Especificación de Requerimientos

**Descripción del Diseño**

**Ingenio**

Portal Web Ingenio - plataforma que busca acercar a todos aquellos ingenieros (sin importar su especialidad), que desean ofrecer sus servicios profesionales bien sea en modo freelance o con otro medio de contratación con aquellos potenciales clientes que están buscando a un profesional con ciertas características.

|  |  |  |
| --- | --- | --- |
| Apellidos, Nombres | Correo electrónico | Rol |
| Laura Daniela Pachón |  | Development Team |
| Melissa Friné Coral |  | Development Team |
| Wilson Jurado Valbuena |  | Product Owner |
| John Jairo Mack |  | Development Team |
| José Julián Jaramillo |  | Scrum Master |

Fecha de presentación: 07/Dic/2021

Contenido

[1 INTRODUCCIÓN 3](#_Toc77242272)

[1.1 Propósito 3](#_Toc77242273)

[1.2 Alcance o Ámbito del Sistema 3](#_Toc77242274)

[1.3 Definiciones, Acrónimos y Abreviaturas 3](#_Toc77242275)

[1.3.1 Definiciones 3](#_Toc77242276)

[1.3.2 Acrónimos 3](#_Toc77242277)

[1.3.3 Abreviaturas 3](#_Toc77242278)

[1.3.4 Referencias 4](#_Toc77242279)

[1.4 Perspectiva General del Documento 4](#_Toc77242280)

[2 DESCRIPCIÓN GENERAL DE LA APLICACIÓN 4](#_Toc77242281)

[2.1 Perspectiva de la Aplicación 4](#_Toc77242282)

[2.2 Funciones de la Aplicación 4](#_Toc77242283)

[2.3 Características de los Usuarios 5](#_Toc77242284)

[2.4 Restricciones 5](#_Toc77242285)

[2.5 Suposiciones y Dependencias 5](#_Toc77242286)

[2.6 Requerimientos Diferidos 5](#_Toc77242287)

[3 REQUERIMIENTOS ESPECÍFICOS 5](#_Toc77242288)

[3.1 Requerimientos 6](#_Toc77242289)

[3.1.1 Product Backlog 6](#_Toc77242290)

[3.1.2 Ciclo de Sprints del proyecto 6](#_Toc77242291)

[3.1.3 Sprint Backlog 6](#_Toc77242292)

[3.1.4 Historias de usuario (Tareas y Subtareas) 6](#_Toc77242293)

[3.1.5 Mecánica de organización del grupo. (Reuniones, evidencias/artefactos) 6](#_Toc77242294)

[3.2 Modelo de Requerimientos 7](#_Toc77242295)

[3.2.1 Modelo de Casos de Uso 7](#_Toc77242296)

[4 DESCRIPCIÓN DEL DISEÑO 9](#_Toc77242297)

[4.1 Interfaz gráfica (Mockups) 9](#_Toc77242298)

[5 Gestión de la configuración 9](#_Toc77242299)

[6 PRUEBAS 9](#_Toc77242300)

[6.1 Descripción de pruebas unitarias 9](#_Toc77242301)

[6.2 Descripción de pruebas de aceptación 9](#_Toc77242302)

[7 GLOSARIO 10](#_Toc77242303)

[8 ANEXO(S) 10](#_Toc77242304)

# Introducción

En el presente documento se describe la ingeniería de detalle del portal Web Ingenio que es una plataforma que busca acercar a todos aquellos ingenieros (sin importar su especialidad), que desean ofrecer sus servicios profesionales bien sea en modo freelance o con otro medio de contratación con aquellos potenciales clientes que están buscando a un profesional con ciertas características.

## Propósito

En este documento se explicará cuáles son las razones para la construcción del aplicativo “Ingenio”, cuál será su función, los servicios que ofrece, el público objetivo, las herramientas tecnológicas usadas en su construcción, así como conocer en forma general y específica los diferentes componentes.

Este documento va orientado a los tutores del curso Misión TIC 2021, Ciclo 4.

## Alcance o Ámbito del Sistema

El sitio Web Ingenio es una plataforma que busca acercar a todos aquellos ingenieros (sin importar su especialidad), que desean ofrecer sus servicios profesionales bien sea en modo freelance o con otro medio de contratación con aquellos potenciales clientes que están buscando a un profesional con ciertas características.

La plataforma permitirá a los profesionales (en su primer versión orientada a ingenieros) realizar un registro de su profesión, sus conocimientos, habilidades, servicios que ofrecen, así como de la tarifa que cobran, para que de esta manera quienes requieran un servicio específico puedan contactar de una forma simple y sencilla con uno o varios profesionales.

