  
 **UNIVERSIDAD AUTÓNOMA DE ENCARNACIÓN**

**FACULTAD DE CIENCIA, ARTE Y TECNOLOGÍA**

**LICENCIATURA EN ANALISIS DE SISTEMAS INFORMATICOS**

**PROYECTO DE INVESTIGACIÓN DE GRADO**

**PLATAFORMA VIRTUAL PARA EL REGISTRO, CONTROL Y GESTIÓN DE DATOS DE PACIENTES INFECTADOS CON DENGUE DESTINADO A LA SEPTIMA REGION SANITARIA DEL DEPARTAMENTO DE ITAPUA, REPUBLICA DEL PARAGUAY**

**LINEA DE INVESTIGACIÓN:  
  
ESTUDIANTE:**

**MARIO DAVID SEGOVIA TROCHE**

**TUTOR:**

**OSVALDO MICNIUK  
  
Encarnación – Paraguay**

**Mes – 2020**

INDICE

[1. INTRODUCCIÓN 3](#_Toc67737751)

[2. MARCO TEÓRICO 5](#_Toc67737752)

[2.1. Consecuencias sociales y económicas de la enfermedad del Dengue en el Paraguay 5](#_Toc67737753)

[2.2. Utilización de las Tecnologías de la Información y Comunicación en el control y la erradicación del Dengue. 6](#_Toc67737754)

[2.3. Desarrollo de una API REST utilizando una metodología de desarrollo ágil para una plataforma de registro de personas infectadas con el virus del Dengue. 7](#_Toc67737755)

[3. PREGUNTAS, HIPOTESIS Y OBJETIVOS 9](#_Toc67737756)

[3.1. Preguntas: 9](#_Toc67737757)

[3.2. Hipótesis: 9](#_Toc67737758)

[3.3 Objetivos 10](#_Toc67737759)

[4. METODOLOGÍA 11](#_Toc67737760)

[4.1. Resumen de las etapas de trabajo. Delineación de la estructura de la plataforma y del módulo API REST principal 11](#_Toc67737761)

[4.2. Descripción de las actividades a realizarse durante cada etapa 12](#_Toc67737762)

[5. RESULTADOS Y DISCUSIÓN 15](#_Toc67737763)

[6. CONCLUSIÓN 16](#_Toc67737764)

[7. RECOMENDACIONES 17](#_Toc67737765)

[8. REFERENCIAS BIBLIOGRÁFICAS. 18](#_Toc67737766)

[9. ANEXO 20](#_Toc67737767)

# 1. INTRODUCCIÓN

Este trabajo tiene por finalidad realizar una plataforma virtual para el registro, control y gestión de datos de pacientes infectados con Dengue, que sea de utilidad para los responsables de la región sanitaria del departamento de Itapúa, República del Paraguay, en la toma de decisiones y definir estrategias en la lucha contra esta enfermedad.

El Dengue en el Paraguay crea una gran carga económica y social en el sistema de salud, con epidemias que desde hace años generan una cantidad de fallecidos y enfermos cuya atención médica representan un alto costo (Arbo, 2019).

En este sentido, se buscan soluciones y estrategias para erradicar o disminuir el impacto de este flagelo utilizando diversas herramientas, y una de ellas es la tecnología. Existen iniciativas como la utilización de plataformas digitales y aplicaciones para móviles con el fin de involucrar a la comunidad a participar activamente en la detección y eliminación de criaderos del mosquito Aedes Aegypti, transmisor del virus del Dengue (Parra, 2017).

Otras aplicaciones utilizadas son modelos matemáticos computacionales para predecir el comportamiento y expansión de posibles epidemias mediante datos y estadísticas históricas (Mello Román et al, 2019). Así también tenemos portales de gestión de datos e información sobre la enfermedad para su mejor manejo y reducir el impacto de éste sobre nuestro país (Pane et al, 2018).

Actualmente hay estudios aplicando las últimas tecnologías, como la aplicación de redes neuronales para mostrar escenarios posibles que puedan presentarse de esta dolencia (Ughelli et. al, 2017), y otras como investigación en sistemas de información geográficos para localizar las zonas de más incidencia y registrar en mapas los casos reportados (Condor Camara et. al, 2018).

Así pues, el objetivo es desarrollar una plataforma de registro y gestión de personas infectadas con el virus del Dengue destinado para la Séptima Región Sanitaria del departamento de Itapúa.

Esto será realizado mediante la planificación de actividades y el flujo de trabajo estructurado mediante una metodología de desarrollo ágil (Navarro Cadavid et. al, 2013). El software se desarrolla en un entorno web a partir de módulos independientes que trabajan en forma conjunta ([Molina Ríos et. al,](http://repositori.uji.es/xmlui/browse?type=author&authority=00e9b65e-d5f5-4133-8488-98af581641f5) 2018).

El modulo API (Application Programming Interfaces) se establece como el centro de los procesos y almacenamiento de datos de todo el sistema. A partir del protocolo de comunicación REST (Representational State Transfer), la API ofrece una interfaz que permite el intercambio de datos con los otros módulos (Revuelta Arribas, 2020). Para el almacenamiento de los datos se utiliza un Sistema de Gestión de Bases de Datos (SGBD) (Marín, 2019).

El desarrollo de la API REST se establece en cuatro etapas o fases:

Primera etapa: análisis y definición de requisitos y requerimientos, como el alcance del proyecto, software de programación y desarrollo necesarios.

Segunda etapa: diseño de la API, creación de los casos de uso, interfaces y diagramas, establecimiento de estándares de programación.

Tercera etapa: Programación del software a partir del diseño realizado, codificación en el lenguaje elegido, respaldar los repositorios con el software de versionado.

Cuarta etapa: Testeo del software, corrección de errores y documentación.

# 2. MARCO TEÓRICO

## 2.1. Consecuencias sociales y económicas de la enfermedad del Dengue en el Paraguay

El dengue es una enfermedad causado por un virus, del género Flavivirus, de la familia Flaviviridae, del cual se conocen cuatro serotipos identificados como: DENV-1, DENV-2, DENV-3 y DENV-4. Se transmite a los humanos a través de la picadura del mosquito Aedes Aegypti, una especie que esta mayormente activa en horas de la mañana y antes de oscurecer, y se reproduce colocando su huevos en espacios y lugares donde se almacene o acumule agua, como floreros, tanques, piscinas y todo tipo de desechos de la actividad humana como neumáticos viejos, elementos de plásticos como bolsas, vasos, recipientes de toda clase. Los síntomas típicos de esta enfermedad suelen presentarse como fiebre alta, dolores de cabeza, musculares y articulares., Los cuadros más graves pueden generar sangrados internos y daños profundos en órganos, hasta llevar a la muerte del paciente. Se extiende en regiones de climas tropicales y subtropicales, como África, norte de Australia, México, Centroamérica y Sudamérica (Lage et. al, 2015).

