**Especificación de requisitos de software**

**Proyecto: Gauchada**

Revisión:



**Ficha del documento**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Revisión** | **Autor** | **Verificado** |
|  |  |  |  |

Documento validado por las partes en fecha:

|  |  |
| --- | --- |
| Por el cliente | Por la empresa suministradora |
|  | Untitled.png |
| Aclaración: | Sin título-1.pngAclaración: |

**Contenido**

[Introducción](#_otpmt8buh1hx)

[Propósito](#_ggbzbvu07udj)

[Alcance](#_flvexnj3x9s6)

[Definiciones, acrónimos y abreviaturas](#_jl8d1too5ze)

[Referencias](#_datf8wf5huxt)

[Resumen](#_ekl5om3v6sir)

[Descripción general](#_790iyrcql999)

[Perspectiva del producto](#_7y7qdtmaggd3)

[Funcionalidad del producto](#_t9pg70luzzsc)

[Características de los usuarios](#_xbwnyfc6gpng)

[Restricciones](#_uzcyout483wy)

[Suposiciones y dependencias](#_an92x3mlsf32)

[Evolución previsible del sistema](#_yv6e7arzu4ya)

[Requisitos específicos](#_tnns0trksjfh)

[Requisitos comunes de los interfaces](#_b9sznc60w0a)

[Interfaces de usuario](#_hpi7plj3ohln)

[Interfaces de hardware](#_n889uu3t3d70)

[Interfaces de software](#_k0tlq79xvupm)

[Requisitos funcionales](#_4ya5581iwouv)

[Requisitos no funcionales](#_caoutxc6erud)

[Requisitos de rendimiento](#_zbmw6fjew912)

[Seguridad](#_n0x1e42prhxc)

[Fiabilidad](#_6inre4xef4jy)

[Disponibilidad](#_obj27jrs45qz)

[Mantenibilidad](#_gl8fcccjbb9e)

[Portabilidad](#_o10xc4bc8rrz)

[Otros requisitos](#_48pkkifei393)

[Apéndices](#_3x2fhimfvdkh)

# Introducción

Teniendo en cuenta las pautas e indicaciones dadas, el presente informe abarca las necesidades y requerimientos que fueron solicitados.

## Propósito

* Tiene como objetivo especificar información pertinente sobre el producto que se va a desarrollar.
* Este informe está dirigido a la Srta. Nancy Díaz.

## Alcance

* El producto a desarrollar se denominada *"Gauchada".*
* *Gauchada*, es una aplicación o página web en la cual se puede publicar favores.
* Dirigida a usuarios dentro del rango de Argentina.

## Definiciones, acrónimos y abreviaturas

* Los **usuarios registrados** en esta aplicación se clasifican en tres: por un lado el **Favorecido**, el cual pide un favor en la aplicación y está a la espera de que algún usuario se ofrezca a realizarla; este último se lo denomina **Voluntario**. Luego de que el Favorecido elija quién de los voluntarios va a realizar el favor, el elegido se convierte ahora en **Ayudante**.
* El patrón de arquitectura **MVC** (Modelo Vista Controlador) es un patrón que define la organización independiente del **Modelo** (Objetos de Negocio), la **Vista** (interfaz con el usuario u otro sistema) y el **Controlador** (controlador del workflow de la aplicación).
* **Ruby on Rails**, también conocido como **RoR** o **Rails**, es un framework de aplicaciones web de código abierto escrito en el lenguaje de programación Ruby, siguiendo el paradigma del patrón Modelo Vista Controlador (MVC).

## Referencias

|  |  |  |  |
| --- | --- | --- | --- |
| **Referencia** | **Título** | **Fecha** | **Autor** |
| - | Entrevista de elicitación. | 19/09/2016 | Master Branch |
| - | Cuestionario. | 19/09/2016 | Master Branch |

## Resumen

En el presente documento se detallarán las especificaciones de requerimientos del sistema para un correcto uso y desarrollo del mismo.

# Descripción general

## Perspectiva del producto

Es un producto independiente que, debido al crecimiento de los usuarios se debe migrar del blog a una plataforma web apropiada para el correcto funcionamiento.

## Funcionalidad del producto

Publicar la necesidad o favores en la página (para que la gente pueda ayudarlo). Solicitar y realizar el favor. Preguntar y responder sobre alguna duda o inquietud sobre el favor. Luego de cumplido el favor, se puede calificar si la ayuda aportada fue buena o mala. Esta calificación restará o sumará puntos, como corresponda, a la persona que entrega su ayuda. El Favorecido podrá dejar un comentario en el caso de que no le hayan hecho bien el favor.

