****

**Arquitectura de software**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre del proyecto: | Diagnóstico de plagas en los cultivos de la región. | | |
| Alias del proyecto: | Healthy Plants | | |
| Versión: | 1.0 | Fecha última  modificación: | 22/11/2016 |

**INTRODUCCIÓN**

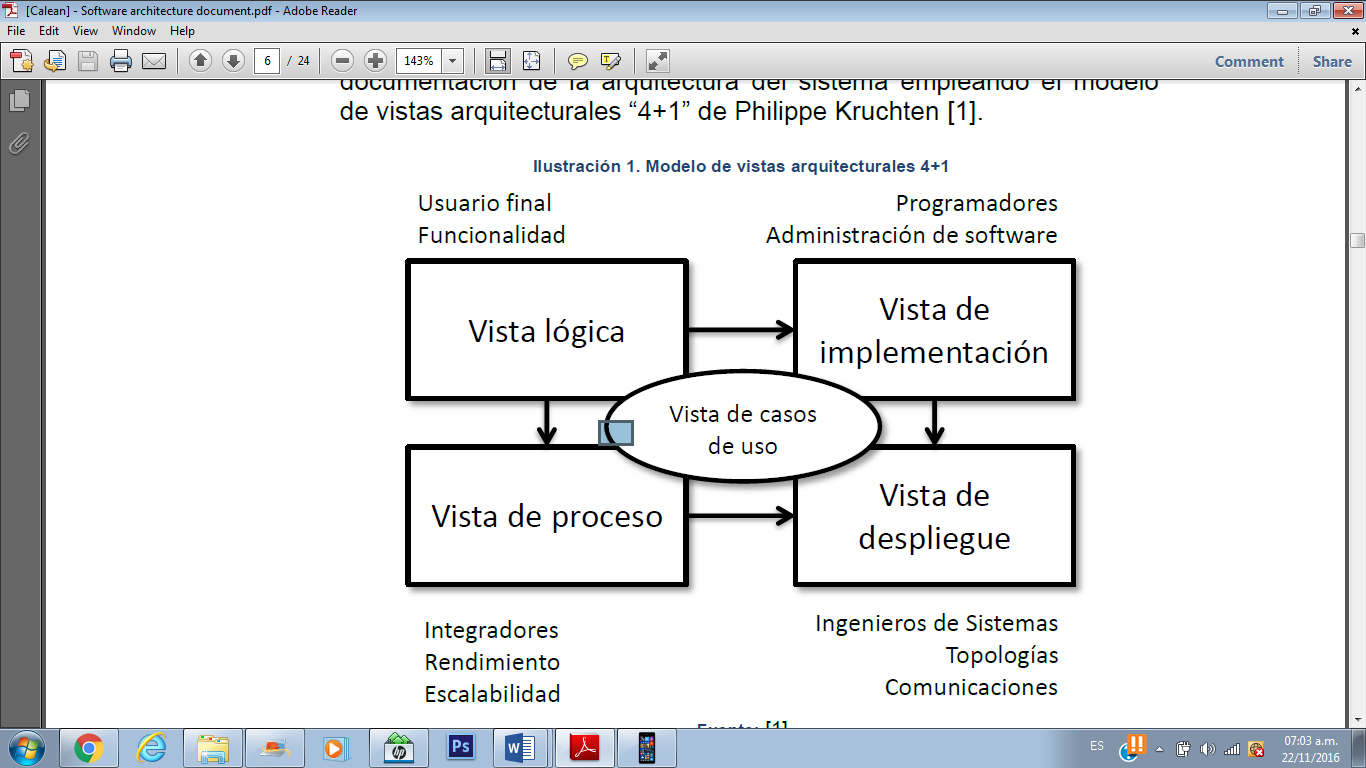
El presente documento pretende describir la arquitectura del presente proyecto Healthy Plants mediante la descripción de cada una de sus vistas, de manera que se pueda generar los conceptos generales y detallados del funcionamiento de la aplicación antes durante su desarrollo y posteriormente en su fase de implementación.

En proyecto se concibe como una aplicación web para móviles Android, es decir, se trata de un producto generado a partir de tecnologías web como HTML Y JavaScript que se ejecutará en un dispositivo móvil haciendo uso de librerías e intérpretes, esto con el fin de agilizar su desarrollo y concentrar la mayor carga de trabajo fuera de los terminales.