En el mundo se calcula que más de la mitad de la población mundial, alrededor de 3.600 millones de habitantes residen en zonas donde son susceptibles de contraer la enfermedad del Dengue. Según la Organización Mundial de la Salud (OMS), alrededor de 34 millones de casos de Dengue clínico ocurren cada año, así como 2 millones de casos de Dengue hemorrágico y más de 20 mil muertes (Arbo, 2019).

El Paraguay ha sufrido varias epidemias de Dengue desde la llegada de esta enfermedad al país. Se registran epidemias en los años 1989, 1990, 2007 y 2013, y un crecimiento sostenido de casos reportados desde los años 2009 hasta 2015, registrándose más de 300.000 casos de los cuales más de 100 fueron pacientes fallecidos. Específicamente en el año 2013 se reportaron 149.429 casos, el registro de mayor cantidad de casos confirmados (Arbo, 2019). Según datos del boletín correspondiente al periodo comprendido entre las fechas del 29 de diciembre del año 2019 al 25 del mes de abril del año 2020 de la Dirección General de Vigilancia de la Salud, se reportan 40.250 casos de Dengue, y un total de 59 fallecidos.

Según estimaciones realizadas, cada paciente representa un gasto de servicios e insumos sanitarios en guaraníes de aproximadamente 5.700.000 o en dólares americanos de 1.054, tomando como promedio 4 días de internación, sin considerar otros gastos indirectos, como el costo de transporte, los acompañantes, alimentos, así como la pérdida de ingresos por la imposibilidad de asistir a su puesto de trabajo, elevando significativamente el costo económico a los involucrados (Arbo, 2019; Rodriguez-Castro et. al, 2019).

Cifras obtenidas en un estudio realizado sobre el impacto económico del Dengue en el Instituto de Previsión Social del Paraguay, en la epidemia ocurrida entre el mes de diciembre del año 2006 hasta el mes de julio del año 2007, de un total de 5960 casos, los costos por internación de pacientes, insumos, contratación de médicos, estudios laboratoriales y subsidios por reposo totalizaron un monto de más 12 mil millones de guaraníes o más de 2,5 millones de dólares americanos (Flores et. al, 2015).

Según resultados obtenidos en un trabajo de investigación en relación a los costos del Dengue en el Paraguay entre los años 2010 al 2013, se reportaron al Ministerio de Salud Pública y Bienestar Social (MSPBS) 272.997 casos probables y confirmados de Dengue. Así también un número estimativo de 2.162.538 casos de pacientes ambulatorios o que no requerían de hospitalización y 45.801 casos de enfermos hospitalizados. En cuanto a los costos, calculados en dólares americanos, en relación a los pacientes ambulatorios, en gastos médicos directos se estima una cantidad de 113.533.245 US$, en gastos médicos no directos la cantidad de 23.787.918 US$ y en gastos indirectos la cantidad de 134.077.356 US$, con un total global de 276.804.864 US$. En relación a pacientes hospitalizados, los gastos médicos directos ascienden a 13.740.300 US$, los gastos médicos no directos a 5.816.727 US$ y gastos indirectos a 7.648.767 US$, totalizando 23.450.112 US$. Contrastando estos datos con el presupuesto anual total destinado a la salud publica en el Paraguay, que asciende a aproximadamente 800 millones de dólares americanos, en el año 2010 los costos por la enfermedad del Dengue fue del 2,9% del presupuesto anual de salud, en el año 2011 subió a un 6,6%, al año 2012 un 2,14%, y en el año 2013 un 20,9%. Estos montos son altamente significantes, y repercuten negativamente en el presupuesto destinado al sistema sanitario del país (Cuellar et al, 2020).

Estos números suponen una elevada carga económica y social en el sistema de atención médica y en la población en general.

## 

## 2.2. Utilización de las Tecnologías de la Información y Comunicación en el control y la erradicación del Dengue.

De las distintas estrategias que se utilizan para eliminar o disminuir los casos de Dengue en la población, una de ellas son las Tecnologías de la Información y la Comunicación (TIC), así como las relacionadas con la informática y la computación.

El proyecto TopaDengue, llevado a cabo por estudiantes de la Universidad Católica de la Ciudad de Asunción, Parguay, con financiamiento del Consejo Nacional de Ciencia y Tecnología (CONACYT), tenía por finalidad reducir el alcance de las epidemias de Dengue mediante la eliminación y control de criaderos del vector transmisor, el mosquito Aedes Aegypti, a través de las TIC, utilizando una plataforma web, combinado con una aplicación móvil, llamada DengueChat, por medio de la cual, la comunidad tenía una activa participación en los reportes sobre criaderos. Se desarrolló entre los años 2018 y 2019, en la ciudad de Asunción, Paraguay, en la zona llamada Bañado Sur, comprendiendo los barrios San Ignacio, San Cayetano y Caacupemí. Los resultados obtenidos evidenciaron reducciones en los índices de infestación del mosquito en las zonas intervenidas con respecto a las zonas de control. En relación a la plataforma web DengueChat, es una iniciativa desarrollada por el Laboratorio de Aplicaciones Sociales de la Universidad de California en Berkeley. Esta plataforma, que es de libre acceso, combina juegos digitales, comunicación e interacción entre los participantes a través de chats o mensajería instantánea y redes sociales, a las que se pueden acceder por medio de teléfonos celulares, computadoras y otros aparatos digitales. Los usuarios de esta plataforma son motivados a identificar y eliminar criaderos de mosquitos, por los cuales son recompensados con distinciones y otros tipos de reconocimientos, y principalmente socializando con la mayor cantidad de personas involucradas, con el fin de concienciar sobre el problema de la enfermedad y que mediante la participación de la mayor cantidad posible de individuos de una comunidad, se puede erradicar este problema que genera el Dengue. Además de propiciar la participación ciudadana, permite recolectar datos sobre los índices de infestación del mosquito vector transmisor del Dengue, con la cual tener información de la situación en tiempo real, que puede ser utilizada por los responsables o autoridades para la toma de decisiones. Entre las ciudades donde se está utilizando esta plataforma con iniciativas locales podemos mencionar a la Ciudad de Managua, del país de Nicaragua, a la ciudad de Armenia, de Colombia y lugares donde se realizaron ensayos piloto podemos mencionar a la ciudad de Rio de Janeiro, Brasil y las ciudades de Cuernavaca y Tepalcingo, Mexico (Parra, 2019).