La aplicación no cobra dinero por cualquier servicio ofertado/contratado, ni es responsable por la calidad del servicio o pago realizado entre el profesional y el cliente. La mayor función del aplicativo es servir de “puente” entre la basta variedad de profesionales que ofertan sus servicios y la cada vez mayor demanda por estos.

## Definiciones, Acrónimos y Abreviaturas

### Definiciones

* estándares: IEEE Std-830-1998 (IEEE Recommended Practice for Software Requirements Specifications);
* IEEE Std-1016-2009 (IEEE Standard for Information Technology — Systems Design — Software Design Descriptions)
* ISO/IEC/IEEE 29148:2011(E) (Systems and software engineering — Life cycle processes — Requirements engineering).
* ISO/IEC 23026 (IEEE Std 2001-2002) (Software Engineering — Recommended Practice for the Internet — Web Site Engineering, Web Site Management, and Web Site Life Cycle)
* IEEE/EIA 12207.1-1997 (ISO/IEC 12207) (Standard for Information Technology—Software life cycle processes—Life cycle data).
* OMG Unified Modeling Language (OMG UML), Infrastructure, Version 2.4.1 (OMG Document Number: formal/2011-08-05
* Standard document URL: <http://www.omg.org/spec/UML/2.4.1/Infrastructure>)
* OMG Unified Modeling Language (OMG UML),.

### Acrónimos

* MongoDB
* REACT
* Heroku
* Mongo Atlas
* HTML
* CSS

### Abreviaturas

* IEEE - Institute of Electrical and Electronics Engineers - Instituto de Ingenieros Eléctricos y Electrónicos
* ISO - Organización Internacional de Normalización
* ISO (Organización Internacional de Normalización) e IEC (Comisión Electrotécnica Internacional
* IEEE/EIA 12207.1-1997 (ISO/IEC 12207) (Standard for Information Technology—Software life cycle processes—Life cycle data).
* OMG Unified Modeling Language (OMG UML).

### Referencias

* ISO/IEC 23026 (IEEE Std 2001-2002) (Software Engineering — Recommended Practice for the Internet — Web Site Engineering, Web Site Management, and Web Site Life Cycle)
* IEEE/EIA 12207.1-1997 (ISO/IEC 12207) (Standard for Information Technology—Software life cycle processes—Life cycle data).
* OMG Unified Modeling Language (OMG UML), Infrastructure, Version 2.4.1 (OMG Document Number: formal/2011-08-05
* Standard document URL: <http://www.omg.org/spec/UML/2.4.1/Infrastructure>)
* OMG Unified Modeling Language (OMG UML),.

## Perspectiva General del Documento

En el presente documento describe la ingeniería de detalle del portal Web Ingenio que es una plataforma que busca acercar a todos aquellos ingenieros (sin importar su especialidad), que desean ofrecer sus servicios profesionales.

En este documento se explicará cuál será su función del aplicativo “Ingenio”, los servicios que ofrece, el público objetivo, las herramientas tecnológicas usadas en su construcción, así como conocer en forma general y específica los diferentes componentes.

# Descripción general de la aplicación

Portal Web Ingenio es una plataforma que busca acercar a todos aquellos ingenieros (sin importar su especialidad), que desean ofrecer sus servicios profesionales bien sea en modo freelance o con otro medio de contratación con aquellos potenciales clientes que están buscando a un profesional con ciertas habilidades en temas de programación.

## Perspectiva de la Aplicación

Ingenio es una plataforma por medio de la cual los profesionales de ingeniería pueden ofertar sus servicios de una manera simple, sencilla, cómoda, rápida y lo que es mucho mejor, gratuitamente.

Conocedores del enorme potencial que brindan las nuevas tecnologías y teniendo como marco de referencia la pandemia global del Covid-19, que llevó a muchos trabajadores a realizar sus labores desde casa, se presenta esta plataforma por medio de la cual los ingenieros ofertaran sus servicios, capacidades y conocimientos adquiridos a lo largo del ejercicio de su actividad profesional, para que aquellos potenciales clientes que estén buscando un servicio específico, una solución a un problema planteado o simplemente deseen contratar a uno de ellos, puedan hacerlo de una manera ágil y eficiente.

