**HIVESOFT**

**Plan de Mantenimiento de Software**

Versión 1.0

Elaborado por:

Rosas Saucedo Lenin Gael

Pérez Yam Ana Concepción

Rosas Marín Pablo André

Martínez Olague María Vianey

Morales Reinisch Carlos Morales

Negron May Hebert Jesús

23 de abril de 2023

**Contenido**

[**1. Introducción**](#_195nsfpsvt4q) **3**

[**2. Alcance y propósito**](#_xtzy6nghee7o) **3**

[2.1 Alcance](#_o9fffryfpr53) 3

[2.2 Propósito](#_6btkouxei4ea) 4

[**3. Documentos aplicables**](#_11jygasmkssd) **4**

[**4. Definiciones**](#_to3vcdadfnba) **4**

[**5. Descripción del sistema**](#_b4302crp1uik) **6**

[**6. Roles y responsabilidades**](#_7fxzw6r9prxm) **6**

[**7. Modelo de ciclo de vida**](#_g000ug8y0spy) **8**

[**8. Fase 1: Proceso de implementación**](#_pb9mhpn62nq7) **8**

[**9. Fase 2: Análisis de modificaciones**](#_5jqywpomc8ms) **11**

[**10. Fase 3: Implementación de la modificación**](#_uj5lq8r6tecx) **16**

[**11. Fase 4: Aceptación/Revisión del mantenimiento**](#_8vva744xqy9c) **18**

[**12. Fase 5: Migración**](#_qlyjceduhgzt) **19**

[**13. Fase 6: Retiro del software**](#_9n1nkj7jnxh0) **19**

[**14. Anexos**](#_wntrp3pqbfyv) **19**

[Anexo 1. Solicitud de Modificación](#_yx772e9ob3wu) 20

[Anexo 2. Registro del Historial de la Solicitud de Modificación](#_4rzll46ospx) 20

[Anexo 3. Registro de Pruebas de la Solicitud del Modificación](#_cc6griexml4r) 21

[Anexo 4. Registro de Opciones](#_lxyut51vic9) 21

[Anexo 5. Registro de Asignación de Recursos](#_xqmwo7ypn8wq) 21

[Anexo 6. Registro del Análisis de la Solicitud de Modificación](#_t32wge6b003i) 22

[Anexo 7. Registro del Equipo de Trabajo](#_bpj7qkrup9ys) 22

[Anexo 8. Listado de los Elementos a Modificar](#_29mu6n3lf76l) 23

[Anexo 9. Registro de Pruebas sobre el Sistema Modificado](#_ex1ppx1x2583) 23

[Anexo 10. Registro de Pruebas de Aprobación](#_e58j04l57gto) 23

[Anexo 11. Tabla de Impacto al Sistema](#_n8pwrpm76dj4) 24

[**15. Referencias**](#_ah2equpu7n81) **25**

# 

# 1. Introducción

En este plan de mantenimiento de software se tiene como objetivo garantizar que el módulo de E-commerce de la aplicación *VetPet* funcione de manera óptima, es decir, que cumpla correctamente con las funcionalidades del sistema y no existan errores que pudieran comprometer su rendimiento, que se encuentre disponible para el uso de los clientes y se ajuste a las necesidades del negocio, además, para mantenerla actualizada y segura. Para lograr esto, el plan incluirá diversas actividades de mantenimiento, que se llevarán a cabo de manera periódica.

Este módulo es fundamental para la aplicación completa, debido a que permite a los usuarios adquirir productos relacionados con sus mascotas de manera fácil y rápida dentro de la misma aplicación de *VetPet*, sin tener que ir a páginas exteriores.

El mantenimiento del módulo de E-commerce incluirá la actualización del software, corrección de errores y la adición de nuevas funcionalidades y características. Se llevarán a cabo pruebas para garantizar el correcto funcionamiento del módulo, así como de los cambios y actualizaciones para asegurarse de que estos no hayan afectado la calidad del software.

Además, el plan incluirá medidas de seguridad, como la revisión de vulnerabilidad en el código, para garantizar la protección de los datos de los usuarios y de la aplicación en sí.

Este plan es fundamental para garantizar la disponibilidad, confiabilidad y seguridad de la aplicación en todo momento. Con esto, se podrá garantizar que el módulo funcione según lo esperado y esté actualizado para satisfacer las necesidades de los usuarios y del negocio.

# 2. Alcance y propósito

## 2.1 Alcance

En este documento se presentan los procesos de mantenimiento necesarios para aplicar cada uno de los cuatro tipos de mantenimiento (correctivo, perfectivo, adaptativo y preventivo).

Se busca asegurar que la funcionalidad del módulo de E-commerce se mantenga en un nivel óptimo que cumpla totalmente la satisfacción del cliente en la operación del sistema. Este plan incluye actividades como; la corrección de errores, actualizaciones de seguridad, mejoras de rendimiento y actualizaciones de funcionalidades.

## 2.2 Propósito

Se busca garantizar que el módulo de E-commerce para productos relacionados con mascotas siga siendo útil y relevante para los usuarios finales, manteniendo su calidad y eficiencia. Además, con el plan se quiere establecer una guía para que realizar las modificaciones al sistema sea más sencillo al tener una mejor organización, siguiendo los procesos que se detallarán más adelante en este documento para la documentación y actividades relacionadas al mantenimiento. Con esto, no solo se logrará mayor orden, sino que agilizará las tareas para proporcionar un mejor servicio de mantenimiento para los usuarios, tanto en tiempos como en calidad.