Otra solución desarrollada es la plataforma de Datos Abiertos y Alertas sobre Dengue, un software libre para la recolección, almacenamiento, análisis y visualización de datos, y a partir del procesamiento de grandes cantidades de datos, poder obtener información útil, como filtrar datos específicos y buscar patrones de interés para investigadores o generar alertas tempranas sobre posibles epidemias. Este proyecto también contó con la financiación del CONACYT y el desarrollo de este software dio como resultado un sistema integrado por un módulo de recolección y publicación de datos, un módulo de análisis y visualización de datos y un módulo de alertas tempranas. Todas las herramientas utilizadas en el desarrollo del software son open source o de uso libre, con el objetivo que pueda ser utilizado por los potenciales interesados sin costo alguno. Otras características a destacar son la flexibilidad para poder recolectar datos de distintas fuentes y tipos, y almacenarlas en una base de datos integrados y ofrecer un acceso de forma fácil para los posibles usuarios, y poder aplicar sobre ellos distintas metodologías de análisis de datos. El entorno de desarrollo o framework utilizado también ofrece la posibilidad de ser extensible y poder agregar más funcionalidades y mejoras (Pane et al, 2018).

También se estudian modelos matemáticos computacionales, estadística, redes neuronales artificiales, minería de datos para aplicar en datos recolectados, y generar proyecciones de posibles escenarios que se presentarían de la enfermedad. Pruebas realizadas presentaron resultados que serían de utilidad para detectar tempranamente casos de Dengue (Ughelli et. al, 2017; Mello Román et al, 2019).

Los Sistemas de Información Geográfica (SIG), se utilizan para obtener datos de referencias geográficas y ubicación en mapas, mediante estos se pueden delimitar las áreas o zonas donde se tienen brotes de personas infectadas, índices de infestación del mosquito transmisor o lugares propicios para criaderos de larvas del vector, y aplicar medidas de contención o protocolos establecidos de acción, como realizar pruebas para detectar casos posibles que no se reportaron, fumigación para eliminar los mosquitos, o realizar limpiezas de criaderos de larvas. Este proyecto se llevó a cabo entre el mes de setiembre del año 2017 hasta el mes de diciembre del año 2019 en las ciudades de Lima y de Iquitos – Loreto, del Perú. Por medio de encuestas y digitalización de fichas de notificación de cuadros de Dengue se obtenían los datos de los pacientes, con sus datos de georeferencia. A partir de estos datos se cargaban en la plataforma SIG, con el cual era posible visualizar en los mapas digitales la áreas de influencia de la enfermedad (Condor Camara et. al, 2018).

Otras aplicaciones y trabajos de investigación que utilizaron las TIC para su realización que podemos mencionar:

- Medisys, creado en Italia, un sistema de alertas mediante el análisis de de noticias y tweets relacionados al Dengue.

- DengueTrends, de la empresa Google, es una aplicación que genera estadísticas a partir de búsquedas de palabras claves relacionadas con la enfermedad del Dengue.

- Denfree, proyecto que tiene la finalidad de detectar las causas más importantes que ocasionan la transmisión, infección y epidemia de dengue, conjuntamente con el desarrollo de nuevas herramientas para la detección y diagnóstico de infecciones asintomáticas.

- IDAMS, iniciativa que busca el desarrollo de herramientas innovadoras y nuevas para ser utilizadas al control del Dengue en un entorno global

- DengueTools, herramienta para establecer una distribución espacio-temporal de enfermedades de transmisión por vectores para un control efectivo (Pane et al, 2015).

De esta forma, podemos encontrar una gran cantidad de proyectos, trabajos e investigaciones sobre la enfermedad del Dengue que implementan y utilizan herramientas TIC y otras tecnologías de la computación e informática para su realización y concreción de los objetivos.

## 

## 2.3. Desarrollo de una API REST utilizando una metodología de desarrollo ágil para una plataforma de registro de personas infectadas con el virus del Dengue.

En la revisión de antecedentes de trabajos de investigación y otros proyectos relacionados a la enfermedad del Dengue, se evidencia que en varios casos se basaron en la implementación de herramientas TIC y otras tecnologías relacionadas a la computación y la informática para lograr sus objetivos propuestos. De esta forma, se considera factible llevar a cabo la propuesta de una plataforma web de registro de personas infectadas con el virus del Dengue, utilizando una variedad de herramientas informáticas para su desarrollo, que deben establecerse de manera clara y precisa, y una planificación cuidadosa para obtener un resultado exitoso.

La plataforma web propuesta se define como un desarrollo de software, con una estructura modular que funcionan en conjunto como un sistema unitario. Por lo tanto, es necesario establecer la forma de trabajo y las actividades que se llevaran a cabo, y las herramientas tecnológicas a ser utilizadas.

Primeramente, antes de iniciar todo desarrollo de software, se debe especificar qué tipo de metodología se utilizará. Se entiende por metodologías de desarrollo de software a un conjunto de métodos, procesos y técnicas que se utilizan de guía para ejecutar un proyecto con el objetivo de crear un nuevo software que sea funcional y de utilidad para los usuarios finales ([Molina Ríos et. al,](http://repositori.uji.es/xmlui/browse?type=author&authority=00e9b65e-d5f5-4133-8488-98af581641f5) 2018).

Las metodologías de desarrollo de software se clasifican en metodologías tradicionales y metodologías ágiles. Las metodologías tradicionales se caracterizan por un enfoque rígido en la planificación y el proceso, las etapas de trabajo se deben establecer de forma precisa y los periodos de tiempo deben cumplirse sin retrasos, se respalda en una documentación minuciosa y exhaustiva, la comunicación entre los involucrados es mínima, ya que los encargados de cada parte del proyecto se dedican exclusivamente a su trabajo establecido y los flujos de actividades no posibilitan opciones de cambios o ajustes, los cuales si son necesarios deben realizarse a través de procesos burocráticos que afectan a todo el proyecto.(Navarro Cadavid et. al, 2013).

Las metodologías de desarrollo ágil permite un flujo de trabajo dinámico, con un enfoque hacia el producto y evitando una documentación excesiva. También está centrado en el trabajo colaborativo, con un intercambio constante de información entre los involucrados en el proyecto. El cliente o usuario final también tiene una participación activa en el proceso de desarrollo del producto, interactuando con los involucrados dando retroalimentación para introducir mejoras o corregir errores. La flexibilidad es otro punto que caracteriza a este tipo de metodología, permitiendo realizar cambios o introducir correcciones durante el proceso de trabajo sin afectar al proyecto en general. Este tipo de metodología permite crear productos en periodos cortos de tiempo. Como ejemplo de metodologías ágiles más utilizadas podemos citar Scrum y Extreme Programming (XP) (Navarro Cadavid et. al, 2013).