**PROPÓSITO**

El documento de arquitectura de software (SAD) especifica los actores, componentes, funciones y acciones del sistema empleando el modelo de vistas arquitecturales “4+1” de Philippe Kruchten los cuales son:

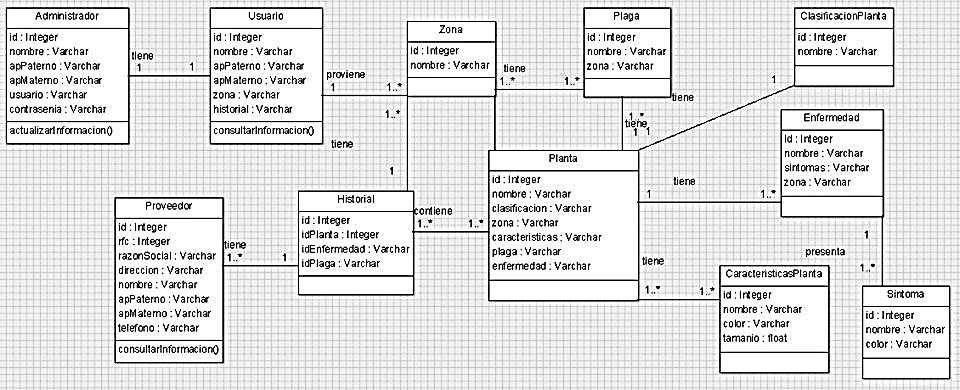
* Vista lógica: Enfocada a describir la estructura y funcionalidad del sistema.
* Vista despliegue: Describe en sistema desde la perspectiva del desarrollador.
* Vista de procesos: Describe la escalabilidad, distribución, rendimiento y concurrencia.
* Vista física: Está enfocada a los aspectos físicos y a la topología del software.
* Vista de escenarios: Define los casos de uso.



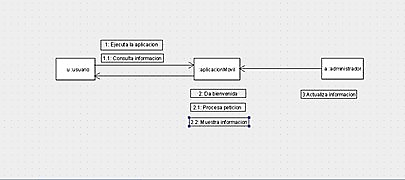
**VISTA LÓGICA**

En primera instancia se tiene el diagrama de clases el cual representa la estructura interna de la base de datos necesaria para el funcionamiento de la aplicación y está conformada por los siguientes elementos:

* **Administrador:** Representa a los usuarios administradores del sistema.
* **Proveedor:** Son aquellos usuarios que están registrados en el sistema para ofrecer productos o servicios.
* **Usuarios:** Es el usuario general, registrado solamente para hacer uso de la plataforma.
* **Historial:** Son las búsquedas realizadas por los usuarios.
* **Zona:** Datos de ubicación geográfica de las plantas, enfermedades o plagas.
* **Planta:** Son las plantas catalogadas dentro del sistema.
* **Plaga:** Son las afecciones de tipo animal.
* **Enfermedad**: Son las afecciones provocadas por virus, bacterias y hongos.
* **ClasificacionPlanta:** Agrupa y clasifica las plantas catalogadas.
* **CaracteristicasPlanta:** Detalla las características de las plantas catalogadas.
* **Sintoma:** Son las características que pueden presentar las plantas afectadas.



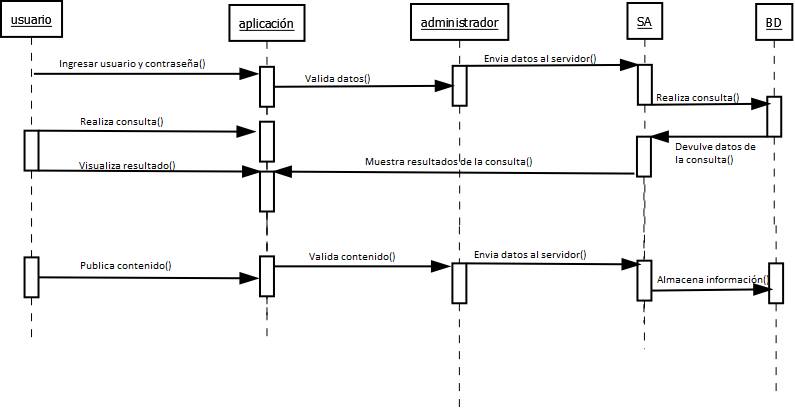
La aplicación seguirá un modelo cliente-servidor, es decir, esta se comunicará con un servidor remoto para consultar los datos almacenados en el mismo por medio de un servidor web y un gestor de base de datos MySQL, de esta manera se garantizará que la información sea fácil de actualizar, se sobreentiende que el usuario en ningún momento tendrá acceso el servidor y como consecuencia de ello a la base de datos, interactuará con la misma por medio de la aplicación.



