requisitos

## Planeación

El proceso de ingeniería de requisitos debe contener un conjunto de actividades que permitan poder descubrir, analizar, documentar, validar, y mantener los requisitos funcionales y no funcionales de un sistema [25]. El proceso de ingeniería de requisitos de Gdevlink consiste en tres fases o actividades principales:

1. Levantamiento: Consiste en identificar y reunir las necesidades o funciones básicas del sistema, de acuerdo con las necesidades de los Stakeholders y los clientes (equipo de desarrollo). Estos son posteriormente analizados y formalizados como los requisitos que debe tener el sistema [25].
2. Especificación: Trata de especificar en detalle los requisitos establecidos en el levantamiento.
3. Verificación: Implica comprobar que los requisitos identificados definan el sistema como lo quieren los clientes y los Stakeholders. Está fase funciona junto con la fase de levantamiento, para verificar la validez de los requisitos formalizados del sistema [26].

Cada actividad tendrá uno o más encargados de realizarlas, como se detalla en la tabla 14. Sin embargo, todos los miembros del equipo tendrán la labor de revisar el resultado final de cada actividad.

|  |  |
| --- | --- |
| Actividad | Encargados |
| Levantamiento | Todos los miembros del equipo |
| Especificación | Líder de desarrollo y líder de calidad |
| Validación y Verificación | Líder de configuración |

Tabla Encargados procesos de requisitos

Nuevos requisitos del sistema son descubiertos o los existentes pueden ser entendidos de otra forma, ya sea por un miembro del equipo, o los clientes y *Stakeholders*, durante el desarrollo. Por ello, el proceso de requisitos debe ser repetido antes de la planeación del siguiente sprint. Además, el grupo de desarrollo debe discutir que tan factible puede ser cada requisito.

## Levantamiento

El proceso de levantamiento de requisitos tiene el propósito de tanto descubrir las necesidades de los clientes como entender el contexto en que el sistema va a operar. Para ello se pueden usar técnicas como las tradicionales y las colaborativas. Las tradicionales incluyen cuestionarios, entrevistas, y análisis de documentación existente. En cuanto a las colaborativas están las lluvias de ideas [27]. Para este proyecto, como el equipo de desarrollo hace a su vez el rol de los clientes, se decidió realizar en reuniones en las que participan todos los miembros equipo, una lluvia de ideas para determinar que necesidades o funcionalidades que debería satisfacer el proyecto.

La fase de levantamiento de requisitos está compuesta a su vez por 3 actividades:

1. Descubrimiento de los requisitos: Consta de realizar una reunión con todos los miembros del equipo, en donde se hace una lluvia de ideas en donde se identifican las necesidades que debe satisfacer el proyecto. Estas necesidades son discutidas con respecto a su factibilidad y si el equipo decide incluirla, está es documentada solo con su descripción. Luego es validada y verificada (Por todo el equipo, pero en mayor medida el líder de configuración) según la sección 10, para ver si la necesidad o requisito es aprobado, o debe ser descartado o replanteado. Además, en la reunión también se discuten las necesidades expresadas por el Stakeholder.
2. Clasifiación de los requisitos: Para la clasificación de los requisitos, se tienen 3 puntos de vista: Interacción, Indirectos, y de dominio. Los de interacción representan otras personas o sistemas que interactúan directamente con GDevlink. Los indirectos representan a los stakeholders que influencian los requisitos. Y los puntos de vista de dominio se relacionan con restricciones organizacionales como la persistencia de la información, físicas como restricciones en la red, o regulatorias [28]. Después de ser descubiertos los requisitos, estos son clasificados, esta actividad se hace en la misma reunión de descubrimiento, y participan todos los miembros del equipo.
3. Documentación de los requisitos: Después de que se hallan definido y clasificado los requisitos, estos son escritos en un documento con una breve descripción. Actividad realizada por el líder de proyecto.