La metodología ágil Scrum ofrece un marco de trabajo con una variedad de métodos, artefactos y herramientas para desarrollar un software de manera efectiva y eficaz. Se caracteriza por equipos de trabajo auto-gestionados y multifuncionales, los integrantes se organizan de forma autónoma y colaboran activamente entre sí para llegar a los objetivos. Los avances del proyecto se realizan a partir de iteraciones y en cada iteración se van incrementando las funcionalidades del producto o se van agregando o corrigiendo características según los requerimientos del cliente. Scrum establece tres roles para el equipo de trabajo, el scrum master, el product owner y el equipo de desarrollo. El scrum master que tiene la función de líder del equipo de desarrollo y velar por el correcto desempeño del grupo, el product owner o dueño del producto, se encarga de controlar el correcto desarrollo del producto y gestionar las funcionalidades para obtener un resultado exitoso y por ultimo tenemos el equipo de desarrollo, que se encarga de realizar todas las labores requeridas para obtener el producto final (Navarro Cadavid et. al, 2013).

En cuanto a los elementos principales que componen la metodología Scrum, se destacan los eventos como el Sprint, definido como un periodo especifico de tiempo donde se realiza el trabajo de desarrollo del producto y debe dar como resultado un incremento a ser evaluado por los responsables. El Sprint a su vez se compone de otros elementos como el Sprint planning, en el cual se planifican todas las actividades a ser desarrolladas en el Sprint, el Daily Scrum, que son reuniones diarias de corta duración donde se discuten las tareas realizadas y a realizarse, así como las dificultades encontradas, la revisión del Sprint, que se realiza al final de un Sprint donde el product owner tiene participación activa, se realiza una revisión general de las tareas realizadas y no realizadas, los problemas que se presentaron y la verificación del producto resultante. Otro evento a destacar es la retrospectiva de Sprint, un análisis del trabajo del equipo desarrollador, que se hizo bien y que mal, y crear un plan de mejoras para el siguiente Sprint. También debemos mencionar los artefactos de Scrum, piezas claves en este marco de trabajo. El Product Backlog es una lista de funcionalidades o características que el producto final debe tener de acuerdo a las necesidades solicitadas por el cliente, y puede ir cambiando, incrementándose o modificándose de acuerdo a las necesidades del dueño, por lo tanto no es una lista fija, sino es dinámica y evoluciona a lo largo del desarrollo del proyecto. El Sprint backlog es una lista derivada del product backlog, y se establece de acuerdo a las prioridades que cada ítem representa para el dueño, y se desarrollan en el Sprint planificado finalizando en el incremento del producto. El monitoreo de progreso es un control de los avances que se realizan en el Sprint, consistente en la revisión de las tareas que ya se cumplieron y las tareas que están pendientes. Por ultimo tenemos los Incrementos, que son la sumatoria de todos los ítems terminados de la lista del Product Backlog y que van determinando el estado de avance del proyecto (Navarro Cadavid et. al, 2013).

El sistema se modela en un formato de funcionamiento dentro de un entorno web, alojado en un servidor que nos ofrece accesibilidad desde internet. Esto nos permite utilizar la plataforma y disponer de los datos sin restricciones de horarios, y desde cualquier dispositivo que tenga acceso a internet. De esta forma, se trabaja con el esquema cliente/servidor, donde los usuarios que necesiten utilizar la plataforma podrán ingresar al sistema por medio de cualquier navegador web, ya sea desde una computadora, una tablet, un teléfono inteligente u otro dispositivo electrónico con conectividad a internet, a través de una interfaz de páginas web con la cual podrán interactuar con el servidor que contenga el sistema. Asi también no se necesita de ninguna instalación de software específico ni de equipos electrónicos especiales para utilizar la plataforma por parte de los usuarios y el sistema alojado en un servidor permite un mayor control y mejor mantenimiento, y aplicar medidas de seguridad y protección de datos más confiables, y copias de respaldo en caso de problemas o inconvenientes con el servicio, que garantiza una rápida respuesta en caso de alguna contingencia y restablecer el funcionamiento normal. Otra ventaja considerable es la posibilidad de crear una arquitectura de software basada en bloques o módulos independientes, que pueden ser desarrollados de acuerdo a las necesidades o requerimientos de funcionalidades, y pueden agregarse, modificarse o eliminarse sin que el sistema completo se vea afectado en su funcionamiento ([Molina Ríos et. al,](http://repositori.uji.es/xmlui/browse?type=author&authority=00e9b65e-d5f5-4133-8488-98af581641f5) 2018).

El software a desarrollar se pretende estructurar con una API principal que servirá de núcleo y va a concentrar el procesamiento y almacenamiento seguro de los datos. Una API, cuyas siglas provienen de los términos del inglés Application Programming Interfaces, que en español se traduce como Interfaces de Programación de Aplicaciones, se entiende como un conjunto de protocolos y definiciones que se establecen para poder establecer una comunicación entre dos o más aplicaciones, permitiendo su interacción y el intercambio de datos. Esta característica es la que posibilita el desarrollo de distintas aplicaciones con determinadas funcionalidades de forma independiente, y una vez terminados, todos los módulos puedan ser acoplados y funcionar conjuntamente. Las APIs se clasifican en locales y remotas. Las APIs locales se caracterizan por utilizarse dentro de un solo equipo computacional, en la comunicación entre el hardware que compone la computadora o la comunicación entre el software. Como ejemplo podrían ser: sistemas operativos, librerías de software, el hardware del equipo. Las APIs remotas son las comunicaciones entre distintos tipos de hardware o software utilizando un protocolo común establecido, normalmente separados físicamente. Se consideran APIs remotas: sistemas de bases de datos, binarias, web. Las APIs web son actualmente las más conocidas y con una amplia aplicación en desarrollos de software, basando su estándar de comunicación común utilizado en internet, el Protocolo de Transferencia de Hipertexto, más conocido por las siglas HTTP, abreviatura del inglés Hipertext Transfer Protocol (Revuelta Arribas, 2020).

Existen dos especificaciones de APIs web más utilizadas actualmente, denominadas SOAP y REST. Las APIs basadas en SOAP, del inglés “Simple Object Acces Protocol,” que en español podríamos traducir como “Protocolo de Acceso a Objetos Simples”, se caracteriza por utilizar XML, del inglés “Extensible Markup language”, en español “Lenguaje de Marcado Extensible” como formato de trasferencia de datos, y utiliza herramientas de desarrollo muy específicas para su implementación. La especificación REST, del inglés “Representational State Tranfer”, que traducido al español sería “Transferencia de Estado Representacional”, se resalta que puede implementar como formato de trasferencia de datos JSON, del inglés “JavaScrip Object Notation”, y en español “Notación de objeto JavaScript”, un formato sencillo y liviano, que actualmente tiene una gran aceptación y es cada vez más utilizado como alternativa al XML. La especificación REST se basa en los métodos HTTP para la comunicación e intercambio de datos entre las aplicaciones, por lo tanto utiliza los mismos métodos como GET para obtener datos, POST para el envío de datos para su almacenamiento, PUT para la modificación de datos existentes y DELETE para eliminar datos, por lo tanto esta especificación es una de las más utilizadas en la actualidad por su facilidad en la implementación, flexibilidad en su uso, eficiencia en el intercambio de datos y la gran cantidad de herramientas de desarrollo disponibles para su utilización. (Revuelta Arribas, 2020).