# 3. Documentos aplicables

* IEEE. IEEE Std 14764-2006 Software Engineering - Software Life Cycle Processes - Maintenance.
* V. Martínez, A. Robles, D. Barajas, P. Hernández y N. Escobar. VetPet Software Requirements Specification, 2022.
* IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

# 4. Definiciones

* **Mantenimiento correctivo:** Se encarga de corregir los defectos observados en el software, esta es la forma básica de mantenimiento y consiste en localizar defectos para corregirlos o repararlos y que el sistema pueda funcionar de manera correcta de nuevo. Se realiza luego de que ocurre un fallo en el software debido a que por su naturaleza no pueden planificarse en tiempo, presenta costos por reparación y repuestos no presupuestados, pues incluso puede implicar el cambio de algunas piezas del equipo o modificación de código.
* **Mantenimiento perfectivo:** La principal tarea de este tipo de mantenimiento es llevar a cabo las tareas y procesos necesarios para identificar aquellos puntos susceptibles de mejora, aportando las soluciones óptimas y haciendo efectivos los cambios en las aplicaciones. Este no se enfoca únicamente en mejorar técnicamente una solución, sino que también incluye un proceso continuo de optimización a nivel funcional y de procesos. Hace foco en la optimización constante del rendimiento de las aplicaciones mediante análisis técnicos, la adaptación de las aplicaciones a las nuevas necesidades del cliente en función de los análisis funcionales y la detección de posibles puntos a mejorar en el diseño y uso de las bases de datos mediante el análisis de bases de datos.
* **Mantenimiento adaptativo:** Consiste en adaptar el software para hacer frente a los cambios ocurridos en el entorno, es decir, cambios en las condiciones e influencias que actúan desde el exterior. Algunas razones para que se ejecute este tipo de mantenimiento son cambios en las reglas comerciales, modificaciones en los patrones de trabajo o nuevas políticas gubernamentales que impactan en el software de la organización. Toda modificación hecha a un programa o sistema como respuesta a algún cambio en el entorno se clasifica como mantenimiento adaptativo.
* **Mantenimiento preventivo:** En este tipo de mantenimiento se realizan revisiones periódicas del software para evitar errores de seguridad u operativos; es decir, problemas usuales que pueden aparecer en corto, mediano o largo plazo.
* **Mantenimiento de emergencia:** Modificación que no está programada y se realiza para mantener funcionando a un sistema de manera temporal, teniendo como pendiente la realización de un mantenimiento correctivo.
* **Mantenedor:** Suele ser una persona o un grupo de personas que realizan las actividades de mantenimiento.
* **Plan de Mantenimiento:** Es el conjunto de tareas de mantenimiento programado, agrupadas o no siguiendo algún tipo de criterio, y que incluye a una serie de equipos de trabajo, que habitualmente no son todos, o puede incluir personas externas para realizar las actividades de mantenimiento. En este se especifican las actividades que se realizarán para darle mantenimiento al software.
* **Solicitud de Modificación (MR):** Propuesta de un interesado en el sistema para cambiar cualquier parte del producto. Las solicitudes más comunes incluyen defectos, mejoras del producto y solicitudes de nuevas características.

Las solicitudes de modificación pueden tener los siguientes estados e irán cambiando a lo largo de las fases. Estos son los estados con los que pueden definirse en algún momento dado:

* Aprobado: la solicitud es aprobada para pasar las modificaciones a un ambiente de producción.
* Rechazado: la solicitud no fue aprobada por el equipo de SQA o el líder de equipo.
* Pendiente: Por aprobar. La solicitud no ha sido analizada, y se requiere un análisis previo (está en algún proceso intermedio).
* Finalizado: la solicitud pasó a una fase de cierre. Se ha terminado el ciclo de vida de la solicitud.

# 5. Descripción del sistema

El proyecto VetPet consiste en un sistema que se conforma por múltiples módulos, sin embargo, este plan solo está enfocado en el mantenimiento del módulo de E-commerce. Este se encarga de desplegar una lista de productos relacionados a mascotas. Estos productos se pueden seleccionar y se puede agregar la cantidad que el usuario desee adquirir, posteriormente, el usuario puede ver el total de los productos que ha seleccionado y seleccionar el método de pago que prefiera. El cliente puede buscar entre los productos que se encuentren en el catálogo utilizando la barra de búsqueda.

Adicionalmente, el cliente puede iniciar sesión o crear una cuenta, en ambos casos se puede conectar con Google para realizar este proceso de manera más rápida.

Se tienen dos tipos de usuarios, el cliente y el administrador, debido a que se necesita tener un control de los privilegios de los usuarios, para poder añadir productos nuevos y cambiar o agregar sus atributos, y en general, para la gestión del sistema.

El usuario que tenga el rol de cliente podrá realizar las funcionalidades de compra y orden de productos relacionados a las mascotas, cancelación de alguna orden que se encuentre pendiente o modificación de la orden que realizó. Además, puede escribir una reseña del producto con una valoración de estrellas, para que los demás usuarios puedan ver la reseña al entrar a la descripción más detallada del producto.

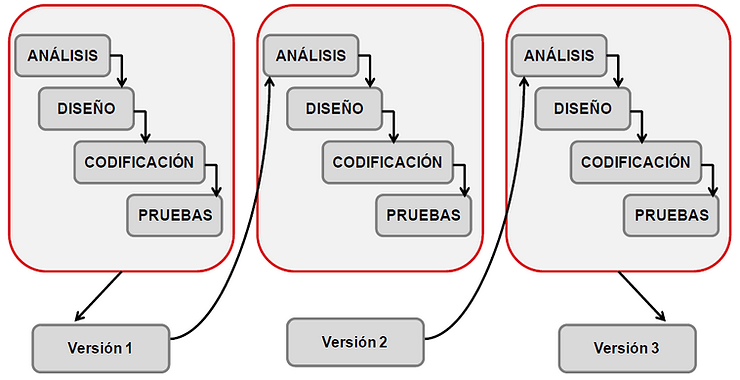
El usuario que tenga el rol de administrador puede modificar los privilegios de los usuarios, agregar nuevos productos, eliminar algún producto, responder a alguna reseña y modificar los atributos de los productos como el precio y cantidad en el inventario, debido a que el inventario se encuentra conectado a una base de datos en donde se tienen los productos registrados.