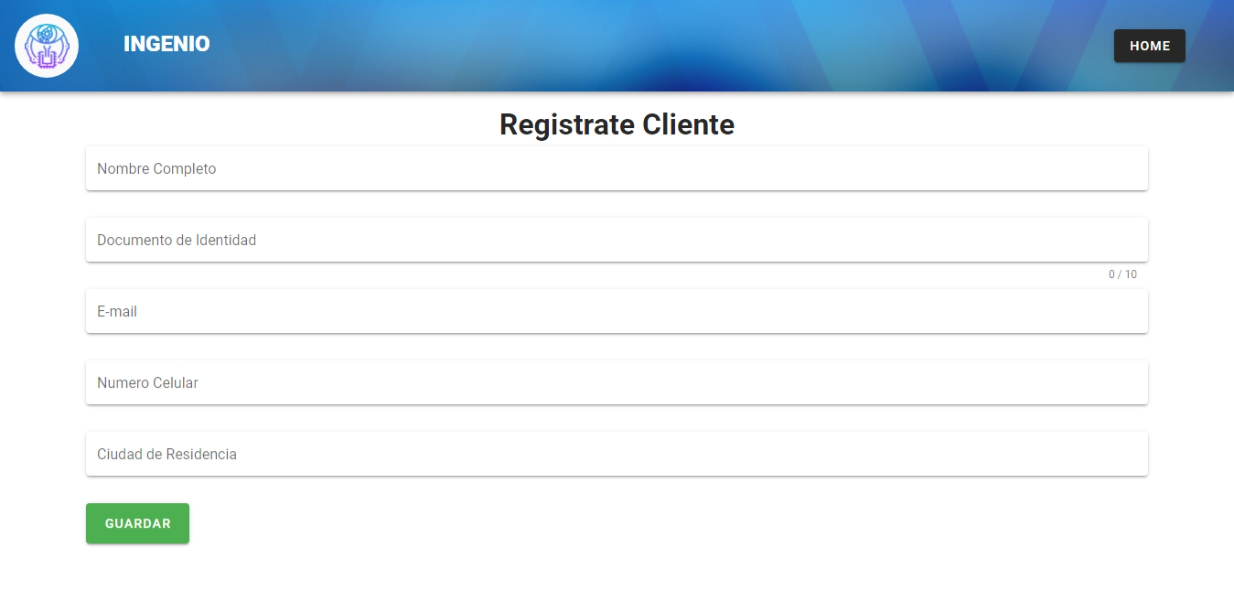
La plataforma está diseñada para ser lo más intuitiva posible, de tal manera que su uso pueda ser potenciado y brinde las mejores soluciones a todos los interesados.

Ingenio viene a ser una suerte de amalgama entre conceptos tales como mercado libre u Olx y redes de contacto tales como Linkedin. Solo que en esta ocasión estará enfocada directamente a suplir la creciente demanda entre los clientes que requieren un profesional con dedicación total, parcial o freelance.

## Funciones de la Aplicación

Dentro de las funciones que tendrá la aplicación se cuenta:

Registro:



Proceso mediante el cual un ingeniero o profesional realiza el registro, indicando campos como profesión, servicios que ofrece, experiencia laboral o relacionada, ciudad de ubicación, tarifa por sus servicios.

Búsqueda:



Función que permite realizar, dentro de la multitud de profesionales y servicios ofertados, aquellos que se ajustan a los requerimientos del cliente. Dentro de esta búsqueda se contempla poder realizarla por profesión, cuidad, servicio, entre otras

## Características de los Usuarios

Para este aplicativo se tienen definidos 2 perfiles o tipos de usuarios:

Usuario profesional: Es aquel que oferta sus servicios.

Usuario cliente: Aquel que está en la búsqueda de un profesional o un servicio específico.

## Restricciones

Portal Web Ingenio es una plataforma que busca acercar inicialmente a todos aquellos ingenieros (sin importar su especialidad) y a futuro las siguientes versiones se enfocarán en todas las profesiones tituladas, así como también registrar servicios de todos los campos como medicina, educación, asistencia y muchos otros campos de servicios contratables, y que los registrados desean ofrecer sus servicios.

## Suposiciones y Dependencias

Descripción desarrollo de la aplicación:

Para el desarrollo de Ingenio se utiliza React Native que es un framework multiplataforma de código abierto lanzado por Facebook para desarrollar aplicaciones tanto en iOS como Android las principales ventajas de desarrollar apps con este framework, y aquí te mostramos cuáles son:

1.- Adecuado para proyectos con restricciones de presupuesto: Trabajar en React Native es rápido y fácil, por lo que es un ahorro de tiempo y dinero.

2.- Una solución obtiene múltiples resultados: Permite a los desarrolladores ejecutar varias tareas complejas que usan código simple. Por ejemplo, la función “Live Reload” permite operar fácilmente los cambios de código en tiempo real mientras se carga toda la aplicación.

3.- Transformación más fácil de sitio web a app: Cualquier desarrollador puede comprender fácilmente una secuencia completa de códigos en React Native, lo que facilita el trabajo en equipo y el cambio de personal.