En términos generales la secuencia para el uso de la aplicación es la siguiente:

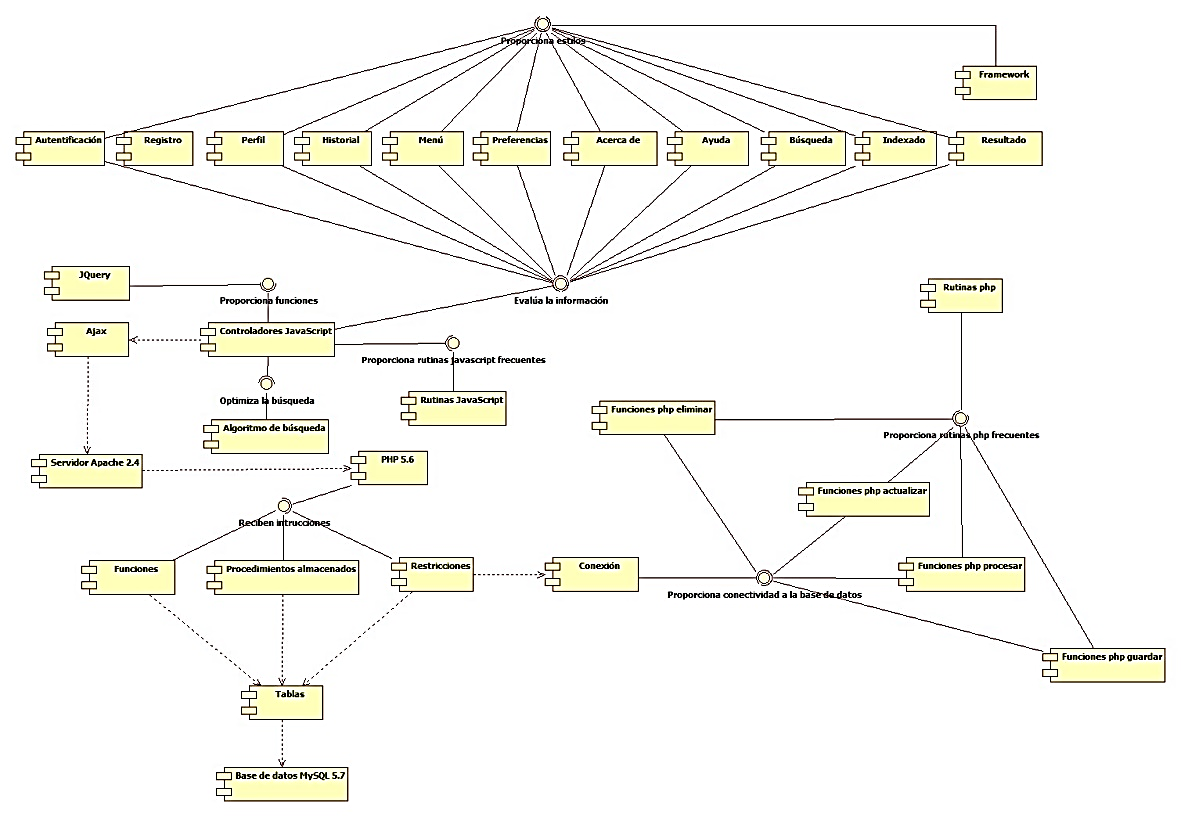
* El usuario ingresa su usuario y contraseña.
* La aplicación manda una solicitud al servidor.
* El servidor permite o rechaza la solicitud.
* Si el servidor permite la solicitud entonces generará una sesión para el usuario.
* El usuario puede hacer uso de los distintos módulos disponibles según el tipo de usuario (administrador, proveedor, usuario).

