**Plan de pruebas de software**

***Sistema de Gestión Documental Histórico para Ayuntamientos.***

***Fecha: 08/10/2024***

Tabla de contenido

[Historial de versiones 4](#_Toc179315619)

[Información del proyecto 4](#_Toc179315620)

[Aprobaciones 5](#_Toc179315621)

[Resumen ejecutivo 5](#_Toc179315622)

[Alcance de las pruebas 6](#_Toc179315623)

[Elementos de pruebas: 6](#_Toc179315624)

[Nuevas funcionalidades a probar 6](#_Toc179315625)

[Pruebas de regresión 7](#_Toc179315626)

[Funcionalidades a no probar 8](#_Toc179315627)

[Enfoque de pruebas (estrategia) 8](#_Toc179315628)

[Criterios de aceptación o rechazo 10](#_Toc179315629)

[Criterios de aceptación o rechazo 10](#_Toc179315630)

[Criterios de suspensión 10](#_Toc179315631)

[Criterios de reanudación 10](#_Toc179315632)

[Entregables 11](#_Toc179315633)

[Recursos 11](#_Toc179315634)

[Requerimientos de entornos – Hardware 12](#_Toc179315635)

[Requerimientos de entornos – Software 12](#_Toc179315636)

[Herramientas de pruebas requeridas 12](#_Toc179315637)

[Personal 13](#_Toc179315638)

[Entrenamiento 13](#_Toc179315639)

[Planificación y organización 14](#_Toc179315640)

[Procedimientos para las pruebas 15](#_Toc179315641)

[Matriz de responsabilidades 15](#_Toc179315642)

[Cronograma 16](#_Toc179315643)

[Premisas 16](#_Toc179315644)

[Dependencias y Riesgos 17](#_Toc179315645)

[Referencias 19](#_Toc179315646)

[Glosario 20](#_Toc179315647)

# Historial de versiones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fecha** | **Versión** | **Autor (es)** | **Organización** | **Descripción** |
| **8 de octubre de 2024** | **1.0** | Omar Yoval Aviles  Axel Arzate Neri | OptiLinex S.A de C.V | **La primera versión incluye los siguientes módulos:**   * **Expediente: Representación de cada expediente en formato pdf** * **Documento: Cada expediente puede contener múltiples documentos, lo cual permite su clasificación y gestión** * **Usuario: Gestiona los roles y accesos al sistema.** * **Clasificación Archivista: Define las claves alfanuméricas y su estructura.** |
|  |  |  |  |  |

# Información del proyecto

|  |  |
| --- | --- |
| Empresa / Organización | OptiLinex S.A de C.V |
| Proyecto | Sistema de Gestión Documental Histórico para Ayuntamientos. |
| Fecha de preparación | 8 de octubre de 2024 |
| Cliente | Ayuntamientos |
| Patrocinador principal | Miguel Angel Lopez Villar, invertirá en el proyecto de acuerdo a las necesidades, ademas se pronostica una participación de la empresa, en un 45% |
| Gerente / Líder de proyecto | Omar Yoval Aviles |
| Gerente / Líder de pruebas de software | Axel Arzate Neri |

# Aprobaciones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nombre y Apellido** | **Cargo** | **Departamento u organización** | **Fecha** | **Firma** |
| Axel Arzate Neri | Líder de pruebas de software | Departamento IT | 08/10/2024 | 22 ideas de Firmas geniales en 2024 | firmas geniales, diseño de firmas,  ejemplos de firmas |
| Omar Yoval Aviles | Líder de proyectos | Departamento de Desarrollo de proyecto | 10/10/2024 |  |
| Miguel Angel Lopez Villar | Encargado en Base de datos | Departamento IT | 10/10/2024 |  |
| Carlos Flores Valdespino | Encargado de seguridad | Departamento de ciberseguridad | 12/10/2024 | 11 ideas de Firmas | letras para firmas, firmas geniales, ejemplos de firmas |

# Resumen ejecutivo

Este Plan de Pruebas de Software tiene como propósito garantizar la calidad y el correcto funcionamiento del Sistema de Gestión Documental Histórico para el Ayuntamiento, desarrollado en Laravel.

El principal objetivo es gestionar y organizar expedientes y documentos en distintas áreas del Ayuntamiento (trámite, concentración e históricos) y controlar el acceso a los usuarios basado en roles y permisos. Además, incluye una integración con un sistema IoT para el monitoreo de la temperatura en los archivos. Esto se hace con el objetivo de mantener los documentos históricos, en excelentes condiciones, por ejemplo se deben preservar en temperatura, luz y un ambiente, en el cual no se afecte la perdida de información.

Este plan maestro pretende cubrir todas las fases del ciclo de pruebas de un software, desde las pruebas unitarias y de integración hasta las pruebas de aceptación del usuario y pruebas de seguridad. Se aborda tanto los aspectos funcionales como los aspectos no funcionales, la seguridad, la interfaz de usuario y la eficacia del sistema.

# Alcance de las pruebas

# Elementos de pruebas:

* Pruebas de autenticación y gestión de usuarios: verificar el control de acceso basado en roles, asegurando que ciertos usuarios solo puedan acceder a las áreas a las que tienen permisos.
* Pruebas funcionales: validación del CRUD completo, que es la creación, consulta, modificación y eliminación de expedientes, la carga de documentos en pdf, así como la correcta clasificación archivista.
* Pruebas IoT: verificación del monitoreo y control de temperatura en tiempo real, con la interacción entre los sensores dentro del archivo y el sistema de gestión.
* Pruebas de seguridad: validación del acceso no autorizado a áreas protegidas, así como la protección de los archivos y datos almacenados.
* Pruebas de interfaz de usuario (UI/UX): validación de la presentación visual, responsiva en diferentes dispositivos y el uso amigable de la interfaz con el usuario.