4.- Hace la interfaz de usuario fluida con acceso a API nativa: De esta manera, el desarrollador puede hacer que la interfaz de usuario parezca más JavaScript y mucho menos framework. Es un patrón utilizado para simplificar el proceso general y hacer que la UI funcione de manera más fluida para proporcionar resultados de alta calidad y una respuesta más rápida.

5. Permite el desarrollo a través de la base del código primario: La base de programación utiliza el mismo código tanto para iOS como para Android. Esencialmente, es como implementar la misma aplicación en todo tipo de SO móvil.

6.- Otorga soporte multiplataforma: React Native permite a las empresas utilizar el mismo código para el desarrollo de aplicaciones tanto iOS como Android. No se requieren varios lenguajes como Swift, Java o C ++. La única necesidad es contratar a un desarrollador que sepa de bibliotecas de interfaz de usuario nativas, API y desarrollo de apps híbridas.

7.- Su ejecución es similar a la de una aplicación nativa: Los bloques básicos de una aplicación React Native se compilan en sus raíces, que tienen plataformas nativas, todos los códigos son comparables con React Native para Android e iOS.

8.- Menor uso de memoria: React Native es bastante compatible con diversos plugins externos. Además, sus herramientas ofrecen un tiempo de ejecución mucho más fluido. Las plataformas React Native para Android e iOS utilizan un espacio de memoria mucho menor, ya que ni siquiera se requiere enlace cruzado y casi todos los códigos se usan durante el tiempo de ejecución.

9.- Tiene un futuro brillante: El portal sostiene que se seguirá usando por décadas dada la facilidad para usarlo. Los desarrolladores no tienen que aprender ningún idioma nuevo ni crear códigos complejos.

10.- Enfoque flexible para el desarrollo de proyectos: Como ya sabemos, este framework es multiplataforma, un solo código es suficiente para corregir errores y ofrece múltiples herramientas de desarrollo, por lo que es amigable para todo el que tenga una noción de desarrollo de aplicaciones.

* **Frontend**: Para el desarrollo de las interfaces de la página web se hará uso de React, que en términos sencillos es una librería para construir interfaces de usuario, basada en componentes. Facilita la creación de aplicaciones dinámicas, tiene diversas extensiones que facilita el trabajo, como el renderizado del lado del servidor y la reutilización de componentes, que con construirlos una vez pueden reutilizarse en toda la aplicación, lo cual reduce en gran medida el tiempo de desarrollo.
* **Backend:** Para el desarrollo de las interfaces en términos sencillos es una librería para construir interfaces de usuario, basada en componentes.
* **MongoDB**: Se manejará una base de datos No Relacional, basada en documentos, como lo es MongoDB. Una de las mayores ventajas es que no tiene restricciones de tipo de datos o valores a la hora de almacenar y que no hay una estructura obligatoria para la construcción de los documentos que almacenaran la información. Ya no hablamos de tablas como en una base de datos relacional, sino de colecciones que no es más que el grupo de documentos (que sin una estructura rígida ) almacenan la información.
* **Node JS**: Llenguaje de programación que se empleará es Javascript, Node JS nos facilita el entorno de ejecución para este lenguaje. Es liviano y eficiente, ya que traaja con entradas y salidas sin bloqueos. Con Node JS podemos trabajar con APIs asíncronas, es decir que no se debe esperar que una API devuelva datos, por medio de notificación de eventos, el servidor obtiene información de que API se acaba de usar y cómo proceder.
* **NPM**: Además será de gran ayuda su Sistema Gestor de Paquetes de node, en donde encontraremos innumerables librerías.
* **Express JS Framework:** framework estable que facilita el uso de Middlewares; código que se puede ejecutar antes de que un usuario acceda a la página web, con el fin de realizar una autentificación, por ejemplo, comprobar autorizaciones u obtener datos de cierto sitio que necesitamos para un request.

## Requerimientos Diferidos

Portal Web Ingenio a futuro las siguientes versiones se enfocarán en todas las profesiones tituladas así como también registrar servicios de todos los campos como medicina, educación, asistencia y muchos otros campos de servicios contratables, y que los registrados desean ofrecer sus servicios.