Considerando que la plataforma propuesta es un software planteado para funcionar en un entorno web, es primordial establecer el lenguaje de programación adecuado para este proyecto. Entendemos por lenguaje de programación como un conjunto de instrucciones que se utilizan para realizar una serie de operaciones que una computadora pueda ejecutar y con ello realizar distintos tipos de tareas (Sala, 2003). Los lenguajes de programación fueron evolucionando a la par que las computadoras, adaptándose y diversificándose en distintos tipos y naturalezas, de acuerdo a sus campos de acción. De esta forma tenemos lenguajes de programación enfocados a software de diversas clases, como software para uso de empresas, aplicaciones de uso personal, software centrado en teléfonos inteligentes y resaltamos los lenguajes destinados al desarrollo web.

De los lenguajes de programación enfocados a la web, podemos mencionar como los más importantes ASP, de la empresa Microsoft, y PHP, una herramienta de código libre. El lenguaje PHP es una de las más utilizadas en la actualidad, desde su creación en el año 1994, fue creciendo en características y funcionalidades y gracias a un sólido respaldo de una gran comunidad de desarrolladores se ha consolidado como referencia en la elección de los programadores (Díaz & Banchoff Tzancoff, 2000).

Asi como el surgimiento y crecimiento de distintos tipos de lenguajes de programación, esto también derivo en la creación de complementos y utilidades para los mismos con el fin de facilitar su utilización e implementación. De esta forma aparecieron los denominados “frameworks” o “entornos de desarrollo”, también se pueden mencionar como “entornos de trabajo”. Los frameworks son un conjunto de herramientas y conjuntos de archivos que ofrecen determinadas funcionalidades y ofrecen al desarrollador una estructura o un esqueleto que le facilite la construcción de su proyecto. De esta forma un programador utiliza estos frameworks para desarrollar su software agregando los complementos, librerías, paquetes y otras opciones que necesita de una forma rápida y sencilla, reduciendo de forma significativa el tiempo de trabajo requerido para completar un proyecto. En relación al lenguaje PHP, existen varias opciones de frameworks basados en este lenguaje como base, y entre las más utilizadas podemos mencionar a symfony, codeIgniter, cakePHP y Laravel (Sierra, Acosta, Ariza & Salas, 2013).

Laravel es uno de los frameworks basado en PHP más utilizados en la actualidad, dispone de una gran cantidad de librerías, paquetes de archivos, utilidades y herramientas que facilitan el desarrollo del software y el trabajo de los programadores. Es un framework de código abierto, fue creado en el año 2011 y actualmente continúa su crecimiento incorporando nuevas características, mejoras de rendimiento y optimizando los procesos de creación de aplicaciones para alcanzar una dinámica de trabajo ágil y veloz. Un punto a destacar en Laravel es la implementación del patrón de diseño Modelo Vista Controlador o MVC, una arquitectura de software que separa la interfaz del usuario de la lógica del negocio y del manejo de datos, consiguiendo un código de programación más ordenado , limpio y comprensible, facilitando su reutilización, mantenimiento y modificación. El diseño MVC se basa en tres componentes que interactúan entre si para realizar los procesos del software, primeramente tenemos el Modelo, que representa la estructura de la información que maneja la aplicación y por lo tanto gestiona el acceso a esta información que se realizan por medio de consultas o modificaciones, por otra parte tenemos el Controlador, que se encarga de responder a los eventos que ocurren en el software, normalmente que provienen de un usuario, procesando los pedidos que se requieren y se comunica con los modelos para acceder a la información que se precisa, básicamente actuando como un intermediario entre las vistas y los modelos, por ultimo tenemos la Vista, que se encarga de mostrar al usuario de una forma ordenada y de fácil comprensión toda la información que son procesados por el controlador y que son proveídos por los modelos. En una forma sencilla de descripción podemos decir que un usuario visualiza la información e interactúa con el sistema por medio de las Vistas, a su vez éstas trasladan los pedidos del usuario a los Controladores que procesan estos pedidos y se conectan con los Modelos para acceder a la información y realizar las manipulaciones solicitadas, ya sean crear nueva información, consultar, modificar o eliminar datos. Por estas características el framework Laravel es una de las opciones más sólidas y viables para realizar a cabo un proyecto de desarrollo web (Ovando Ortega, 2019).

En todo sistema informático que se pretenda desarrollar un aspecto clave y primordial es la forma en que se va a tratar y manipular los datos que el sistema va a generar y almacenar. En respuesta a esta necesidad se crearon los Sistemas de Gestión de Bases de Datos (SGBD), que es un conjunto de aplicaciones que nos posibilitan la creación, gestión y administración de bases de datos, además de ofrecer las estructuras necesarias para el resguardo y la búsqueda de información del modo más eficiente posible. Los SGDB se clasifican en no relacionales (NoSQL) y relacionales (SQL). Los SGBD no relacionales (NoSQL) se identifican por no utilizar estructuras fijas de datos, como tablas y funcionan en entornos distribuidos, una desventaja de estos tipos de SGBD seria no garantizar una integridad completa de los datos. Como ejemplo podemos mecionar a Mongo DB, Redis, Cassandra. Los SGBD relacionales (SQL) son los más utilizados en la actualidad, ya que sus comienzos datan del año 1970, teniendo una larga trayectoria de evolución hasta nuestros días. Su funcionamiento se basa en estructurar los datos en tablas y estableciendo vínculos entre ellos y establece normas rígidas para asegurar la integridad y seguridad de los datos. Entre los SGBD relacionales más conocidos podemos mencionar SQL Server, de la empresa Microsoft y ORACLE, que son de pago y Postgresql, Mysql, MariaDB son software libre que pueden implementarse en un proyecto sin la necesidad de pago por licencias. De las opciones de SGBD relacionales de licencia libre, PostgreSQL es de los más utilizados y más completos, ofreciendo una solución robusta y potente para distintos tipos de desarrollo de software (Marín, 2019).