Debe tomarse en cuenta que durante la sesión se guardará información temporal que permita al usuario trabajar aun cuando esta comunicación presente una conectividad deficiente, también por razones de seguridad se considera el tiempo de inactividad, cerrando las sesiones activas en caso de que no se esté realizando ningún tipo de petición por parte de la aplicación.



**VISTA DE DESPLIEGUE**

En este diagrama de componentes no se contempla las partes específicas necesarias para la implementación y funcionamiento de la aplicación sino de manera general los componentes necesarios para su funcionamiento, entre estos destacan:

* **Framework:** Son los las extensiones, archivos css y javascript que facilitan el desarrollo de las interfaces de la aplicación y las acciones vinculadas a estas.
* **JQuery:** Es un conjunto de librerías o plugins javascript que agilizan el desarrollo al proporcionar rutinas frecuentes.
* **Rutinas frecuentes javascript y php:** Es código propio reutilizable por cualquier otro elemento compatible de la aplicación.
* **Ajax:** Es una técnica el desarrollo de aplicaciones web interactivas.
* **Apache 2.4:** Servicio web necesario para el acceso remoto.
* **PHP 5.6:** Lenguaje necesario para la interpretación de las instrucciones recibidas los terminales cliente.
* **MySQL:** Gestor de base de datos.

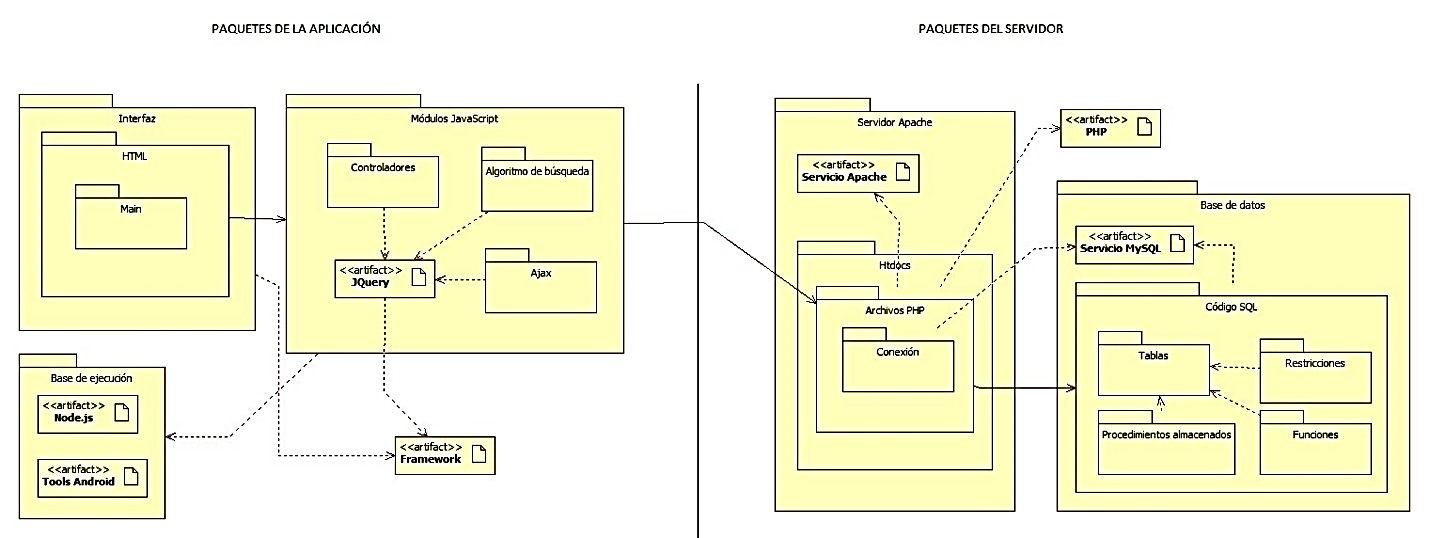
El funcionamiento del servicio se basa en tres conjuntos de paquetes principales, divididos a su vez en dos grandes conjuntos correspondientes a la aplicación y al servidor.