## Especificación

A continuación, se consignarán todos los requisitos de nuestra aplicación web, que facilitarán el entendimiento a los requerimientos de la aplicación por parte de los desarrolladores y la actualización de estos a medida que estos requisitos vayan siendo desarrollados o sean cumplidos en su totalidad. Esta tarea será encargada a todos los miembros del grupo de desarrollo. Esta sección de especificación de requisitos será actualizada cada vez que se identifiquen nuevos requerimientos o modificar algún existente. Categorías: Tipo: funcionales (F), no funcionales (NF), Prioridad: Baja, Media, Alta.

**Nota:** Se desplegarán los primeros requisitos en las siguientes 5 paginas acatando el tamaño sugerido, como la cantidad de requisitos desborda las 5 páginas, se adjunto un anexo llamado “Especificación de requisitos” donde se encontrarán todos los requisitos.

**ID: RF001**

**Requisito: El** sistema debe permitir que el usuario cree una cuenta con un perfil asociado.

**Especificación y restricciones:** El usuario podrá crear una cuenta creando un nombre de usuario único y creando una contraseña alfanumérica, con mayúsculas que supere los ocho caracteres para su mayor seguridad.

**Origen del requisito:** Cliente.

**Tipo:** Funcional.

**Caso de uso asociado:** 1) Como USUARIO que NO ha hecho LOG IN, quiero REGISTRARME para CREAR UNA CUENTA-PERFIL.

**Prioridad:** Alta.

**Estado:** Especificado

**Aspecto de aceptación:** Este requisito será aceptado creando cuentas piloto para verificar si la creación de estas se realiza de manera satisfactoria.

**Versión:** 1.1

**ID: RF002**

**Requisito:** El sistema debe permitir que el usuario se autentique al momento de iniciar sesión.

**Especificación y restricciones:** El usuario podrá autenticarse con nombre de usuario y contraseña de manera correcta para acceder a su cuenta anteriormente creada manteniendo sus respectiva información y credenciales.

**Origen del requisito:** Cliente.

**Tipo:** Funcional.

**Caso de uso asociado:** 2) Como USUARIO que NO ha hecho LOG IN, quiero hacer LOG IN para ACCEDER A FUNCIONALIDADES DE USUARIO.

**Prioridad:** Alta.

**Estado:** Especificado

**Aspecto de aceptación:** Este requisito será aceptado autenticando usuarios de prueba verificando si el servidor persiste los datos ingresados y comprueba su existencia.

**Versión:** 1.1

**ID: RF002**

**Requisito:** El sistema debe permitir que el usuario se autentique al momento de iniciar sesión.

**Especificación y restricciones:** El usuario podrá autenticarse con nombre de usuario y contraseña de manera correcta para acceder a su cuenta anteriormente creada manteniendo sus respectiva información y credenciales.

**Origen del requisito:** Cliente.

**Tipo:** Funcional.

**Caso de uso asociado:** 2) Como USUARIO que NO ha hecho LOG IN, quiero hacer LOG IN para ACCEDER A FUNCIONALIDADES DE USUARIO.

**Prioridad:** Alta.

**Estado:** Especificado

**Aspecto de aceptación:** Este requisito será aceptado autenticando usuarios de prueba verificando si el servidor persiste los datos ingresados y comprueba su existencia.

**Versión:** 1.1

**ID: RF003**

**Requisito:** El sistema debe permitir que el usuario cierre sesión.

**Especificación y restricciones:** El usuario podrá cerrar la sesión que estaba activa mediante la función cerrar sesión.

**Origen del requisito:** Cliente.

**Tipo:** Funcional.

**Caso de uso asociado:** 3) Como USUARIO que ha hecho LOG IN, quiero hacer LOG OUT, para CERRAR MI PERFIL

**Prioridad:** Media.

**Estado:** Verificado.

**Aspecto de aceptación:** Este requisito será aceptado si al ejecutar la función cerrar sesión, no tendrá acceso a las funcionalidades para usuarios autenticados.