# Requerimientos específicos

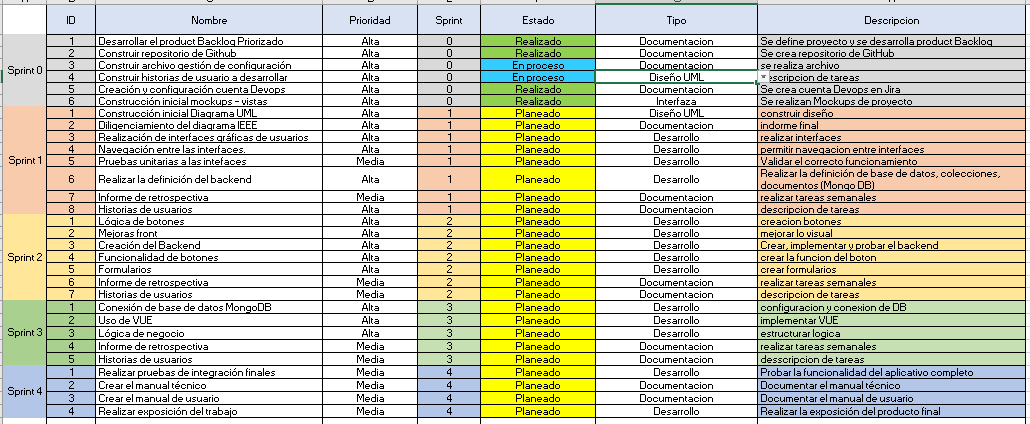
Dentro de las funciones que tendrá la aplicación se cuenta:

* Registrar los servicios a ofrecer.
* Consulta los servicios a ofrecer.

## Requerimientos

* Integrar ingenio a las redes profesionales.
* Conecta a empresas y empleados.
* Aprender las habilidades que necesitas para tener éxito en tu carrera

### Product Backlog



[Lista con todos los requerimientos iniciales del producto que se va a desarrollar que identifique las necesidades del producto para lograr su máxima utilidad. Asimismo, contiene la descripción de las tareas y subtareas que se van a realizar para la ejecución de cada requisito, mismas que se organizarán en función de sus prioridades. Además, la pila de producto también indica una estimación del tiempo en la que cada tarea se va a desarrollar y el valor que cada una le da al producto.]

### Ciclo de Sprints del proyecto

[Listar los sprints contemplados desde el proyecto para la generación de valor al cliente]

### Sprint Backlog

[Lista de elementos seleccionados previamente del Product Backlog para ser desarrollados en el día a día en los diferentes Sprints del proyecto. Tras crear esta lista, el equipo del proyecto tendrá que identificar las funcionalidades y priorizar las que se entregarán en el Sprint.]

### Historias de usuario (Tareas y Subtareas)

[Es una explicación general e informal de una función de software escrita desde la perspectiva del usuario final. Su propósito es articular cómo proporcionará una función de software valor al cliente. Asimismo, deben utilizan un lenguaje no técnico ya que supone la descripción de una función de software descrita por los usuarios finales].

### Mecánica de organización del grupo. (Reuniones, evidencias/artefactos)

## Modelo de Requerimientos

[El *modelo de requerimientos* se basa en dos modelos principales: el *modelo de casos de uso* y el *modelo del dominio*.]

### Modelo de Casos de Uso

[La funcionalidad de una aplicación Web se modela como un conjunto de *casos de uso*, que describen los requerimientos de la aplicación desde las perspectivas de los *actores* (personas y otros sistemas).

Una particularidad de los requerimientos de una aplicación Web es la funcionalidad de navegación, que permite al usuario navegar por el hipertexto y encontrar nodos. El enfoque de UWE es crear un único modelo de casos de uso, que utiliza el estereotipo <<navigation>> para denotar la diferencia entre casos de uso funcionales y casos de uso específicos de hipertexto.

OBSERVACIÓN: Si la cantidad de casos de uso en un mismo diagrama es grande (una cantidad adecuada podría ser 7 ± 2), sería conveniente aplicar el antiguo principio de "divide et impera". Es decir, dividir el diagrama de casos de uso en varias partes agrupando elementos que tengan algún tipo de coincidencia entre sí. El elemento de UML que se utiliza para estas situaciones es el *paquete*. Si todavía algún paquete tuviera demasiados casos de uso, entonces se podría subdividir ese paquete en varios paquetes, y dentro de estos estarían los casos de uso. Se debe asegurar que los paquetes sean cohesivos, es decir, que manejen información o funcionalidad relacionada. Luego, a partir del diagrama de paquetes se derivan los casos de uso; cada paquete implica un diagrama de casos de uso (o de paquetes).]



[Para el estereotipo <<navigation>>, en lugar de la etiqueta se puede usar el ícono ().]



[A continuación, se describen los casos de uso utilizando tablas y diagramas.]