Lado cliente

* **Interfaz:** Permite al usuario el uso y la interación con el servicio.
* **Módulos JavaScript:** Contienen las rutinas y eventos relacionados con las acciones mostradas en la interfaz.
* **Base de ejecución:** Son las liberías y conunto de intérpretes necesarios para la ejecución de la aplicación bajo la plataforma Android.

Lado servidor

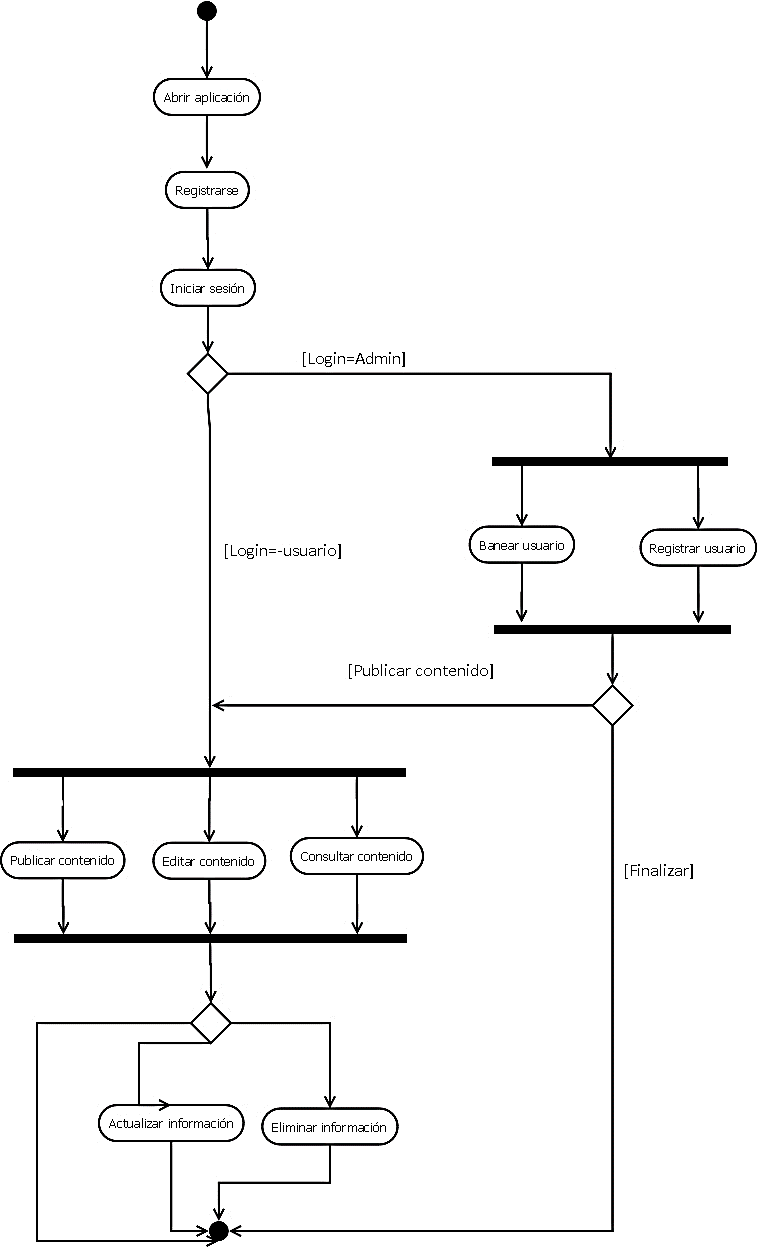
* **Servidor Apache:** Proporciona una vía de acceso remoto a la base de datos.
* **PHP:** Ayuda a interpretar las peticiones al servidor.
* **Base de datos MySQL:** Almacena la información necesaria y acciones relacionadas a los datos.



**VISTA DE PROCESOS**

Existen cuando menos dos tipos de usuarios principales, administradores o usuarios, de estos últimos pueden ser proveedores o simples usuarios, cuando un usuario se autentifica en el sistema el mismo validará las claves y determinará qué tipo de privilegios le corresponden cuales acciones podrá realizar.