**Versión:** 1.2

**ID: RF004**

**Requisito:** El sistema debe permitir que el usuario modifique la información de su perfil.

**Especificación y restricciones:** El usuario podrá editar atributos como, roles, descripción, imagen, frameworks y géneros, de manera exitosa.

**Origen del requisito:** Cliente.

**Tipo:** Funcional.

**Caso de uso asociado:** 5) Como USUARIO que ha hecho LOG IN, quiero EDITAR MI PERFIL, para ACTUALIZAR MI INFORMACION.

**Prioridad:** Media.

**Estado:** Verificado.

**Aspecto de aceptación:** Este requisito será aceptado si al editar campos del perfil de un usuario, pueda ser visualizado el perfil actualizado.

**Versión:** 1.1

**ID: RF005**

**Requisito:** El sistema debe permitir que el usuario vea su perfil.

**Especificación y restricciones:** el usuario podrá ver sus campos de información actual, correspondiente a su perfil.

**Origen del requisito:** Cliente.

**Tipo:** Funcional.

**Caso de uso asociado:** 4) Como USUARIO que ha hecho LOG IN, quiero ENTRAR A MI PERFIL, para VER MI INFORMACIÓN.

**Prioridad:** Media.

**Estado:** Verificado.

**Aspecto de aceptación:** Este requisito será aceptado si al crear perfil y completar los campos relacionados a este, puedan ser vistos por el propio usuario en cualquier momento.

**Versión:** 1.1

**ID: RF006**

**Requisito:** El sistema debe permitir que el usuario vea los perfiles de otros usuarios.

**Especificación y restricciones:** el usuario podrá ver los campos de información como roles, descripción, imagen, frameworks y géneros, correspondientes al perfil de otro usuario.

**Origen del requisito:** Cliente.

**Tipo:** Funcional.

**Caso de uso asociado:** 6) Como USUARIO, quiero ENTRAR AL PERFIL de OTROS USUARIOS, para VER SU INFORMACIÓN.

**Prioridad:** Media.

**Estado:** Verificado.

**Aspecto de aceptación:** Este requisito será aceptado si al, buscar el perfil de un usuario en específico, la información de sus campos será renderizado de manera correcta al usuario

**Versión:** 1.1

**ID: RNFU-1**

**Requisito:** Se mostrarán mensajes que les indiquen a los usuarios las acciones que pueden realizar en cada una de las páginas y como acceder a ellas.

**Especificación y restricciones:** El usuario tendrá una guía, a partir de mensajes de cómo realizar la acción de manera correcta en cada una de las páginas.

**Origen del requisito:** Scrum team.

**Tipo:** No funcional.

**Caso de uso asociado:**

**Prioridad:** Baja.

**Estado:** Verificado.

**Aspecto de aceptación:** Este requisito será aceptado siempre y cuando, el usuario reciba un mensaje indicando el uso correcto de la funcionalidad que está realizando.

**Versión:** 0.1

**ID: RNFU-2**

**Requisito:** En las páginas de registro, creación de proyecto, creación de posición vacante y creación de actualización de progreso se le indicarán al usuario cuáles son los campos obligatorios.

**Especificación y restricciones:** El software debe estar en la capacidad de mostrarle usuario que campos son obligatorios al momento de crear proyectos, posiciones vacantes, actualizaciones de progreso y página de registro.

**Origen del requisito:** Scrum team.

**Tipo:** No funcional.

**Caso de uso asociado: 1)** Como USUARIO que NO ha hecho LOG IN, quiero REGISTRARME para CREAR UNA CUENTA-PERFIL, 7) Como USUARIO que ha hecho LOG IN, quiero CREAR UN PROYECTO, para REPRESENTAR MI PROYECTO y CONVERTIRME EN SU ADMINISTRADOR MAESTRO, 11) Como ADMINISTRADOR DE PROYECTO, quiero PUBLICAR las últimas actualizaciones para COMPARTIR EL PROGRESO DEL PROYECTO, 18) Como ADMINISTRADOR de proyecto, quiero CREAR UNA POSICION VACANTE, para ENCONTRAR un nuevo MIEMBRO DEL EQUIPO

**Prioridad:** Baja.

**Estado:** Verificado.

**Aspecto de aceptación:** Este requisito será aceptado siempre y cuando, el usuario visualice cuales campos en el formulario que este llenando son de carácter obligatorio.