# 6. Roles y responsabilidades

| **Rol** | **Responsabilidad** | **Responsable** |
| --- | --- | --- |
| Líder de proyecto  (LPR) | El líder de proyecto cuenta con la responsabilidad de planificar y gestionar el mantenimiento del software, esto implica la selección de los miembros responsables del mantenimiento y la asignación de las actividades. Además, debe coordinar y supervisar el equipo de mantenimiento, asegurándose de que los miembros del equipo estén trabajando de manera eficiente y efectiva para mantener y mejorar el software. | Carlos Morales Reinisch |
| Equipo Mantenedor  (EMAN) | Responsables de llevar a cabo las modificaciones en el sistema, incluyendo la actualización de toda la documentación afectada. Para ello, deberá primeramente analizar las solicitudes de cambio entrantes y verificar los problemas que han sido reportados. | Vianey Olague Martínez, Hebert Negrón May |
| Equipo de Soporte  (ESOP) | Su función es la de atender las inquietudes del cliente, recabar toda la información de la solicitud y elaborar la MR que almacenará en el repositorio y servirá como entrada del proceso completo de mantenimiento. | Ana Pérez Yam |
| Equipo de SQA  (ESQA) | Responsable de la elaboración y aplicación del Plan de Aseguramiento de la Calidad (SQAP).  El equipo de SQA debe realizar revisiones y pruebas del software para identificar posibles defectos o problemas en el software, así garantizar que se sigan los estándares y las mejores prácticas de la industria. | Lenin Rosas Saucedo |
| Administrador de la configuración de software (ACS) | Llevar a cabo todas las prácticas relacionadas con la administración de la configuración y control de cambios. Esto incluye la identificación, evaluación, aprobación y seguimiento de los cambios en el software. Debe mantener la documentación actualizada y accesible para todos los miembros del equipo de desarrollo y otros interesados en el software. | Pablo Rosas Marín |

# 7. Modelo de ciclo de vida

Para el desarrollo y mantenimiento del sistema VetPet se sigue un modelo iterativo. Este consiste en varias iteraciones de las etapas de análisis, diseño, codificación y pruebas para obtener versiones mejoradas o con mayor funcionalidades. Para lograr lo anterior, se requiere de la participación del cliente después de cada iteración para evaluar, corregir y proponer mejoras en el producto.



***Figura 2****. Modelo de desarrollo incremental [1]*

# 8. Fase 1: Proceso de implementación

**8.1. Entradas**

* Inconformidad, mejoramiento o deseo de cambios por parte del cliente

**8.2. Salidas**

* Solicitud de Modificación

**8.3. Roles involucrados**

* Cliente
* Equipo de soporte
* Líder del proyecto

**8.4. Actividades**

1. El cliente se comunica con el equipo de soporte eligiendo uno de los medios de comunicación que se presentan a continuación.
2. Zoom (videollamada) - Compartir el enlace de la reunión por número telefónico o correo electrónico, estos siendo +52 999 963 1679 y [A19216291@alumnos.uady.mx](mailto:A19216291@alumnos.uady.mx) respectivamente.
3. Télefono (llamada) - +52 999 963 1679
4. Correo electrónico - [A19216291@alumnos.uady.mx](mailto:A19216291@alumnos.uady.mx)

El listado se presenta con un orden de prioridad ascendente, por lo tanto, el cliente deberá seleccionar los primeros de la lista y acudir al siguiente en caso de no poderse comunicar con el anterior.

1. El miembro del equipo de soporte toma nota de todo lo que el cliente quiere que se cambie o los problemas que se estuvieron presentando, así como el nombre completo del solicitante. En caso de que la comunicación haya sido vía correo electrónico y no se tenga claridad de todos los cambios, el equipo de soporte deberá comunicarse de vuelta con el cliente por este mismo medio o cualquiera de las dos opciones alternativas (con previo acuerdo con el cliente).
2. Al finalizar la recepción de los cambios, el miembro del equipo de soporte procede a rellenar la Solicitud de Modificación ([Anexo 1](#_yx772e9ob3wu)):
   1. Se asigna un número de identificación único para el MR. El número de MR se compone de la siguiente manera:

**identificador del programa o módulo + versión** \_ **fecha** \_ **número de solicitud del día**.

La fecha debe de escribirse con el día en dos dígitos, seguido del mes en dos dígitos y el año completo. Un ejemplo de ID para el MR es E1\_21022023\_1, donde E1 es el identificador del programa, 21022023 la fecha y 1 indica que es la primera Solicitud de Modificación del día.