**CU-01: "Nombre del Caso de Uso"**

[Se debe realizar una descripción de los escenarios del caso de uso.]

|  |  |  |
| --- | --- | --- |
| Iniciador | Actor que inicia el caso de uso. | |
| Otros actores | Otros actores, si existen. | |
| Precondiciones | Condiciones que deben cumplirse para que pueda realizarse el caso de uso. | |
| Flujo básico | | |
| **Actor** | | **Sistema** |
| 1. … | |  |
|  | | 2. … |
|  | | 3. … |
| 4. … | |  |
|  | |  |
|  | |  |
| Flujo alternativo 1 | Indicar otro camino por el cual se puede realizar el caso de uso, aparte del flujo básico | |
| Flujo alternativo 2 |  | |
| ... |  | |
| Flujo alternativo n |  | |
| Poscondiciones | Indicar los cambios que se producirán o los datos que deben quedar registrados. | |

[Cuando se ejecuta una instancia de un caso de uso, éste interactúa con instancias de actores y realiza una secuencia de acciones de acuerdo a lo que se describió en la especificación del caso de uso. Para mostrar la realización de un caso de uso se puede emplear: un *diagrama de clases* con todas las clases que participan en él; *diagramas de interacción* (*secuencia* y/o *comunicación*); o, como indica UWE, *diagramas de actividad*.

OBSERVACIÓN: UWE sugiere lo siguiente "Siguiendo el principio de utilizar UML siempre que sea posible para la especificación, se refinan los requerimientos con diagramas de actividad UML. Para cada caso de uso no trivial, se construye al menos un diagrama de actividad para el flujo principal de tareas a realizar para proporcionar la funcionalidad indicada por el caso de uso correspondiente. Opcionalmente, pueden utilizarse diagramas adicionales para excepciones y variantes."]

**CU-02: "Nombre del Caso de Uso"**

. . .

. . .

**CU-nn: "Nombre del Caso de Uso"**

# Descripción del diseño

[Para el desarrollo de toda esta sección 4 utilizar como bibliografía de soporte:

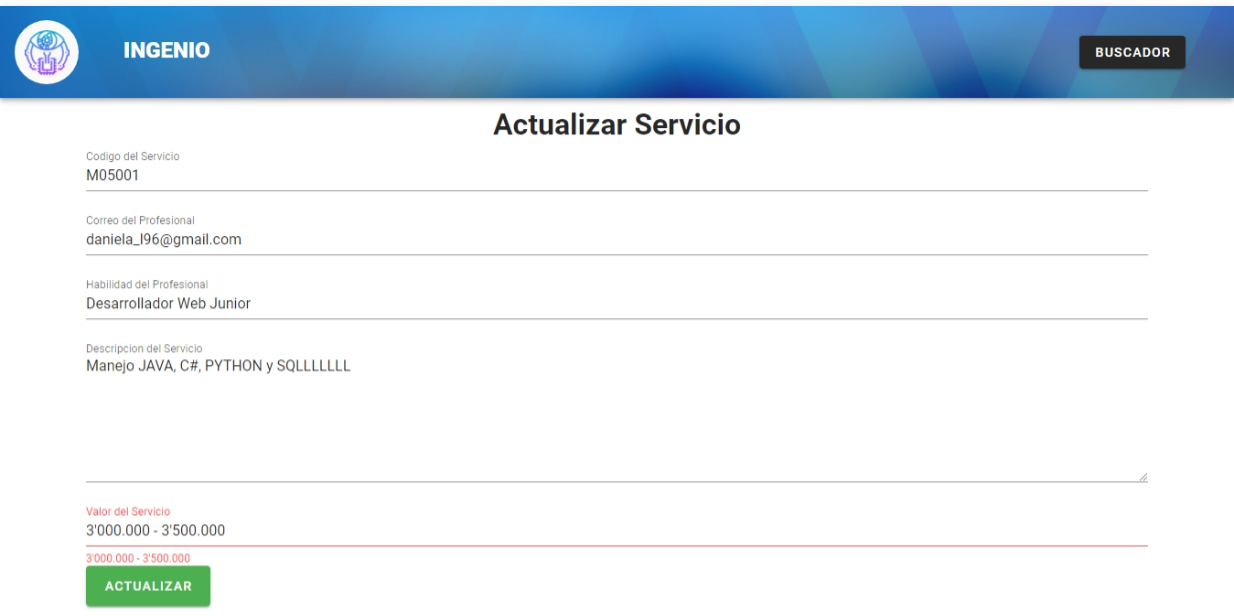
(1) Schwinger, W.; Koch, N. "Modeling Web Applications", Chapter 3 en: Kappel, G.; Pröll, B.; Reich, S.; Retschitzegger, W. (Editors) *Web Engineering. The Discipline of Systematic Development of Web Applications*, John Wiley & Sons Ltd., 2006.

(2) Koch, N.; Knapp, A.; Zhang, G.; Baumeister, H. "UML-Based Web Engineering. An Approach Based on Standards", Chapter 7 en: Rossi, G.; Pastor, O.; Schwabe, D.; Olsina, L. (Editors) *Web Engineering: Modelling and Implementing Web Applications*, Springer-Verlag London Limited, 2008.]