**Versión:** 0.1

**ID: RNFU-3**

**Requisito:** En caso de que ocurra un problema en el registro o en la creación de un proyecto, se le indicará al usuario la razón del problema (por ejemplo, indicar que el usuario ingresado ya existe).

**Especificación y restricciones:** El software debe estar en la capacidad de validar el correcto registro de un proyecto o usuario nuevo, al encontrar una inconsistencia el software deberá advertirle al usuario, que su ingreso fue invalido.

**Origen del requisito:** Scrum team.

**Tipo:** No funcional.

**Caso de uso asociado: 1)** Como USUARIO que NO ha hecho LOG IN, quiero REGISTRARME para CREAR UNA CUENTA-PERFIL, 7) Como USUARIO que ha hecho LOG IN, quiero CREAR UN PROYECTO, para REPRESENTAR MI PROYECTO y CONVERTIRME EN SU ADMINISTRADOR MAESTRO**.**

**Prioridad:** Media.

**Estado:** Verificado.

**Aspecto de aceptación:** Este requisito será aceptado siempre y cuando, el usuario ingrese de manera errónea los datos del proyecto o usuario y este reciba una alerta con el error.

**Versión:** 0.1

**ID: RNFP-1**

**Requisito:** El servidor tendrá capacidad de cómputo de al menos 1000 horas al mes.

**Especificación y restricciones:** El software debe estar en la capacidad prestar un servicio integro, mientras que no se supere la capacidad mensual del servidor.

**Origen del requisito:** Scrum team.

**Tipo:** No funcional.

**Caso de uso asociado:** N/A**.**

**Prioridad:** Media.

**Estado:** Verificado.

**Aspecto de aceptación:** Este requisito será aceptado siempre y cuando, el usuario ingrese en cualquier momento del día y esta aplicación web se encuentre disponible para el uso, y el servidor sigue teniendo capacidad de cómputo.

**Versión:** 0.1

\*Nota: para ver más detalle abrir el anexo llamado “9.3 Especificación - Archivo adjunto esquema de prioridad y requisitos.”

# Proceso de verificación y validación

Para cumplir con todos los requisitos identificados en nuestra aplicación realizaremos el debido proceso de validación y verificación para cada uno de los requisitos tanto funcionales como no funcionales, incluyendo diagramas de flujo de estos procesos y enfocándonos en los más importantes. El resto se especificará en un archivo anexo llamado “10. Anexo de proceso de verificación y validación”. Para esto identificaremos cada requisito con su respectivo ID como se hizo en la sección 9.3 y se hará su debido proceso de validación y verificación.

**Nota:** Se desplegarán los primeros procesos de verificación y validación en las siguientes páginas acatando el tamaño sugerido, como la cantidad de procesos saturaría el documento, se adjuntó un anexo llamado “procesos de verificación y validación”.

**ID: RF002**

**Proceso de validación:** Para poder validar este requisito el usuario podrá autentificarse con su cuenta entrando a su página de inicio, esto se cumplirá mediante pruebas unitarias, creando usuarios de prueba para realizar el debido proceso de autentificación. Este será un requisito que es calificado como comprensible, no ambiguo, realizable, verificable y modificable.

**Proceso de verificación:** Para que se pueda cumplir con este requisito tendrá que llenar su nombre de usuario y su contraseña de manera correcta, su contraseña tendrá que estar compuesta por más de 8 caracteres**,** además este usuario que quiera autentificarse deberá haber hecho previamente el proceso de registro de nuestra página web.