De esta forma, es factible desarrollar una plataforma para el registro de pacientes con dengue, utilizando una metodología de desarrollo ágil, en un entorno web que permita su acceso desde internet, en una estructura modular que permita su desarrollo en bloques independientes que una vez culminados puedan trabajar en conjunto, en la cual uno de los módulos sirva como núcleo de todo el sistema para el procesamiento y almacenamiento de los datos, basando su funcionamiento en una API REST que se encargaría de toda la lógica para la manipulación de los datos y su almacenamiento en un SGBD y ofrecer una interfaz de comunicación e intercambio de datos con los demás módulos de acuerdo a sus funcionalidades y requerimientos.

# 3. PREGUNTAS, HIPOTESIS Y OBJETIVOS

## 3.1. Preguntas:

- ¿Por qué desarrollar una plataforma para el registro de personas infectadas con Dengue destinado a la Séptima Región Sanitaria del departamento de Itapúa, república del Paraguay?

- ¿Qué metodología de desarrollo de software presenta las características necesarias para el sistema propuesto?

- ¿Qué sistema de software es posible desarrollar para cumplir con los requerimientos de una API REST como núcleo principal de la plataforma?

- ¿Qué software de programación presenta las funcionalidades requeridas para la generación de la lógica de funcionamiento, el procesamiento y almacenamiento de datos de la API REST?

## 3.2. Hipótesis:

La plataforma de registro de personas infectadas con Dengue permitirá a la Séptima Región Sanitaria del departamento de Itapúa tener acceso a una base de datos constantemente actualizada de los casos de la enfermedad registrados de la zona geográfica a la que pertenece. Además, contendrá un registro de todos los establecimientos de salud dependientes, así como de laboratorios de análisis clínicos, que serán los que registren los casos confirmados. A partir de los datos acumulados en el sistema se pueden aplicar varios procesos de selección, filtrado y algoritmos que puedan proveer información útil. Esta información podrá ser utilizada para crear estrategias de lucha contra la enfermedad, establecer un mejor manejo de recursos y prevenir las consecuencias que la dolencia genera en la población.

La utilización de una metodología de desarrollo de software ágil nos permite establecer una dinámica de trabajo más rápida, y se adapta mejor a los requerimientos contemplados en el proyecto. Esta metodología nos permite un mejor manejo de las etapas necesarias para las actividades previstas y la administración del tiempo en periodos más reducidos. Se reduce de manera significativa la documentación al menor nivel posible, evitando con esto la burocracia y la pérdida de tiempo. Se hace énfasis en el trabajo en equipo y colaborativo entre los involucrados, con una comunicación constante entre los miembros. El flujo de actividades es flexible e iterativo, se realizan evaluaciones constantes de los avances, los cuales ayudan en la corrección de errores y la toma de decisiones para definir nuevos objetivos o metas.

Para el desarrollo de la plataforma de registro de personas contagiadas de dengue, el sistema se orienta en un entorno web, instalado en un servidor conectado a internet. La ventaja principal de esa disposición es la capacidad de tener acceso al sistema desde cualquier dispositivo electrónico conectado a la web, a toda hora y desde cualquier lugar. El. El.

Para desarrollar la API REST principal se utilizará software de código abierto, cuya principal ventaja es la gratuidad de uso y posee una comunidad de desarrolladores que constantemente van actualizando, implementando mejoras de rendimiento y agregando nuevas funcionalidades. Se tiene una gran cantidad de software de desarrollo disponible y para esta plataforma la opción más valida sería el entorno de desarrollo LARAVEL con el lenguaje de programación PHP, para modelar la API REST con los procedimientos para el tratamiento y procesamiento de los datos y establecer la interfaz de comunicación para interactuar con los otros módulos. En relación al almacenamiento de la información se utiliza un sistema de gestión de bases de datos PostgreSQL, un software consolidado y con herramientas sólidas para el manejo óptimo y de forma segura de los datos.

## 3.3 Objetivos

- Desarrollar una plataforma de registro de personas infectadas con el virus del Dengue destinado a la Séptima Región Sanitaria del Departamento de Itapúa, República del Paraguay

- Seleccionar una metodología de desarrollo de software de acuerdo a las necesidades del sistema propuesto.

- Desarrollar una API REST para el sistema informático de la plataforma de registros de casos de personas infectadas con Dengue.

- Probar el funcionamiento de la API REST. Realizar

# 4. METODOLOGÍA

## 4.1. Resumen de las etapas de trabajo. Delineación de la estructura de la plataforma y del módulo API REST principal

Para el desarrollo de la plataforma, se realizará mediante una metodología de desarrollo ágil, dividiendo en cuatro etapas para trabajar el proyecto los cuales serían:

- 1ª Etapa: Definición y análisis de los requerimientos y necesidades del proyecto.

- 2ª Etapa: Diseño integral

- 3ª Etapa: Desarrollo del diseño

- 4ª Etapa: Pruebas de funcionamiento, corrección de errores y documentación

La estructura principal del proyecto se establece a partir de módulos a ser desarrollados de manera independiente y, una vez desarrollados, funcionarán en conjunto mediante interfaces de comunicación a través de las cuales podrán intercambiar y transferir datos e información.

Cada módulo estará a cargo de un responsable, que realizará todas las etapas del proyecto hasta su culminación, además de tener una comunicación constante con sus pares para coordinar los puntos en común a ser considerados para la funcionalidad total del sistema propuesto.

Modulo API Principal

Este módulo contendrá la programación que servirá de núcleo para interactuar con los demás módulos y ofrecer una interfaz de comunicación para el intercambio de datos. Se encarga del procesamiento y el almacenamiento de los datos, y gestionar las solicitudes de los demás módulos sobre éstos.

La estructura de interfaz de programación de aplicaciones o generalmente denominado por las siglas API, permite crear un conjunto de comandos, funciones y protocolos informáticos que facilitan la creación de software a los desarrolladores.

Módulo de Base de datos

Este módulo estará encargado de la estructura y el almacenamiento seguro de los datos a ser recolectados en el sistema, además de permitir un acceso a los datos para consultas o modificaciones.

Se debe establecer un software apropiado que nos permita realizar todos los procesos requeridos, existen varias opciones de gestores de bases de datos que nos ofrecen las funcionalidades esperadas. Se va a optar por un software libre, que presenta la ventaja de no generar un costo monetario para su implementación.

## 4.2. Descripción de las actividades a realizarse durante cada etapa

4.2.1. 1ª Etapa: Definición y análisis de los requerimientos y necesidades del proyecto.

Alcance del proyecto. Modulo API REST

Funciones de la API principal

- Creación, lectura, modificación y eliminación de registros de Usuarios del sistema. Se registran los datos personales, como nombres y apellidos, fecha de nacimiento, número de documento, dirección de vivienda, números telefónicos, correo electrónico.