* 1. Se establece la prioridad del MR para ser atendida con base en la información de la solicitud y la siguiente tabla de prioridades:

| **Tabla de prioridades** | |
| --- | --- |
| **Prioridad** | **Se aplica si un problema:** |
| **1** | * Evita la realización de una actividad esencial para el funcionamiento del software. * Pone en peligro la seguridad del software. |
| **2** | * Afecta negativamente a la realización de una actividad esencial de funcionamiento y no se conoce ninguna solución temporal. |
| **3** | * Afecta negativamente a la realización de una actividad esencial para el funcionamiento del software pero existe una solución temporal. * Puede resultar inconveniente o molesto para el técnico encargado del sistema pero se conoce una solución temporal. |
| **4** | * Presenta molestias al usuario pero no afecta la ejecución de actividades esenciales y no impide el cumplimiento de sus responsabilidades. |
| **5** | * Cualquier otro efecto. |

* 1. Reemplazar las guías incluidas en la plantilla del MR ([Anexo 1](#_yx772e9ob3wu)) por los datos pertinentes en la Sección 1. Es importante que la descripción del cambio sea detallada, incluyendo: la razón del cambio o el problema que se está presentando, descripción de la modificación que se desea efectuar y beneficios esperados tras realizarse la modificación.
  2. El miembro del equipo de soporte guarda la Solicitud de Modificación en la ruta “[documentos/mantenimiento/fase1/solicitudes-cambios/](https://github.com/leningael/vetpet/tree/main/documentos/mantenimiento/fase1/solicitudes-cambios)” del [repositorio](https://github.com/leningael/vetpet) correspondiente. El nombre del documento incluye el prefijo “MR-” junto al número de identificación del MR. (P.ej. MR-E1\_210223\_1.docx).

**8.5. Controles**

8.5.1. Las MR generadas durante esta fase deben ser verificadas para asegurar que se hayan llenado de manera correcta. Para esto, el equipo de SQA debe de emplear la lista de verificación especificada en el [Anexo 2 del Plan de SQA](https://docs.google.com/document/d/1xgUy9CAFdhsJ7jq_at7qDKZZXdhFZUTQuiw_Uxcn4cc/edit#heading=h.a1v1olsoo727).

# 9. Fase 2: Análisis de modificaciones

**9.1 Entradas**

* Solicitud de Modificación
* Documentación del sistema, disponible en la ruta “[documentos/desarrollo](https://github.com/leningael/vetpet/tree/main/documentos/desarrollo)” del repositorio.

**9.2 Salidas**

* Solicitud de Modificación actualizada
* Registro del Historial de la Solicitud de Modificación actualizada
* Registro de Asignación de Recursos
* Registro de Pruebas de la Solicitud de Modificación.
* Registro del Equipo de Trabajo
* Registro del Análisis de la Solicitud de Modificación.

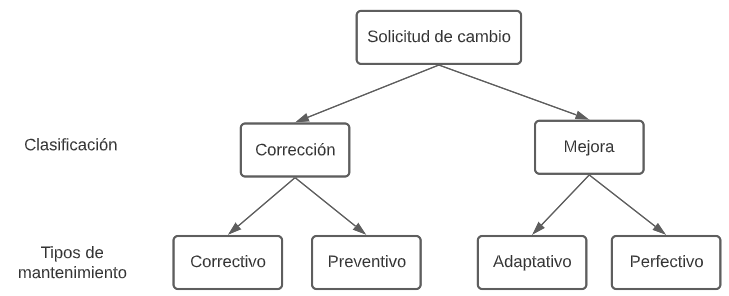
**9.3 Roles involucrados**

* Equipo mantenedor
* Líder de equipo
* Equipo de SQA

**9.4 Actividades**

1. El equipo mantenedor analiza la Solicitud de Modificación para determinar si el tipo de mantenimiento a realizar será preventivo, correctivo, perfectivo o adaptativo. Para esto se debe analizar la redacción de la Solicitud de Modificación y clasificarlo a grandes rasgos entre: mejora o corrección.

Por último, el Equipo Mantenedor seleccionará el tipo de mantenimiento adecuado con base en las descripciones mencionadas en el apartado *4.Definiciones* del presente planpara tener en cuenta procesos y/o actividades aplicables en fases posteriores indicadas en este plan.



***Figura 2****. Tipos de mantenimiento y clasificaciones.*

1. El equipo mantenedor llena el Registro del Historial de la Solicitud de Modificación ([Anexo 2](#_4rzll46ospx)):.

* Si la Solicitud de Modificación no se ha resuelto se llena el historial existente para esta solicitud ubicado en la ruta “documentos/mantenimiento/fase2/regs-historial-mr/H-[*ID del MR*].docx” del repositorio.
* Si la Solicitud de Modificación es nueva en el historial, se procede a llenar un historial en blanco y almacenarlo en la ruta “[documentos/mantenimiento/fase2/regs-historial-mr/](https://github.com/leningael/vetpet/tree/main/documentos/mantenimiento/fase2/regs-historial-mr)” con el nombre incluyendo el prefijo “H-” junto al número de identificación del MR. (P.ej. H-E1\_210223\_1.docx).

En el historial se escriben los datos de la Solicitud de Modificación: ID asignado, la fecha de recepción (dd/mm/aa), fecha de análisis (dd/mm/aa), estado de la solicitud de modificación (los estados del MR se especifican en el apartado *4.Definiciones* del presente plan) y se lista la documentación del sistema afectada hasta el momento.

1. El equipo mantenedor obtiene e instala el software en caso de no contar con este.
2. El equipo mantenedor realiza las pruebas para replicar la falla reportada en la MR, se selecciona a la persona con más experiencia que se encuentre disponible (en caso de ser más de un integrante) para registrar los resultados en el Registro de Pruebas de la Solicitud de Modificación anotando la siguiente información:

* ID de prueba. Se asigna un identificador único para la prueba conformado por el prefijo “RP-”, seguido del número de identificación del MR. Por ejemplo: RP-E1\_210223\_1.
* ID de la Solicitud de Modificación. El ID de la solicitud de modificación sobre la que se está trabajando, tal cual como se indica en el inciso C de las actividades del punto 8.4 de la Fase 1: Proceso de Implementación.
* Nombre del Sistema. El nombre del sistema de software actual sobre el cual se está trabajando agregando el módulo, en este caso *VetPe*t módulo de E-Commerce.
* Responsable de la prueba. El nombre de la persona encargada de llenar el Registro de Pruebas de la Solicitud de Modificación.
* Objetivo. El objetivo de la prueba con el resultado que se espera obtener al replicarlo.
* Resultado de la prueba. El resultado obtenido tras realizarse la prueba. De ser posible, se deberá adjuntar capturas de pantalla que evidencien la falla.