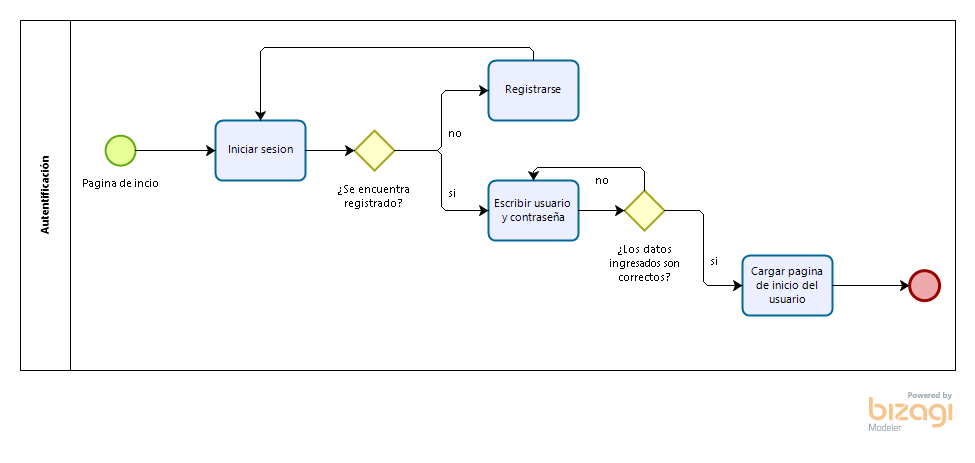


Ilustración 2. Proceso de verificación y validación login

**ID: RF009**

**Proceso de validación:** Para poder validar este requisito el usuario que este autentificado podrá crear un proyecto, este se cumplirá mediante pruebas unitarias creando proyectos piloto. Este será un requisito que es calificado como no ambiguo, realizable, verificable, modificable, trazable y de alta importancia.

**Proceso de verificación:** Para que se pueda cumplir con la verificación de este requisito tendrá que llenar correctamente cada uno de los campos obligatorios que corresponden a la creación del proyecto. Posteriormente el usuario podrá visualizar el proyecto creado en la información de su perfil.

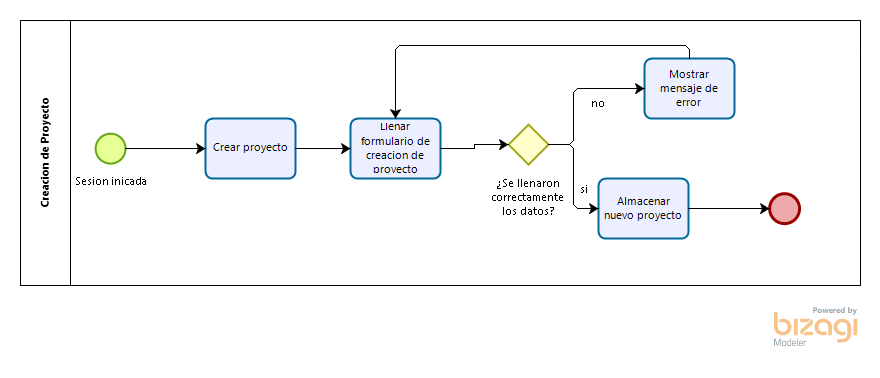


Ilustración 3. Proceso de verificación y validación Creación Proyecto

**ID: RNF003**

**Proceso de validación:** Para poder validar este requisito el usuario podrá visualizar los errores que cometió al momento de registrarse o al momento de crear un proyecto, esto se cumplirá llenando los campos obligatorios de una manera errónea a propósito para que el usuario pueda caer en cuenta de su error. Este será un requisito que es calificado como preciso, no redundante, realizable, verificable y modificable.

**Proceso de verificación:** Para que se pueda cumplir con este requisito se tendrá que comprobar uno a uno los campos que el usuario llena mal y que el sistema muestre cual es el campo y/o los campos que el usuario ha llenado de manera errónea.

