# Requerimientos funcionales

|  |  |  |
| --- | --- | --- |
| Requerimientos Funcionales | | |
| Número | Requerimiento | Descripción |
| RF1 | Crear ruta óptima. | El sistema permitirá buscar una ruta óptima entre los usuarios a visitar de manera que se minimizará la pérdida de tiempo en el recorrido. |
| RF2 | Envío de mensajes. | El sistema permitirá enviar mensajes entre enfermeros y pacientes. |
| RF3 | Pacientes con más de una visita diaria. | El sistema permitirá que se pueda visitar más de una vez al día a un paciente en caso de que este lo requiera. |
| RF4 | Registro de visita. | Se podrá dejar un registro en tiempo real en la plataforma indicando que el enfermero concurrió con dicho paciente. |
| RF5 | Registro en plataforma de enfermeros/pacientes. | Se podrá realizar el registro de médicos y/o pacientes para la utilización de la plataforma (solo administrador podrá modificar registro). |
| RF6 | Generar reporte. | Se podrá generar un reporte del enfermero indicando cada caso atendido del día. |
| RF7 | Biblioteca de información. | Los usuarios podrán acceder a una biblioteca en la aplicación con el fin de informarse de diferentes enfermedades. |
| RF8 | Modificación de ruta. | La ruta se podrá modificar en tiempo real |
| RF9 | Crear de ruta. | El sistema permitirá agregar una ruta de pacientes a los que hay que visitar (de 1 hora y 30 a 1 clic). |
| RF10 | Visualización de ruta. | El sistema permitirá visualizar la ruta optima creada. |
| RF11 | Agregar próximo día de visita. | El sistema permitirá ingresar la próxima visita del paciente. |
| RF12 | Visualización de visitas a pacientes. | El sistema permitirá visualizar los próximos pacientes a ser visitados. |
| RF13 | Modificación usuarios. | El sistema permitirá modificar la información de los usuarios del sistema. |
| RF14 | Eliminación de usuarios. | El sistema permitirá eliminar un usuario de sistema. |
| R15 | Ver enfermedad paciente | El sistema permitirá ver que enfermedades tiene el paciente. |
| RF16 | Ver complejidad paciente | El sistema permitirá visualizar la complejidad del paciente (baja-intermedia-alta). |
| RF17 | Complejidad del paciente | El sistema permitirá ingresar la complejidad del paciente (baja-intermedia-alta) |
| RF18 | Fotografiar heridas de pacientes. | El sistema permitirá sacar fotografías de las heridas de los pacientes y subirlas al sistema. |
| RF19 | Visualización de heridas de pacientes. | El sistema permitirá visualizar el historial de fotografías de heridas. |
| RF20 | Finalización de ruta. | El sistema permitirá finalizar la ruta realizada. |

# Requerimientos no funcionales

|  |  |  |
| --- | --- | --- |
| Requerimientos No Funcionales | | |
| Número | Requerimiento | Descripción |
| RNF1 | Eficiencia. | El sistema debe ser capaz de procesar la creación de ruta optima en un máximo de 10 segundos. |
| RNF2 | Eficiencia. | La creación de usuarios debe responder en un máximo de 5 segundos. |
| RNF3 | Eficiencia. | El sistema debe ser capaz de responder mínimo a 100 usuarios concurrentes usando el sistema. |
| RNF4 | Eficiencia. | Los cambios realizados en la base de datos se ven reflejadas a todos los usuarios del sistema en menos de 5 segundos. |
| RNF5 | Seguridad de datos. | Los datos de los usuarios solo podrán ser vistos/modificados/eliminados por el administrador del sistema. |
| RNF6 | Seguridad. | El ingreso al sistema está restringido con usuarios definidos y contraseñas. |
| RNF7 | Usabilidad. | El tiempo de aprendizaje del sistema debe ser menor a 5 horas. |
| RNF8 | Usabilidad. | El sistema debe contar con manuales de usuario. |
| RNF9 | Usabilidad. | El sistema despliega mensajes de error informativos. |
| RNF10 | Usabilidad. | La aplicación contará con un diseño “responsive” para garantizar la correcta visualización en dispositivos celulares, tabletas y computadores. |
| RNF11 | Usabilidad. | El sistema posee interfaces amigables. |
| RNF12 | Producto. | El sistema de desarrollará con Django y base de datos MySQL |

# Características de los requerimientos:

* **No ambiguo:** El texto debe ser claro, preciso y tener una única interpretación posible.
* **Conciso:** Debe redactarse en un lenguaje comprensible por los inversores en lugar de uno de tipo técnico y especializado, aunque aun así debe referenciar los aspectos importantes.
* **Consistente:** Ningún requisito debe entrar en conflicto con otro requisito diferente, ni con parte de otro. Asimismo, el lenguaje empleado entre los distintos requisitos debe ser consistente también.
* **Completo:** Los requisitos deben contener en sí mismos toda la información necesaria, y no remitir a otras fuentes externas que los expliquen con más detalle.
* **Alcanzable:** Un requisito debe ser un objetivo realista, posible de ser alcanzado con el dinero, el tiempo y los recursos disponibles.
* **Verificable:** Se debe poder verificar con absoluta certeza, si el requisito fue satisfecho o no. Esta verificación puede lograrse mediante inspección, análisis, demostración o testeo.

Los **requerimientos** se basan en las restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo, estándares, usabilidad, portabilidad, entre otros. 

Los **Requerimientos no funcionales** son los requerimientos que no se refieren directamente a las funciones específicas que entrega el sistema, sino a las propiedades emergentes de éste como la fiabilidad, la respuesta en el tiempo y la capacidad de almacenamiento.