En caso de ser un usuario podrá publicar, editar y consultar contenido (realizar búsquedas) en tanto que si se tratara de un administrador podrá adicionalmente eliminar o bloquear usuarios.

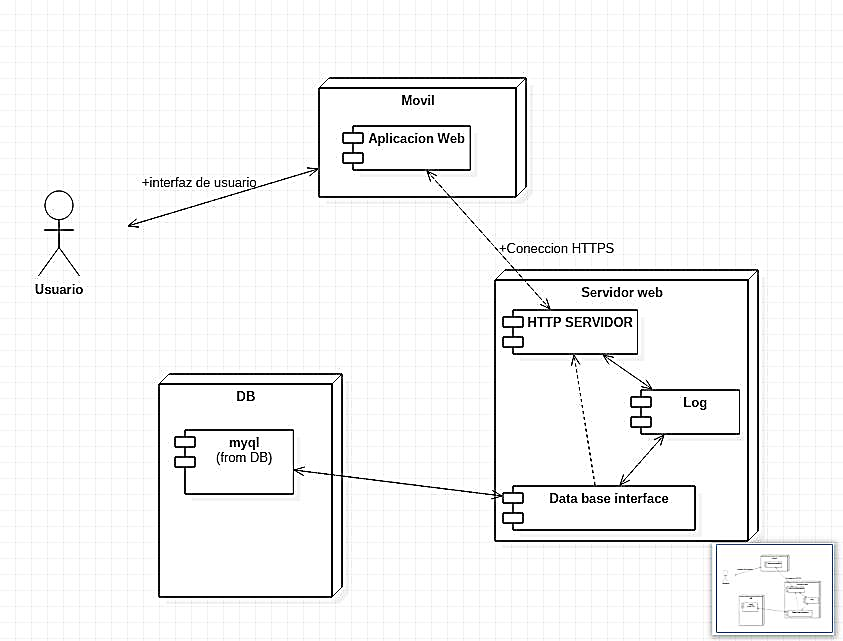


**VISTA FISICA**

De manera general la aplicación funcionará bajo plataformas Android haciendo uso de Node.js para su ejecución, se hará uso de tecnología web para su construcción, el dispositivo se conectará a una base de datos remota que almacenara y gestionará la información.

Se consideran los siguientes elementos para el funcionamiento del producto.

* **El usuario:** Que interactuará con la aplicación.
* **La aplicación:** Construida con tecnología web contendrá la mayor parte del algoritmo y será la principal responsable de analizar los datos.
* **Servidor web:** Proporcionará un medio de acceso remoto a la base de datos.
* **Base de datos:** Contendrá toda la información indispensable sobre los usuarios, plantas y afecciones indispensables.



**VISTA DE ESCENARIOS**

A continuación se describen los correspondientes casos de uso.

|  |  |  |
| --- | --- | --- |
| Caso de uso | Registrarse | CU1 |
| Descripción | Los usuarios pueden registrarse por su propia cuenta. | |
| Actores | Usuario, proveedor y administrador. | |
| Precondiciones | * El usuario debe tener instalada la aplicación. | |
| Flujo normal | 1. El usuario accede a la aplicación. 2. El usuario selecciona la opción Registro. 3. El usuario proporciona sus datos de registro. 4. La aplicación realiza la petición al servidor. 5. El servidor valida el registro. 6. El usuario queda dado de alta. | |
| Flujo alternativo | 1. El usuario accede a la aplicación. 2. El usuario selecciona la opción Registro. 3. El usuario proporciona sus datos de registro. 4. La aplicación realiza la petición al servidor. 5. El servidor niega el registro. | |
| Post-condiciones | * El usuario recibe una confirmación de su registro. | |
| Frecuencia | Media | |
| Importancia | Alta | |

|  |  |  |
| --- | --- | --- |
| Caso de uso | Iniciar sesión | CU2 |
| Descripción | Los usuarios registrados pueden acceder al sistema. | |
| Actores | Usuario, proveedor y administrador. | |
| Precondiciones | * El usuario debe contar con un registro previo. | |
| Flujo normal | 1. El usuario accede a la aplicación. 2. El usuario ingresa su usuario y contraseña. 3. La aplicación realiza una petición al servidor. 4. El servidor valida los datos. 5. El usuario accede al sistema. | |
| Flujo alternativo | 1. El usuario accede a la aplicación. 2. El usuario ingresa su usuario y contraseña. 3. La aplicación realiza una petición al servidor. 4. El servidor niega el acceso. | |
| Post-condiciones | * El usuario recibe una notificación. * El usuario accede al sistema. | |
| Frecuencia | Alta | |
| Importancia | Alta | |