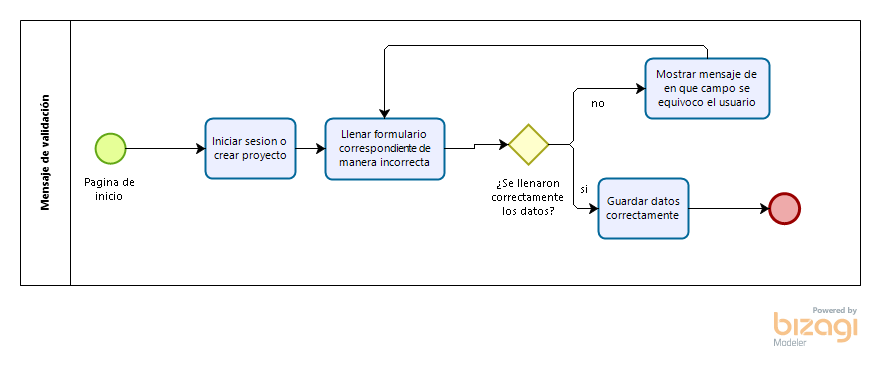


Ilustración 4. Proceso de verificación y validación de errores

**ID: RNF006**

**Proceso de validación:** Para poder validar este requisito el usuario podrá ingresar a nuestra página web sin importar que sistema operativo tenga su dispositivo electrónico. Este será un requisito que es calificado como preciso, no redundante, realizable, verificable y consistente.

**Proceso de verificación:** Para que se pueda cumplir con la verificación de este requisito se tendrá que comprobar el ingreso a nuestra página web desde diferentes dispositivos con distintos sistemas operativos comprobando así su buen funcionamiento.

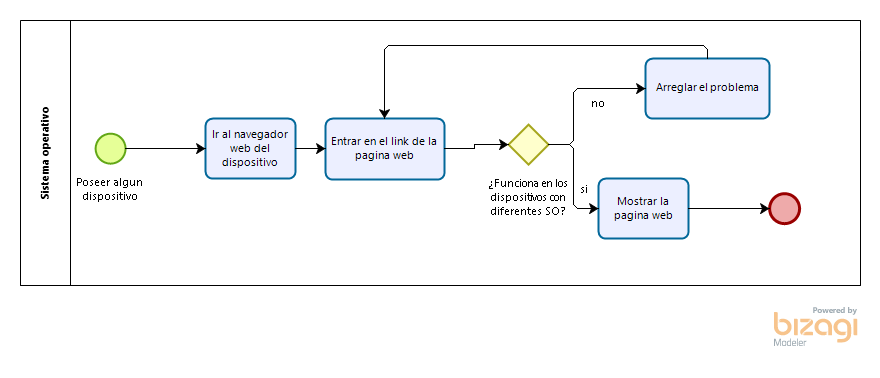


Ilustración 5. Proceso de verificación y validación sistema operativo

**ID: RNF008**

**Proceso de validación:** Para poder validar este requisito el equipo de desarrollo deberá tener los archivos de la aplicación en el repositorio de github. Este será un requisito que es calificado como preciso, no redundante, realizable, verificable y modificable.

**Proceso de verificación:** Para que se pueda cumplir con este requisito se tendrá que comprobar en el repositorio que todos los integrantes del grupo tengan acceso a estos archivos permitiendo un sistema de control de versiones que la herramienta nos proporciona.