Los usuarios tendrán la posibilidad de utilizar la plataforma, se registrarán y se le asignará un rol dentro del sistema, que limitará las acciones y las interacciones que podrá realizar con la plataforma. Dependiendo del tipo de rol asignado podrán crear registros, solicitar reportes, acceder a la base de datos.

- Creación, lectura, modificación y eliminación de registros de personas infectadas con Dengue. Se registran los datos personales, como nombres y apellidos, fecha de nacimiento, numero de documento, dirección de vivienda, números telefónicos, correo electrónico, datos de localización geográfica de la vivienda.

- Registro de auditoría: todos los movimientos de datos se registrarán para control y verificación de los cambios y quienes son los responsables, con la fecha y hora de cada modificación.

- Interfaces de comunicación: se establecerán los procedimientos para la interacción de la API principal con los demás módulos. Mediante el protocolo REST se podrá crear peticiones para crear, modificar o eliminar registros y responder a pedidos de rangos de registros específicos para reportes.

Definición del Software de desarrollo y herramientas necesarios para el proyecto

PHP: lenguaje de programación de código abierto con enfoque al entorno web

Laravel: es un entorno de desarrollo para crear aplicaciones y servicios web, utilizando el lenguaje PHP

PostgreSQL: Sistema de gestión de base de datos para la estructuración y administración de almacenamiento de los datos

Git: software para el control de versiones para el registro de las modificaciones y el avance de un proyecto.

Github: plataforma web que trabaja conjuntamente con Git para tener un control de versiones con repositorios en línea.

4.2.2. 2ª Etapa: Diseño integral de la API

Casos de uso y diagramas

Los casos de uso describen paso a paso las interacciones de la API con los módulos externos para cada proceso. Los diagramas son una representación gráfica de estos casos de uso.

Se utilizan otros diagramas para definir estructuras de las bases de datos, establecer el flujo de datos, de acuerdo a las necesidades del proyecto.

Diseño de interfaces

Las interfaces se establecen para definir la forma que la API puede comunicarse con agentes externos al sistema, en este caso no tendrá interacción directa con usuarios, solo con otros módulos a través de peticiones y solicitudes. Cada responsable de los otros módulos debe establecer sus requisitos de conexión y coordinar con el encargado de la API la correcta configuración.

Estándares de programación

Son convenciones en la nomenclatura y la escritura del código de programación. Son reglas establecidas para facilitar la lectura y lograr un texto del lenguaje más ordenado y homogéneo. Permite a los programadores entender de forma más rápida el funcionamiento del software realizado

4.2.3. 3ª Etapa: Desarrollo del diseño

Se inicia la generación de código de acuerdo al diseño previamente realizado. Se trabaja principalmente con el entorno de desarrollo seleccionado para realizar los procesos necesarios para el tratamiento de los datos.

Se configura la base de datos para estructurar el almacenamiento. Por medio del gestor de versiones se van creando puntos de control y registrando el avance del código. Se debe comentar de forma detallada cada bloque de código realizado para establecer una referencia de qué función cumple dentro del software.

4.2.4. 4ª Etapa: Pruebas de funcionamiento, corrección de errores y documentación

Una vez culminada la codificación, se procede a la verificación de la operatividad del software. Se realiza el testeo de todas funciones, la detección y corrección de fallas. Se contempla si es posible la introducción de mejoras o características nuevas que favorezcan el rendimiento general del sistema.

El testeo del software se verifica en formularios en los cuales se detallan los procesos realizados (*Ilustración 1*).

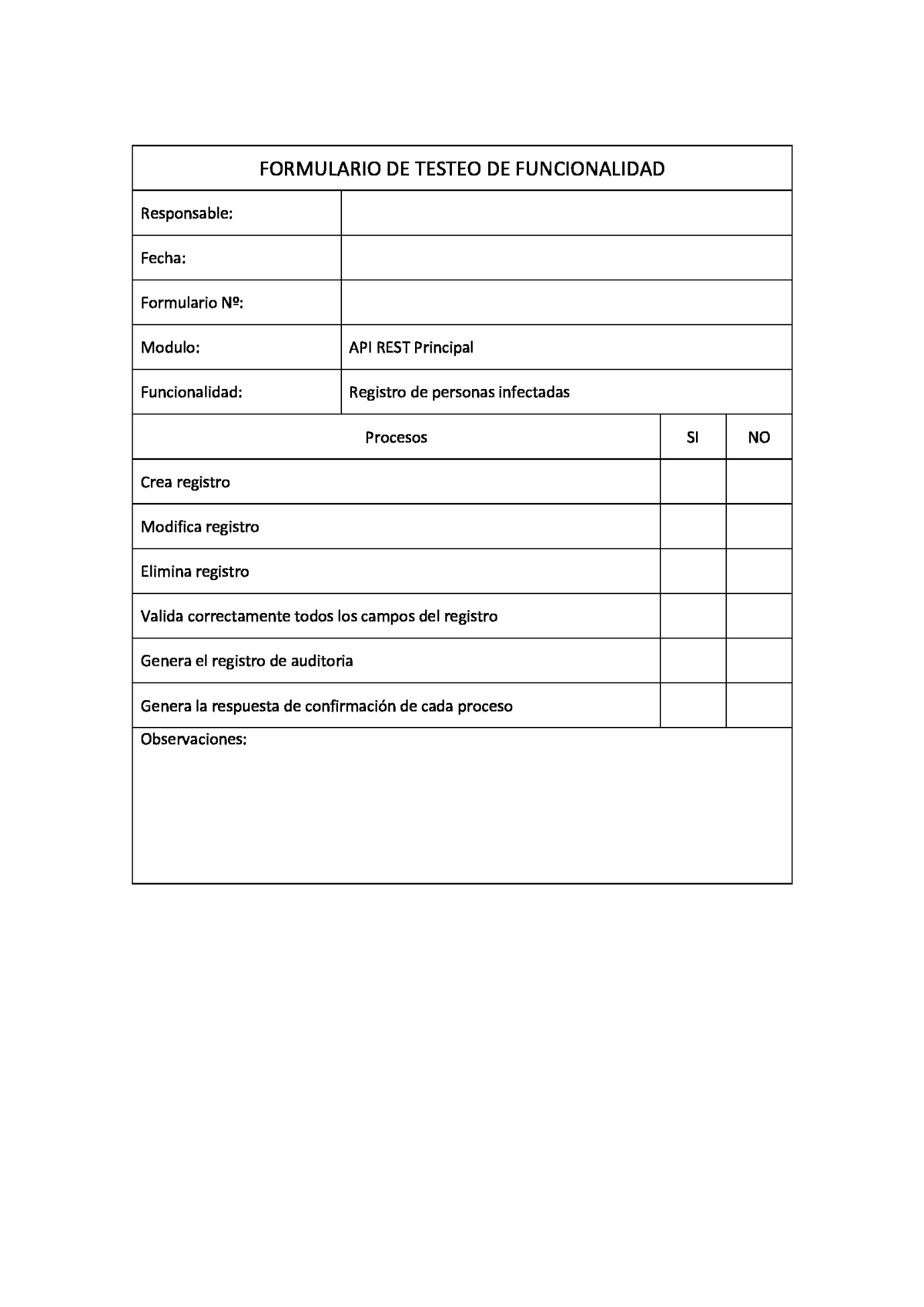


Ilustración 1. Formulario para testeo de funcionalidades de un software.