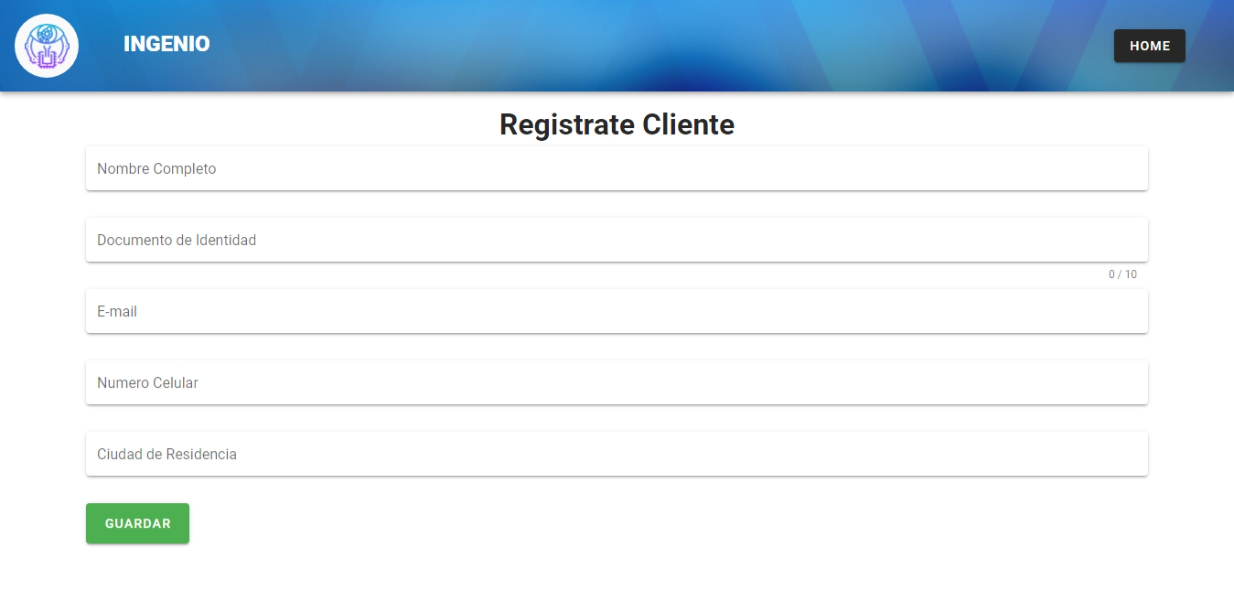
## Interfaz gráfica (Mockups)



Mockup inicial de la página Home del proyecto



Mockup de la vista “Actualizar Servicio”



Mockup de la vista “Registrar Cliente”

# Gestión de la configuración

Debe especificar los lenguajes de programación y el stack tecnológico que debe ser utilizado en la construcción de la solución.

especificar el uso y estructura de los repositorios de código

acceso y uso de la plataforma tecnológica en la nube con la que cuenta cada equipo

a configuración y uso de los ambientes de desarrollo y producción.

# Pruebas

En este apartado se describirán en detalle cada uno de los casos de pruebas que se hayan identificado como necesarios para verificar la funcionalidad completa del portal Web Ingenio que es una plataforma que busca acercar a todos aquellos ingenieros (sin importar su especialidad), que les gusta o quieren ofrecer sus servicios profesionales bien sea en modo freelance o con otro medio de contratación con aquellos potenciales clientes que están buscando a un profesional con ciertas características para asegurar el correcto despliegue de la aplicación.

## Descripción de pruebas unitarias

|  |  |  |
| --- | --- | --- |
| Registrar usuario en la plataforma | CP001 |  |
| ¿Prueba de despliegue? | Si/No |
| **Descripción:**  Necesito registrarme en la plataforma | | |
| **Prerrequisitos**  **<Enumerar los prerrequisitos para la prueba>**   * El usuario registro los campos de Nombre, Apellido, Usuario y Contraseña? | | |
| **Pasos:**   * Se verifica que el usuario es único en la plataforma * Se valida que la contraseña cumple con mínimo 8 caracteres y al menos uno numérico | | |

|  |  |  |
| --- | --- | --- |
| Registrar los servicios profesionales | CP002 |  |
| ¿Prueba de despliegue? | Si/No |
| **Descripción:**  Registrar los servicios profesionales que ofrezco | | |
| **Prerrequisitos**  **<Enumerar los prerrequisitos para la prueba>**   * El usuario está registrado en la plataforma? * Se verifica que el servicio ofrecido tenga un título. * Se verifica que el servicio cuente con una etiqueta. * Se verifica que el servicio cuente con un costo de realización * El servicio ofrecido se guardó en la base de datos? | | |
| **Pasos:**   * Se verifica que el usuario está registrado en la plataforma * Se valida que los datos de usuario son correctos * Se valida que los datos del servicio ofrecido son correctos y se guardaron en la BD | | |