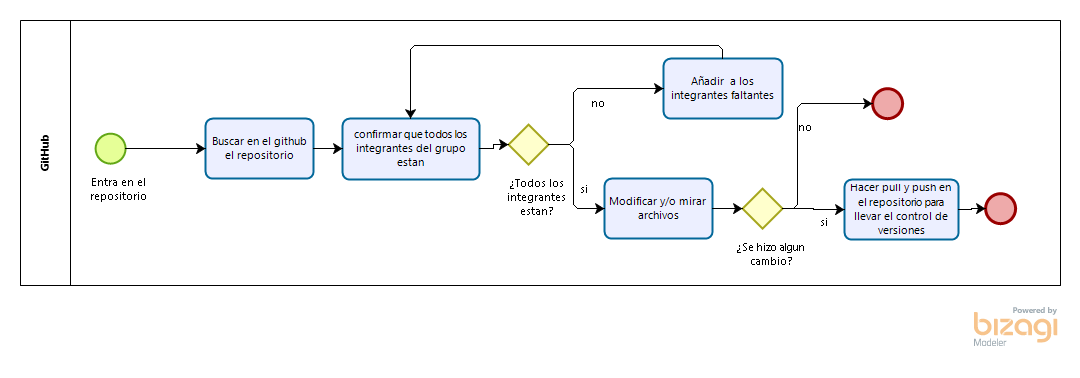


Ilustración 6. Proceso de verificación y validación repositorio

# Anexos

1. Diagrama Modelo de dominio
2. Documentación de modelo dominio
3. Diagrama de clases
4. Diagrama de casos de uso
5. Especificación de casos de uso y mockups
6. Reporte gerencial
7. Características del producto de software
8. Especificación de requisitos
9. Procesos de verificación y validación