Si el problema no puede ser replicado, debe ser revisada la documentación del sistema para asistir a la identificación del problema. Este inciso, junto con el inciso C no será requerido en caso de realizar un mantenimiento adaptativo y perfectivo.

La plantilla de este Registro de Pruebas de la Solicitud de Modificación se encuentra en el [Anexo 3](#_cc6griexml4r). El registro de pruebas deberá almacenarse en la ruta “[documentos/mantenimiento/fase2/regs-pruebas-mr](https://github.com/leningael/vetpet/tree/main/documentos/mantenimiento/fase2/regs-pruebas-mr)” con el ID de prueba como el nombre del archivo. (P.ej. RP-E1\_210223\_1.docx)

1. El equipo mantenedor debe definir en el Registro de Opciones ([Anexo 4](#_lxyut51vic9)) los requerimientos de la organización como lo son el tiempos y los recursos para llevar a cabo la modificación, así como también el ID de la Solicitud de Modificación que se está analizando, el nombre del sistema: *VetPet* y el encargado del análisis de la Solicitud de Modificación.

De igual manera, se deben proponer al menos dos opciones para resolver la Solicitud de Modificación que serán registrados en el Registro de Opciones, junto con el tiempo estimado que tomará realizar la modificación expresado en días (Extensión), un análisis del impacto en el sistema (Impacto), en una escala Bajo-Medio-Alto, haciendo uso de la Tabla de Impacto al Sistema ([Anexo 11](#_n8pwrpm76dj4)) para decidir en cuál de las tres categorías pertenece la modificación. y los riesgos de implementar el cambio (Riesgo).

Cada opción tendrá un ID formado por el prefijo OP#, donde el # representa el número de la opción, seguido del ID de la Solicitud de Modificación (ejemplo: OP1-E1\_210223\_1). Por último, se podrá agregar una recomendación. El Registro de Opciones generado deberá guardarse en la ruta “[documentos/mantenimiento/fase2/regs-opciones/](https://github.com/leningael/vetpet/tree/main/documentos/mantenimiento/fase2/regs-opciones)” con el nombre incluyendo el prefijo “OP-” y el número de identificación del MR. (P.ej. OP-E1\_210223\_1.docx).

1. El líder de proyecto debe realizar un trabajo de estimación de los recursos humanos y costos que se requiere para cada opción registrada. Esta información se empieza documentando el ID asignado a la Solicitud de Modificación que se está analizando, se especifica el nombre del sistema: *VetPet* y el Responsable de la solicitud de modificación elegido por el equipo de mantenedor.

Siguiendo con la información necesaria para el Registro de Asignación de Recursos, es necesario especificar para cada opción lo siguiente:

* El ID de la opción correspondiente que se asignó en la actividad anterior.
* El número de personas requeridas para realizar la modificación.
* El número estimado de horas totales necesarias para realizar la modificación.
* El costo final estimado, expresado en pesos mexicanos (MXN) y,
* Una observación opcional sobre los recursos identificados.

*Ver plantilla completa en el* [*Anexo 5*](#_xqmwo7ypn8wq)*.*

El Registro de Asignación de Recursos resultante se debe de guardar con el nombre “AR-[número de identificación del MR].docx” (P.ej. AR-E1\_210223\_1.docx) en la ruta “[documentos/mantenimiento/fase2/regs-asignacion-recursos](https://github.com/leningael/vetpet/tree/main/documentos/mantenimiento/fase2/regs-asignacion-recursos)”.

1. El equipo mantenedor en conjunto con el equipo de SQA deben realizar una revisión sobre la documentación realizada acerca de las pruebas y opciones, para asegurarse que se han llenado todas las plantillas con información apegada a los objetivos de la empresa (Ver el punto *9.5.1 Controles*), de igual manera con las estimaciones a los trabajos anteriores de la empresa para lograr una mayor precisión.
2. El equipo mantenedor debe obtener la aprobación del análisis por parte del líder de proyecto. (Ver el punto *9.5.2 Controles*.) Una vez obtenida la aprobación/rechazo de la Solicitud de Modificación, se deberá actualizar el estado de la solicitud en el registro del historial y documentar el resultado en el Registro del Análisis de la Solicitud de Modificación ([Anexo 6](#_t32wge6b003i)). En esta plantilla es necesario describir la siguiente información:

* Nombre del sistema y módulo(s) a: *VetPe*t módulo de E-Commerce.
* La fecha que se recibió la Solicitud de Modificación. Formato: dd/mm/aa.
* Fecha de aprobación/negación. La fecha en la que se aprobó la Solicitud de Modificación, o en su defecto, la fecha en la que la solicitud fue rechazada. Formato: dd/mm/aa.
* Fecha de cierre. La fecha en la que se concluyó con el análisis. Formato: dd/mm/aa.
* ID de la Solicitud de Modificación. Siguiendo el formato ya establecido anteriormente en este plan.
* Descripción de la solución.
* El estado de la Solicitud de Modificación:
  + Si fue aprobada se continúan el resto de las actividades de este plan.
  + Si fue rechazada, no se continúan las actividades de este plan, debido a que no hay mantenimiento que realizar, además, en este mismo Registro de Análisis de la Solicitud de Modificación se debe especificar la razón del rechazo.