Finalizado estos procesos, se realiza la documentación completa de todo el trabajo realizado, así como los manuales de configuración y del usuario.

# 5. RESULTADOS Y DISCUSIÓN

# 6. cONCLUSIÓN

# 7. RECOMENDACIONES

# 8. REFERENCIAS BIBLIOGRÁFICAS.

Arbo, Antonio (2019). *Dengue: pesada carga para la salud pública del Paraguay*. http://dx.doi.org/10.18004/imt/20191411-2

Parra, Cristhian (2017). *TopaDengue.PY: tecnologías de la información y la comunicación para la promoción y soporte de iniciativas comunitarias contra arbovirus*. https://cicco.on.worldcat.org/oclc/1147979481

Mello Román, Jorge D., Mello-Román, Julio C., Gómez Guerrero, Santiago, & García Torres, Miguel (2019). *Predictive Models for the Medical Diagnosis of Dengue: A Case Study in Paraguay*. https://doi.org/10.1155/2019/7307803

Pane, Juan, Paciello, Julio, Lisnichuk, Yohanna, Martínez, Héctor, Valdez, Santiago & Durañona, Nelson (2018). *DATOS ABIERTOS Y ALERTAS SOBRE DENGUE*. https://publicaciones.fctunca.edu.py/jspui/handle/123456789/77

Ughelli, V., Lisnichuk, Y., Paciello, J., & Pane, J. (2017). *Prediction of Dengue Cases in Paraguay Using Artificial Neural Networks*. https://csce.ucmss.com/cr/books/2017/LFS/CSREA2017/HIM3277.pdf

Condor Camara, Daniel, Nolasco Cardenas, Oscar Patricio, Carrasco Escobar, Gabriel & Egoavil Ayala, Miguel (2018*). Sistema de Información basado en Tecnologías de Información y Comunicación para geolocalización de Zika, Dengue, Chikungunya y Malaria.*

http://coloquioenfermeria2018.sld.cu/index.php/coloquio/2018/paper/view/440/96

Navarro Cadavid, Andrés, Fernández Martínez, Juan Daniel, & Morales Vélez, Jonathan (2013). *Revisión de metodologías ágiles para el desarrollo de software.* https://www.redalyc.org/pdf/4962/496250736004.pdf

Molina Ríos, J.R., Zea Ordóñez, M.P., Contento Segarra, M.J. & García Zerda, F.G. (2018). *Comparación de metodologías en aplicaciones web*. 3C Tecnología: glosas de innovación aplicadas a la pyme, 7(1).1-19. DOI: <http://dx.doi.org/10.17993/3ctecno.2018.v7n1e25.1-19/>

Revuelta Arribas, Alberto (2020). *Creación de un complemento para Google Apps Script para la gestión de datos UPM.* http://oa.upm.es/58093/1/TFG\_ALBERTO\_REVUELTA\_ARRIBAS.pdf

Marín, Rafael (2019). *Los gestores de bases de datos más usados en la actualidad*. https://revistadigital.inesem.es/informatica-y-tics/los-gestores-de-bases-de-datos-mas-usados

Lage, Ricardo Juan, Herrera Graña, Tahimí, Simpson Johnson, Bertha & Zulueta Torres, Zulema (2015). *Aspectos actualizados sobre dengue*. https://dialnet.unirioja.es/servlet/articulo?codigo=6027633

Rodríguez-Castro, Aldo Ismael, Rolón, Jorge, & Ríos-González, Carlos Miguel. (2019). *Costos de internación del dengue en un hospital de tercer nivel de atención de Paraguay, 2017*. Revista del Instituto de Medicina Tropical, 14(1), 14-20. https://dx.doi.org/10.18004/imt/201914114-20

Flores, Laura, Giménez Caballero, Edgar, Díaz Duba, Sebastián & Torales, Judith. (2015). *Impacto Económico del dengue en el Instituto de Previsión Social: epidemia diciembre 2006 - julio 2007*. Memorias del Instituto de Investigaciones en Ciencias de la Salud, 13(2), 78-85. https://dx.doi.org/10.18004/Mem.iics/1812-9528/2015.013(02)78-085

Cuellar, C. M. D., Lovera, D., Merlo, O., & Arbo, A. (2020*). Impacto económico del dengue en Paraguay*. Revista chilena de infectología, 37(4), 356-361. doi: 10.4067/S0716-10182020000400356. PMID: 33399655.

Parra, C., Rojas, R., Espinoza, G. A., & Coloma, J (2019). *Participación en programas salud pública desde la comunidad: el caso TopaDengue*. https://www.researchgate.net/publication/334279550\_Participation\_in\_Public\_Health\_Programs\_from\_the\_Community\_the\_TopaDengue\_case

Pane Juan, Ojeda Verena & Valdez, Natalia (2015). *Dengue Open Data*. https://idatosabiertos.org/wp-content/uploads/2015/10/7.Dengue-Pane-Ojeda-Valdez.pdf

Díaz, F. J., & Banchoff Tzancoff, C. M. (2000). *PHP: una solución" open source" para el desarrollo de páginas Web dinámicas. In VI Congreso Argentino de Ciencias de la Computación.* <http://sedici.unlp.edu.ar/handle/10915/23700>

Sierra, F., Acosta, J., Ariza, J., & Salas, M. (2013). *Estudio y análisis de los framework en php basados en el modelo vista controlador para el desarrollo de software orientado a la web. Investigación y desarrollo en TIC, 4(2), 14-26*. <http://revistas.unisimon.edu.co/index.php/identic/article/view/2480>

Sala, J. J. R. (2003). *Introducción a la programación. Teoría y práctica (Vol. 3, p. 2)*. <http://virtual.usalesiana.edu.bo/web/conte/archivos/572.pdf>

Ovando Ortega, D. J. (2019). *Bootstrap y Laravel, herramientas para el desarrollo de aplicaciones web*. http://repositorio.upsin.edu.mx/Fragmentos/tesinas/142016030030OvandoOrtegaDenzelJavier10843.pdf

# 9. ANEXO