## Características de los usuarios

|  |  |
| --- | --- |
| Tipo de usuario | Favorecido |
| Formación | Toda persona registrada, mayor a 18 (dieciocho) años, con residencia en la República Argentina. |
| Actividades | Pedir favores, elegir quién lo va a realizar. |

## 

|  |  |
| --- | --- |
| Tipo de usuario | Voluntario |
| Formación | Toda persona registrada, mayor a 18 (dieciocho) años. |
| Actividades | Ofrecimiento a realizar un favor. |

## 

|  |  |
| --- | --- |
| Tipo de usuario | Ayudante |
| Formación | Toda persona registrada, mayor a 18 (dieciocho) años. |
| Actividades | Realizador de un favor en particular. |

Cabe destacar que un usuario puede ser de los tres tipos anteriormente mencionados, pero no dentro de un mismo favor, por lo tanto Un favorecido puede ayudar a otros usuarios con diferentes favores.

## Restricciones

Las limitaciones a tener en cuenta a la hora de diseñar y desarrollar el sistema van a estar dadas por:

* Metodologías de desarrollo:
  + Patrón MVC (Model View Controller).
* Lenguaje de programación:
  + Ruby v2.3.1. Particularmente la utilización del framework “Ruby on Rails”, con Rails v 4.2.6.
* Restricciones de Hardware: N/A.
* Restricciones de Sistema Operativo: Utilizar navegador actualizado.

## Suposiciones y dependencias

No corresponde.

## Evolución previsible del sistema

A futuro está pensado crear un chat e implementar una versión para móviles (smartphone o tablet).

# Requisitos específicos

## Requisitos comunes de los interfaces

No corresponde.

### Interfaces de usuario

Se conservará el logo del antiguo logo del blog: <http://unagauchada.tumblr.com/>.

Para el diseño de la página se respetarán los colores del logo: tonos anaranjados.

En la página principal se verá un listado de todos los favores, donde para cada uno se verá logo o imagen en el caso de que se haya subido una.

### Interfaces de hardware

No corresponde

### Interfaces de software

No se utiliza ningún plugin o sistema de software para la página por el momento.

## Requisitos funcionales

Serán descritos como Historias de Usuario.

## Requisitos no funcionales

### Requisitos de rendimiento

* El 90 % de los posteos de favores deben realizarse en menos de 1 segundo.
* Se espera un máximo de (cantidad) de usuarios conectados simultáneamente.

Estos requisitos van a estar relacionados con el hardware que posea cada usuario a la hora de entrar en la página, que no debería de haber problema alguno o lentitud con cualquier hardware a la venta en la actualidad.

Así mismo, la conexión a internet establecida por el usuario con su proveedor afectará en la rapidez de respuesta de la misma pero no llegando a ser un problema funcional para el mismo.

### Seguridad

Para mantener la información segura de todos los usuarios:

* Única visibilidad a la información personal de cada usuario sólo la va a tener el administrador.
* Solamente se podrá ver la información de contacto entre usuarios una vez acuerden realizar un favor (entre Favorecido y Ayudante).
* Verificación de las cuentas de usuario mediante el mail.

### Fiabilidad

Se asegura una cantidad de incidentes de seguridad menor o igual a uno por mes durante los primeros tres meses de funcionamiento del sistema. Se asegura una cantidad de incidentes de caída del sistema menor o igual a tres por mes durante los primeros tres meses de funcionamiento del sistema.

### Disponibilidad

La disponibilidad de la plataforma será de 24 hs. Las únicas excepciones de esta disponibilidad son para mantenimiento o diversas acciones de los desarrolladores/administrador de la página con previo aviso al no estar disponible.

### Mantenibilidad

El mantenimiento será para solucionar los errores y conflictos que puede tener el sistema en el momento de usar la plataforma, como también incrementar la funcionalidad del mismo. Dicho mantenimiento se discutió en la entrevista pero se llegó a que más tarde se iba a confirmar.

### Portabilidad

Al ser un producto con plataforma web de por sí conlleva una portabilidad general en la cual se puede acceder mediante distintos dispositivos a la web “Gauchada”.

Además se puede prever a futuro un desarrollo de aplicación para smartphones.

El lenguaje de programación utilizado es “Ruby on Rails” que facilita la portabilidad de la plataforma.

## Otros requisitos

* Se puede pedir cualquier favor mientras esté en el marco legal.

# Apéndices

No corresponde.