El lugar donde se almacenará este Registro de Análisis de la MR es “documentos/mantenimiento/fase2/regs-analisis-mr” con el nombre incluyendo el prefijo “AN-” y el número de identificación del MR. (P.ej. AN-E1\_210223\_1.docx).

Continuando con las actividades, se deberá llenar la sección II de la Solicitud de Modificación.

1. El líder de proyecto debe especificar las personas que estarán involucradas durante las modificaciones al sistema en el Registro del Equipo de Trabajo ([Anexo 7](#_bpj7qkrup9ys)). En este se debe incluir información como:

* Nombre del sistema y módulo(s) a trabajar: en este caso *VetPe*t módulo de E-Commerce*.*
* ID de la Solicitud de Modificación. Siguiendo el formato ya establecido anteriormente en este plan.
* Detalle del mantenimiento. Descripción breve de la modificación a realizar.
* Nombre del miembro del equipo en específico que se encargará de hacer ese cambio, así como su rol dentro del equipo.
* Observación. Adicional, en caso de ser necesario.

El directorio donde se colocará este registro cuenta con la siguiente ruta: “[documentos/mantenimiento/fase2/regs-equipo-trabajo](https://github.com/leningael/vetpet/tree/main/documentos/mantenimiento/fase2/regs-equipo-trabajo)” con el nombre iniciando con “ET-” y continuando con el número de identificación del MR. (P.ej. ET-E1\_210223\_1.docx).

**9.5 Controles**

9.5.1. Para asegurar que las opciones del MR estén documentadas de manera correcta se debe de aplicar la lista de verificación para el Registro de Opciones disponible en el [Anexo 3 del Plan de SQA](https://docs.google.com/document/d/1xgUy9CAFdhsJ7jq_at7qDKZZXdhFZUTQuiw_Uxcn4cc/edit#heading=h.yalg1su0gr5k).

9.5.2. Para la aprobación de la opción a seleccionar se deberá llevar a cabo una revisión entre el líder de proyecto y los miembros mantenedores. Esta se encuentra documentada en el punto [6.2. Revisión para el análisis de la MR](https://docs.google.com/document/d/1xgUy9CAFdhsJ7jq_at7qDKZZXdhFZUTQuiw_Uxcn4cc/edit#heading=h.i7vb15j04c42) del Plan de SQA.

9.5.3. El Registro del Análisis de la Solicitud de Modificación deberá ser revisado por medio de la lista de verificación adjunta en el [Anexo 4 del Plan de SQA](https://docs.google.com/document/d/1xgUy9CAFdhsJ7jq_at7qDKZZXdhFZUTQuiw_Uxcn4cc/edit#heading=h.2ordiq3eylj3).

# 10. Fase 3: Implementación de la modificación

**10.1. Entradas**

* Código fuente, documentación del sistema y salidas de la fase anterior.

**10.2. Salidas**

* Documentación de las pruebas realizadas (Registro de Pruebas sobre el Sistema Modificado).
* Listado de los Elementos a Modificar.
* Código fuente modificado.
* Registro del Historial de la Solicitud de Modificación.

**10.3. Roles involucrados**

* Equipo mantenedor

**10.4. Actividades**

1. El equipo mantenedor debe realizar las modificaciones en el sistema según lo planificado en las fases anteriores. Es por ello que es necesario llevar un registro en el Listado de Elementos a Modificar ([Anexo 8](#_29mu6n3lf76l)), en donde se deben identificar todos los elementos que se verán afectados afectados de alguna manera durante las modificaciones ya establecidas. Este listado deberá incluir:

* Nombre del sistema y módulo(s) a trabajar: en este caso *VetPe*t módulo de E-Commerce.
* Mantenedor. Nombre de él o los encargados en realizar la modificación.
* Número de la modificación. Números que irán de forma ascendente, empezando por el 1.
* Elemento a ser modificado. Se debe especificar de manera corta el elemento que será cambiado, así como los nombres de las clases del sistema que se ven involucrados (ejemplo: sistema de pago - *PaymentScreen.js; OrderActions.js [payOder]*)
* Capa de la arquitectura. Capa del sistema en la que se encuentra situada la modificación que se realizará.
* Descripción de la modificación a realizar. Debe ser breve y responder a la pregunta ¿qué se quiere lograr en el sistema con esta modificación?
* Responsable de la modificación. Nombre de la persona en específico que se encargó de la modificación (en caso de ser más de una persona el modificador indicado al inicio).
* Duración. Tiempo que se tomará para concluir con la modificación. El formato que se me aplicará será en horas (ejemplo: 07 horas, 48 horas).

Se guarda el Listado de Elementos a Modificar en la ruta “[documentos/mantenimiento/fase3/listados-elementos-a-modificar/](https://github.com/leningael/vetpet/tree/main/documentos/mantenimiento/fase3/listados-elementos-a-modificar)” del [repositorio](https://github.com/leningael/vetpet) correspondiente. El nombre del documento incluye el prefijo “LEM-” junto al número de identificación del MR. (P.ej. LEM-E1\_210223\_1.docx).