|  |  |  |
| --- | --- | --- |
| Caso de uso | Consultar información | CU3 |
| Descripción | Los usuarios registrados pueden realizar búsquedas o realizar diagnósticos por medio de la aplicación. | |
| Actores | Usuario, proveedor y administrador. | |
| Precondiciones | * El usuario debe haber accedido al sistema por medio de la aplicación con un usuario y contraseña válidos. | |
| Flujo normal | 1. El usuario accede a la sección de búsqueda. 2. El usuario realiza una búsqueda. 3. La aplicación solicita una consulta al servidor. 4. El servidor devuelve los resultados de la consulta. 5. La aplicación indexa los resultados. 6. El usuario selecciona un resultado. 7. El resultado se muestra a detalle. | |
| Flujo alternativo | 1. El usuario accede a la sección de búsqueda. 2. El usuario realiza una búsqueda. 3. La aplicación solicita una consulta al servidor. 4. El servidor no devuelve resultados. 5. La aplicación indica que no se encontraron coincidencias. | |
| Post-condiciones | * El usuario recibe una notificación. * El usuario puede seleccionar uno de los resultados indexados. * Se genera una nueva entrada en el historial. | |
| Frecuencia | Alta | |
| Importancia | Alta | |

|  |  |  |
| --- | --- | --- |
| Caso de uso | Asignar privilegios | CU4 |
| Descripción | Los administradores pueden asignar privilegios o beneficios a los usuarios. | |
| Actores | Administrador. | |
| Precondiciones | * El usuario debe contar con un registro en la base de datos. * El usuario no debe estar dado de baja. | |
| Flujo normal | 1. El administrador accede al sistema. 2. El administrador busca al usuario. 3. El administrador solicita al sistema el cambio de privilegios. 4. El sistema valida los privilegios del usuario. 5. El sistema realiza el ajuste de privilegios. | |
| Flujo alternativo | 1. El administrador accede al sistema. 2. El administrador busca al usuario. 3. El administrador solicita al sistema el cambio de privilegios. 4. El sistema valida los privilegios del usuario. 5. El sistema determina que no se pueden cambiar los privilegios. | |
| Post-condiciones | * El administrador recibe una notificación. * El usuario adquiere los nuevos privilegios. | |
| Frecuencia | Media | |
| Importancia | Alta | |

|  |  |  |
| --- | --- | --- |
| Caso de uso | Bloquear Usuario | CU5 |
| Descripción | Los administradores pueden bloquear el acceso a los usuarios que no acaten las normas del servicio. | |
| Actores | Administrador. | |
| Precondiciones | * El usuario debe estar activo en el sistema. | |
| Flujo normal | 1. El administrador accede al sistema. 2. El administrador busca al usuario. 3. El administrador solicita el bloqueo del acceso del usuario al sistema. 4. El sistema valida el estado del usuario. 5. El sistema bloquea el acceso del usuario al sistema. | |
| Flujo alternativo | 1. El administrador accede al sistema. 2. El administrador busca al usuario. 3. El administrador solicita el bloqueo del acceso del usuario al sistema. 4. El sistema valida el estado del usuario. 5. El sistema determina que no se puede bloquear el acceso del usuario al sistema. | |
| Post-condiciones | * El administrador recibe una notificación. * El usuario queda sin permisos de acceso al sistema. | |
| Frecuencia | Media | |
| Importancia | Alta | |