|  |  |  |
| --- | --- | --- |
| Buscar los servicios profesionales | CP003 |  |
| ¿Prueba de despliegue? | Si/No |
| **Descripción:**  Quiero buscar los servicios ofrecidos por los ingenieros | | |
| **Prerrequisitos**  **<Enumerar los prerrequisitos para la prueba>**   * Se verifica que se muestre la información de los servicios ofrecidos * Se verifica que el buscador filtre la información correctamente | | |
| **Pasos:**   * Se verifica que los datos del servicio ofrecido son correctos y se guardaron en la BD | | |

## Descripción de pruebas de aceptación

|  |  |  |
| --- | --- | --- |
| Registrar usuario en la plataforma | CP001 |  |
| ¿Prueba de despliegue? | Si/No |
| **Descripción:**  Necesito registrarme en la plataforma | | |
| **Prerrequisitos**  **<Enumerar los prerrequisitos para la prueba>**   * El usuario registro los campos de Nombre, Apellido, Usuario y Contraseña? | | |
| **Pasos:**   * Se verifica que el usuario es único en la plataforma * Se valida que la contraseña cumple con mínimo 8 caracteres y al menos uno numérico | | |
| **Resultado esperado:**  Mensaje exitoso de que el usuario se encuentra registrado en la plataforma. | | |
| **Resultado obtenido:**  <Resultado obtenido de la ejecución del caso de prueba, incluir pantalla del mensaje obtenido> | | |

|  |  |  |
| --- | --- | --- |
| Registrar los servicios profesionales | CP002 |  |
| ¿Prueba de despliegue? | Si/No |
| **Descripción:**  Registrar los servicios profesionales que ofrezco | | |
| **Prerrequisitos**  **<Enumerar los prerrequisitos para la prueba>**   * El usuario está registrado en la plataforma? * Se verifica que el servicio ofrecido tenga un título. * Se verifica que el servicio cuente con una etiqueta. * Se verifica que el servicio cuente con un costo de realización * El servicio ofrecido se guardó en la base de datos? | | |
| **Pasos:**   * Se verifica que el usuario está registrado en la plataforma * Se valida que los datos de usuario son correctos * Se valida que los datos del servicio ofrecido son correctos y se guardaron en la BD | | |
| **Resultado esperado:**   * Mensaje exitoso de que el usuario se encuentra registrado en la plataforma. * Mensaje exitoso de que el servicio se guardó en la BD y se encuentra registrado en la plataforma. | | |
| **Resultado obtenido:**  <Resultado obtenido de la ejecución del caso de prueba, incluir pantalla del mensaje obtenido> | | |

|  |  |  |
| --- | --- | --- |
| Buscar los servicios profesionales | CP003 |  |
| ¿Prueba de despliegue? | Si/No |
| **Descripción:**  Quiero buscar los servicios ofrecidos por los ingenieros | | |
| **Prerrequisitos**  **<Enumerar los prerrequisitos para la prueba>**   * Se verifica que se muestre la información de los servicios ofrecidos * Se verifica que el buscador filtre la información correctamente | | |
| **Pasos:**   * Se verifica que los datos del servicio ofrecido son correctos y se guardaron en la BD | | |
| **Resultado esperado:**   * Mensaje exitoso del buscador y que se presente en pantalla la información actual registrada en la plataforma con los profesionales que cumplen con los criterios seleccionados en los filtros del buscador. | | |
| **Resultado obtenido:**  <Resultado obtenido de la ejecución del caso de prueba, incluir pantalla del mensaje obtenido> | | |

## ESTRATEGIA DE EJECUCIÓN DE PRUEBAS

.

La estrategia a seguir en la ejecución de las pruebas para identificar los posibles ciclos de pruebas se determinaron en los ciclos 2 y 3, y la aplicación de los 3 casos de pruebas que lo componen de tal forma que puedan relacionarse los casos de prueba con los ciclos donde son ejecutados.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Prueba ID/ Ciclo de ejecución** | **<Ciclo 0>** | **<Ciclo 1>** | **<Ciclo 2>** | **<Ciclo 3>** |
| **<CP001>** | NA | NA | X | X |
| **<CP002>** | NA | NA | X | X |
| **<CP003>** | NA | NA | X | X |

# Glosario

# Anexo(s)

[ES OPCIONAL]

[Se pueden agregar anexos, si se consideran necesarios para obtener mayor claridad en el contenido del documento.]