1. El equipo mantenedor, al finalizar la identificación de los elementos a modificar, debe realizar los cambios al código fuente del sistema, respetando el listado anterior.
2. Al finalizar los cambios se deben ejecutar pruebas en los elementos modificados para verificar el funcionamiento correcto, por lo tanto, se requiere la documentación y establecimiento de criterios para realizarlas. Esta información irá en el Registro de Pruebas sobre el Sistema Modificado ([Anexo 9](#_ex1ppx1x2583)) y debe incluir:

* ID de la Solicitud de Modificación. Siguiendo el formato ya establecido anteriormente en este plan.
* ID de prueba. Siguiendo el formato ya establecido anteriormente en este plan.
* Tipo de prueba. La elección del tipo de prueba dependerá de la modificación que se haya realizado y cómo impacta al sistema. Esta selección debe ser algún tipo de *prueba funcional* debido a que los cambios se están realizando directamente al código fuente.
* Objetivo. Debe ser breve y responder a la pregunta: ¿qué se quiere lograr al finalizar con la prueba?
* Descripción. Debe ser breve y responder a la pregunta ¿qué se hará durante la prueba y con qué?
* Responsables. Nombres del o los encargados de ejecutar la prueba y documentarla.
* Elemento a probar. Se debe especificar de manera corta el elemento que será probado, así como los nombres de las clases del sistema que se ven involucrados (ejemplo: sistema de pago - *PaymentScreen.js; OrderActions.js [payOder]*)
* Precondición. Estado en el que se debe encontrar el sistema antes de comenzar con la prueba.
* Datos de entrada. Entradas para la prueba, en caso de no existir escribir: N/A.
* Resultados esperados. Salidas esperadas de la prueba, descritas antes de su ejecución.
* Resultados obtenidos; Salidas resultantes de la prueba, descritas después de su ejecución.

Se guarda el Registro de Pruebas sobre el Sistema Modificado en la ruta “[documentos/mantenimiento/fase3/regs-pruebas-sistema-modif/](https://github.com/leningael/vetpet/tree/main/documentos/mantenimiento/fase3/regs-pruebas-sistema-modif)” del [repositorio](https://github.com/leningael/vetpet) correspondiente. El nombre del documento incluye el prefijo “RPSM-” junto al número de identificación del MR. (P.ej. RPSM-E1\_210223\_1.docx).

1. El equipo mantenedor debe reunirse para hacer en conjunto la actualización de la información del Registro de Historial de la Solicitud de Modificación. Esta actividad es realizada posterior a la conclusión de todas las modificaciones y su respectiva aplicación de pruebas con resultados satisfactorios.

**10.5. Controles**

* El equipo de SQA debe de aplicar la lista de verificación disponible en el [Anexo 5 del Plan de SQA](https://docs.google.com/document/d/1xgUy9CAFdhsJ7jq_at7qDKZZXdhFZUTQuiw_Uxcn4cc/edit#heading=h.xwtznwuyvn59) para validar que el Registro de Elementos a Modificar cuente con la información requerida.

# 11. Fase 4: Aceptación/Revisión del mantenimiento

**11.1. Entradas**

Para iniciar el proceso de Aceptación/Revisión del Mantenimiento se necesitan los siguientes elementos de entrada:

* El software modificado.
* Línea base modificada.
* El Registro de Pruebas del sistema modificado.

**11.2. Salidas**

* La documentación nueva que contiene las modificaciones aceptadas.
* Modificaciones rechazadas.
* Informe de aceptación.

**11.3. Roles Involucrados**

* Equipo Mantenedor.
* Equipo de SQA.
* Líder de proyecto.

**11.4. Actividades**

1. El equipo mantenedor debe dirigir la nueva línea base (toda la documentación modificada registrado en el Listado de Elementos Modificados para el MR correspondiente) hacia el equipo de SQA, con el cual se emplearán revisiones para:

* Verificar que se puede seguir el proceso documentado por medio de los informes/registros requeridos en las fases anteriores.
* Verificar que solo fueron modificados los componentes de software descritos en el Listado de Elementos a Modificar.
* Verificar que las pruebas sean realizadas y pasadas correctamente.

1. El equipo de SQA determina si fueron satisfechos correctamente los puntos mencionados anteriormente. Además a parte del análisis mencionado previamente, el equipo de SQA deberá realizar una revisión para la aprobación de las modificaciones. (Ver  [6.1 en el Plan de SQA](https://docs.google.com/document/d/1xgUy9CAFdhsJ7jq_at7qDKZZXdhFZUTQuiw_Uxcn4cc/#heading=h.10sc8t46z2bm)).

# 12. Fase 5: Migración

*No aplicable para este proyecto.*

# 13. Fase 6: Retiro del software

*No aplicable para este proyecto.*

# 14. Anexos

Las plantillas que se presentan a continuación se encuentran disponibles en los [documentos de mantenimiento](https://github.com/leningael/vetpet) del repositorio del proyecto. La ruta donde se encuentran está formada de la siguiente manera: “documentos/mantenimiento/fase#/plantillas”, donde # representa el número de fase a la que pertenece el documento.

## Anexo 1. Solicitud de Modificación

| **Solicitud de Modificación (MR)** | |
| --- | --- |
| **Sección 1** | |
| Identificador del MR:  id programa/módulo (Inicial del programa+versión) . fecha . número de solicitud | Fecha de Solicitud:  dd/mm/aa |
| Solicitante:  Nombre completo del solicitante | Prioridad:  Número del 1-5 donde 1 es el máximo de prioridad y 5 el mínimo |
| Sistema:  Nombre del sistema o módulo que se desea modificar | |
| Descripción del cambio:   * Razón del cambio * Descripción de la solicitud * Beneficios esperados | |
| **Sección 2** | |
| Tipo de Mantenimiento: | |
| Responsables del mantenimiento: | |
| **Sección 3** | |
| ID opción: | Estado del MR: |
| Resultados del análisis: | |
| Aprobado por: | |