|  |  |  |
| --- | --- | --- |
| Caso de uso | Dar de baja | CU6 |
| Descripción | Los administradores pueden dar de baja a los usuarios registrados. | |
| Actores | Usuario, proveedor y administrador. | |
| Precondiciones | * El usuario debe existir en la base de datos. | |
| Flujo normal | 1. El administrador accede al sistema. 2. El administrador busca al usuario. 3. El administrador solicita solicita la baja al sistema. 4. El sistema valida el registro del usuario. 5. El usuario queda dado de baja del sistema. | |
| Flujo alternativo | 1. El administrador accede al sistema. 2. El administrador busca al usuario. 3. El administrador solicita solicita la baja al sistema. 4. El sistema valida el registro del usuario. 5. El sistema determina que ese usuario ya está dado de baja. | |
| Post-condiciones | * El administrador recibe una notificación. * El usuario queda dado de baja del sistema. | |
| Frecuencia | Baja | |
| Importancia | Media | |

|  |  |  |
| --- | --- | --- |
| Caso de uso | Editar contenido | CU7 |
| Descripción | Los usuarios registrados pueden modificar sus publicaciones. | |
| Actores | Usuario, proveedor y administrador. | |
| Precondiciones | * El usuario debe haber accedido al sistema por medio de la aplicación con un usuario y contraseña válidos. | |
| Flujo normal | 1. El usuario accede a su perfil por medio de la aplicación. 2. El usuario selecciona el apartado de Contenido. 3. El usuario selecciona la opción Nuevo contenido. 4. El usuario edita el nuevo contenido. 5. La aplicación sincroniza el nuevo contenido con el servidor. 6. El servidor almacena el nuevo contenido. | |
| Flujo alternativo | 1. El usuario accede a su perfil por medio de la aplicación. 2. El usuario selecciona el apartado de Contenido. 3. El usuario selecciona la opción Nuevo contenido. 4. El usuario edita el nuevo contenido. 5. La aplicación sincroniza el nuevo contenido con el servidor. 6. El servidor rechaza la sincronización. | |
| Post-condiciones | * El usuario recibe un mensaje de error de sincronización. * El contenido modificado queda almacenado en el servidor. | |
| Frecuencia | Media | |
| Importancia | Alta | |

|  |  |  |
| --- | --- | --- |
| Caso de uso | Eliminar contenido | CU8 |
| Descripción | Los usuarios registrados pueden eliminar el contenido que hayan publicado. | |
| Actores | Usuario, proveedor y administrador. | |
| Precondiciones | * El usuario debe ser propietario de dicho contenido. * El usuario debe ser administrador para eliminar contenido de otros usuarios. | |
| Flujo normal | 1. El usuario accede al perfil del cual desea eliminar el contenido. 2. El usuario accede a la sección Contenido. 3. El usuario selecciona el contenido que desea eliminar. 4. El usuario solicita su eliminación al sistema. 5. El sistema valida los privilegios del usuario y el propietario del contenido. 6. El contenido es eliminado exitosamente. | |
| Flujo alternativo | 1. El usuario accede al perfil del cual desea eliminar el contenido. 2. El usuario accede a la sección Contenido. 3. El usuario selecciona el contenido que desea eliminar. 4. El usuario solicita su eliminación al sistema. 5. El sistema valida los privilegios del usuario y el propietario del contenido. 6. El sistema determina que el contenido no puede ser eliminado. | |
| Post-condiciones | * El usuario recibe una notificación. * El contenido es eliminado de la base de datos. | |
| Frecuencia | Baja | |
| Importancia | Baja | |

|  |  |  |
| --- | --- | --- |
| Caso de uso | Actualizar información | CU9 |
| Descripción | Los usuarios registrados pueden modificar su información de perfil y contacto. | |
| Actores | Usuario, proveedor y administrador. | |
| Precondiciones | * El usuario debe ser propietario de dicha información. * El usuario debe ser administrador para actualizar información de otros usuarios. | |
| Flujo normal | 1. El usuario accede al perfil del cual desea editar información. 2. El usuario selecciona la opción Editar. 3. El usuario edita la información. 4. El sistema valida los privilegios del usuario y el propietario del perfil. 5. La información es actualizada correctamente. | |
| Flujo alternativo | 1. El usuario accede al perfil del cual desea editar información. 2. El usuario selecciona la opción Editar. 3. El usuario edita la información. 4. El sistema valida los privilegios del usuario y el propietario del perfil. 5. El sistema determina que no se puede actualizar la información. | |
| Post-condiciones | * El usuario recibe una notificación. * La nueva información del usuario queda almacenada en la base de datos. | |
| Frecuencia | Media | |
| Importancia | Alta | |