# Referencias

|  |  |
| --- | --- |
| [1] | R. Monferrer, Especificación de Requisitos Software según el estándar de IEEE 830, 2001. |
| [2] | L. C., Applying UML and Patterns, Prentice Hall, 2002. |
| [3] | W3C, «Standards,» 2019. [En línea]. Available: https://www.w3.org/standards/. |
| [4] | Google, «Using OAuth 2.0 to Access Google APIs,» Google Developers, 6 Octubre 2020. [En línea]. Available: https://developers.google.com/identity/protocols/oauth2. |
| [5] | Google, «Google Sign-in,» Google Developers, 14 Septiembre 2020. [En línea]. Available: https://developers.google.com/identity/sign-in/web/sign-in. |
| [6] | O. Bouhaj, «Browser Performance Impact,» 26 3 2019. [En línea]. Available: https://www.go-euc.com/browser-performance-impact/. [Último acceso: 2020]. |
| [7] | Google, «Descargar e instalar Google Chrome,» 2020. [En línea]. Available: https://support.google.com/chrome/answer/95346?co=GENIE.Platform%3DiOS&hl=es&oco=2. |
| [8] | Google, «Requisitos del sistema del navegador Chrome,» 2020. [En línea]. Available: https://support.google.com/chrome/a/answer/7100626?hl=es. |
| [9] | Microsoft, «Requisitos del sistema de Windows 7,» 11 8 2020. [En línea]. Available: https://support.microsoft.com/es-co/help/10737/windows-7-system-requirements#:~:text=Procesador%20a%201%20gigahercio%20(GHz,controlador%20WDDM%201.0%20o%20superior. |
| [10] | Microsoft, «Requisitos del sistema de Windows 10,» 2020. [En línea]. Available: https://support.microsoft.com/es-es/windows/requisitos-del-sistema-de-windows-10-6d4e9a79-66bf-7950-467c-795cf0386715. |
| [11] | Apple Inc, «OS X Yosemite - Especificaciones técnicas,» 27 6 2017. [En línea]. Available: https://support.apple.com/kb/SP711?locale=es\_ES. [Último acceso: 2020]. |
| [12] | C. Iglesias, «Finalmente, lanzamiento de Ubuntu 14.04,» 17 4 2014. [En línea]. Available: https://lignux.com/finalmente-lanzamiento-de-ubuntu-14-04-trusty-tahr/. [Último acceso: 2020]. |
| [13] | Cazador, «Requisitos mínimos para uso fluido de Android 5 - Lollipop,» 17 5 2015. [En línea]. Available: https://www.elsate.com/viewtopic.php?t=33. [Último acceso: 2020]. |
| [14] | Hostinger, «Web Hosting,» 2020. [En línea]. Available: https://www.hostinger.co/hosting-web. |
| [15] | SmarterASP.NET, «Superior ASP.NET Hosting,» 2020. [En línea]. Available: https://www.smarterasp.net/index. |
| [16] | SG, «ag.com.mx,» [En línea]. Available: https://sg.com.mx/revista/15/reglas-negocio-administrando-la-operacion-reglas. [Último acceso: 12 10 2020]. |
| [17] | PMOinformatica, «pmoinformatica.com,» 06 04 2015. [En línea]. Available: http://www.pmoinformatica.com/2015/05/requerimientos-no-funcionales-ejemplos.html#:~:text=Los%20requerimientos%20no%20funcionales%20representan,sistema%20que%20se%20est%C3%A9%20desarrollando.&text=Entre%20los%20ejemplos%20de%20requerimientos,dependibilida. [Último acceso: 12 10 2020]. |
| [18] | Uniwebsidad, «uniwebsidad.com,» 2006. [En línea]. Available: https://uniwebsidad.com/libros/django-1-0/capitulo-20/escalabilidad. [Último acceso: 12 10 2020]. |
| [19] | MERKLE, «merkleinc.com,» 01 09 2020. [En línea]. Available: https://www.merkleinc.com/es/es/blog/hit#:~:text=Una%20p%C3%A1gina%20web%20esta%20formada,un%20archivo%20es%20un%20hit.. [Último acceso: 13 10 2020]. |
| [20] | LINUBE, «linube.com,» 13 11 2019. [En línea]. Available: https://linube.com/blog/contenedor-software/. [Último acceso: 15 10 2020]. |
| [21] | Kubernetes, «kubernetes.io,» 16 06 2020. [En línea]. Available: https://kubernetes.io/es/docs/concepts/overview/what-is-kubernetes/. [Último acceso: 15 10 2020]. |
| [22] | Django, «docs.djangoproject.com,» 2005. [En línea]. Available: https://docs.djangoproject.com/en/3.1/topics/security/. [Último acceso: 14 10 2020]. |
| [23] | Google, «Google developer,» [En línea]. Available: https://developers.google.com/speed/pagespeed/insights/. [Último acceso: 16 10 2020]. |
| [24] | Uptimerobot, «Uptimerobot,» [En línea]. Available: https://uptimerobot.com. [Último acceso: 16 10 2020]. |
| [25] | D. Pandey y U. Suman, «An Effective Requirement Engineering Process Model for Software Development and Requirements Management,» Noviembre 2010. [En línea]. Available: https://www.researchgate.net/publication/224202533\_An\_Effective\_Requirement\_Engineering\_Process\_Model\_for\_Software\_Development\_and\_Requirements\_Management. [Último acceso: 16 Octubre 2020]. |
| [26] | I. Sommerville, Software Engineering, Pearson, 2016. |
| [27] | A. Banaceur, T. Tun, Y. Yu y B. Nuseibeh, «HAl,» 4 Abril 2018. [En línea]. Available: https://hal.archives-ouvertes.fr/hal-01758502/document. [Último acceso: 16 Octubre 2020]. |
| [28] | I. Sommerville, «IFS,» [En línea]. Available: https://ifs.host.cs.st-andrews.ac.uk/Books/SE9/Web/Requirements/Viewpoints.html. [Último acceso: 16 Octubre 2020]. |
| [29] | Internet Engineering Task Force, «IETF Tools,» Junio 2014. [En línea]. Available: https://tools.ietf.org/html/rfc7230. |
| [31] | stackoverrun, «stackoverrun,» 30 06 2015. [En línea]. Available: https://stackoverrun.com/es/q/8569336. [Último acceso: 16 10 2020]. |
| [32] | stackoverrun, «stackoverrun,» 30 06 2015. [En línea]. Available: https://stackoverrun.com/es/q/8569336. [Último acceso: 16 10 2020]. |