Archivo: solicitud-modificacion.docx

## Anexo 2.Registro del Historial de la Solicitud de Modificación

| **Registro del Historial de la Solicitud de Modificación** | | | | |
| --- | --- | --- | --- | --- |
| **ID MR** | **Fecha de recepción** | **Fecha de análisis** | **Estado del MR** | **Documentos afectados** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Archivo: historial-mr.docx

## Anexo 3. Registro de Pruebas de la Solicitud del Modificación

| **Registro de Pruebas de la Solicitud del Modificación** | | | | |
| --- | --- | --- | --- | --- |
| **ID Prueba:** | **ID MR:** | **Nombre del sistema:** | | |
| **Responsable de la prueba:** | | | | |
| **Objetivo** |  | | | |
| **Resultado obtenido:** |  | | | |

Archivo: pruebas-mr.docx

## Anexo 4. Registro de Opciones

| **Registro de Opciones** | | | | |
| --- | --- | --- | --- | --- |
| **ID MR:** | **Nombre del sistema:** | | **Responsable:** | |
| **Requerimientos de la organización:** | | | | |
|
| **ID opción** | **Descripción** | **Extensión** | **Impacto** | **Riesgos** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| **Recomendación**: | | | | |
|

Archivo: registro-opciones.docx

## Anexo 5. Registro de Asignación de Recursos

| **Registro de Asignación de Recursos** | | | | |
| --- | --- | --- | --- | --- |
| **ID MR:** | **Nombre del sistema:** | | **Responsable:** | |
| **ID opción** | **Núm. Personas** | **Num Horas** | **Costo** | **Observación** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Archivo: asignacion-recursos.docx

## Anexo 6. Registro del Análisis de la Solicitud de Modificación

| **Registro del Análisis de la Solicitud de Modificación** | | | |
| --- | --- | --- | --- |
| **Nombre del sistema:** |  | | |
| **Responsable(s):** |  | | |
| **Fecha de recepción del MR** |  | | |
| **Fecha de aprobación/negado:** |  | | |
| **Fecha de cierre:** |  | | |
| **ID MR** | **Descripción de la solución** | **Estado del MR** | **Motivo de rechazo** |
|  |  |  |  |
|
| **Responsable de análisis:** | | | |
|
|

Archivo: analisis-mr.docx

## Anexo 7. Registro del Equipo de Trabajo

| **Registro del Equipo de Trabajo** | | | |
| --- | --- | --- | --- |
| **Nombre del sistema:** | **ID MR:** | | |
| **Detalle del mantenimiento** | **Nombre y apellido** | **Rol** | **Observación** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Archivo: registro-equipo-trabajo.docx

## Anexo 8. Listado de los Elementos a Modificar

| **Listado de los Elementos a Modificar** | | | | | |
| --- | --- | --- | --- | --- | --- |
| **Nombre del sistema:** | | | | | |
| **Mantenedor:** | | | | | |
| **Num.** | **Elemento a ser modificado** | **Capa de la arquitectura** | **Descripción de la modificación a realizar** | **Responsable de la modificación** | **Duración** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Archivo: listado-elementos-a-modificar.docx

## Anexo 9. Registro de Pruebas sobre el Sistema Modificado

| **Registro de Pruebas sobre el Sistema Modificado** | | | | |
| --- | --- | --- | --- | --- |
| **ID MR:** | **ID Pruebas:** | | **Fecha de ejecución** | |
| **Nombre del sistema** | | | | |
| **Tipo de prueba** | | | | |
| **Objetivo:** | | | | |
| **Descripción** | | | | |
| **Responsable:** | | | | |
| **Elemento a probar** | **Precondición** | **Datos de entrada** | **Resultados esperados** | **Resultados obtenidos** |
|  |  |  |  |  |
|  |  |  |  |  |

Archivo: registro-pruebas-sobreel-sistema-modificado.docx

## Anexo 10. Registro de Pruebas de Aprobación

| **Registro de Pruebas de Aprobación** | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Id MR:** | | **Id Prueba:** | | **Fecha de ejecución:** | | |
| **Nombre del sistema:** | | | | | | |
| **Tipo de Prueba:** | | | | | | |
| **Objetivo:** | | | | | | |
| **Descripción:** | | | | | | |
| **Responsables:** | | | | | | |
| **Elemento a Probar** | **Precondición** | **Datos de Entrada** | **Resultados esperados** | **Resultados obtenidos** | **Estado** | **Observaciones** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Archivo: pruebas-aprobación.docx

## Anexo 11. Tabla de Impacto al Sistema

| **Tabla de Impacto al Sistema** | |
| --- | --- |
| **Impacto** | **Características de la modificación** |
| Bajo | Se realizan cambios menores en el código de una o dos clases.y en caso de fallas por la modificación, el rendimiento del sistema no se vería comprometido y aun se podría hacer uso de las demás funcionalidades. |
| Medio | Se realizan varios cambios en el código de una hasta tres clases.y en caso de fallas por la modificación, algunas funcionalidades relacionadas se verían afectadas, incapacitando su uso o teniendo resultados erróneos, sin embargo el resto del sistema sería ejecutándose normalmente. |
| Alto | Se realizan gran cantidad de cambios en el código de más de dos clases.y en caso de fallas por la modificación, todo el sistema se vería afectado, produciendo resultados erróneos o en el peor de los casos, haciendo que el sistema completo deje de funcionar. |

# 15. Referencias

1. Efecto Digital. *Ciclo de vida de desarrollo de Software*. <https://www.efectodigital.online/single-post/2018/04/23/ciclo-de-vida-de-desarrollo-de-software>, Abril del 2018.
2. A. Aguileta, Plantilla para el plan de mantenimiento de software, 2023