## Nuevas funcionalidades a probar

Las funcionalidades a probar son:

* Registro y Gestión de Expedientes: Los empleados del archivo, podrán registrar expedientes de manera individual y múltiples documentos y gestionar su estado para su clasificación y consulta.
* Clasificación Archivista: Los empleados introducirá claves alfanuméricas para organizar los expedientes, facilitando su localización y gestión.
* Gestión de Usuarios y Roles: El encargado del archivo general, podría dar de alta, editar y eliminar usuarios y su información, así como asignarle el rol para su acceso a diferentes áreas.
* Monitoreo de Temperatura: El encargado del archivo podrá visualizar en tiempo real los datos de temperatura del archivo.
* Consulta de Documentos: Los empleados, según su área y permisos, podrán buscar y consultar documentos específicos dentro de cada expediente.

## Pruebas de regresión

De acuerdo con Sánchez 2015 Sobre Pruebas de Software, fundamentos y técnicas, disponible en: https://oa.upm.es/40012/1/PFC\_JOSE\_MANUEL\_SANCHEZ\_PENO\_3.pdf   
  
“Las pruebas de regresión son aquellas que se realizan cuando el software ha sufrido un cambio, ya que este cambio se puede tratar de un nuevo modulo dentro del programa o de cambios que se han hecho en el software al solucionar errores

De acuerdo con esta definición, no se realizarán pruebas de regresión por el momento, ya que la versión 0100 abarca desde el desarrollo hasta la implementación del sistema. Al ser la primera versión, se llevará a cabo una revisión e instalación integral de todo el sistema. Además, no se planea añadir nuevas funcionalidades al software de forma inmediata, o en un corto periodo de tiempo.

## Funcionalidades a no probar

Listado de las funcionalidades que no se van a probar. Debe incluir información de las razones por las cuales no se van a probar y los riesgos que se están asumiendo.

1. Exportar archivos a formatos poco o muy conocidos: Estos formatos no son críticos para el uso diario del sistema, por lo que no se consideran prioritarios en esta fase.
2. Búsquedas avanzadas si no son esenciales: Se enfocará en probar las búsquedas básicas que cubran las necesidades principales de los usuarios.
3. Traducción automática a otros idiomas: No es una necesidad inmediata del sistema, ya que se utilizará en un entorno de habla local.
4. Alertas o notificaciones por cambios menores en documentos: Estos cambios no afectan significativamente el funcionamiento general del sistema.

## Enfoque de pruebas (estrategia)

Tipos de Pruebas.

* Pruebas Funcionales: Se validan todas las funcionalidades del sistema para asegurar que cumplen con los requisitos, las validaciones se harán en los módulos de gestión de usuarios, la gestión documental y la interfaz de usuario.
* Pruebas de Desempeño: Se evaluará la eficiencia del sistema bajo condiciones de carga normal y alta. Las pruebas de tiempo de respuesta y de capacidad de soporte del sistema con múltiples usuarios activos a la vez.
* Pruebas de Interfaces: Se revisarán las integraciones con otros sistemas, aunque no se modificarán, para garantizar que continúen funcionando correctamente.
* Pruebas No Funcionales: Aspectos como la usabilidad, la seguridad, y la confiabilidad del sistema, asegurando que el sistema no solo funcione bien, sino que también proporcione una buena experiencia de usuario.

Requerimientos Especiales de Pruebas:

* Ambiente de Pruebas: Se utilizarán datos de prueba que muestren la producción en las diferentes documentos y usuarios del sistema.
* Herramientas de Pruebas: Se harán pruebas manuales, con el propósito de probar la eficiencia y eficacia del sistema.

Configuraciones Para Probar:

* Se verificarán diferentes configuraciones del sistema, como la configuración del rol del usuario, así como variaciones en los sensores para asegurar que el sistema actúe de manera correcta.

# Criterios de aceptación o rechazo

## Criterios de aceptación o rechazo

**Pruebas Funcionales**:

* Aceptación: Las funcionalidades de gestión de usuarios, gestión documental e interfaz funcionan como se espera.
* Rechazo: Falla en cumplir con los requisitos establecidos o errores en funcionalidades clave, como el menú, herramientas o características.

**Pruebas de Desempeño**:

* Aceptación: El sistema responde rápidamente y maneja múltiples usuarios sin problemas.
* Rechazo: Tiempo de respuesta lento o incapacidad para soportar una alta carga de usuarios.

**Pruebas de Interfaces**:

* Aceptación: Las integraciones con otros sistemas funcionan correctamente.
* Rechazo: Fallas en la conexión o mal funcionamiento de las integraciones.

**Pruebas No Funcionales**:

* Aceptación: El sistema es fácil de usar, seguro y confiable.
* Rechazo: Problemas de usabilidad, brechas de seguridad o inestabilidad.

## Criterios de suspensión

 **Pruebas Funcionales**:

* Fallas graves en la gestión de usuarios, gestión documental o interfaz.
* **Casos fallidos**: Si el 20% o más de los casos de prueba fallan, indicando problemas serios.

 **Pruebas de Desempeño**:

* Incapacidad para manejar la carga esperada o tiempos de respuesta inaceptables.

 **Pruebas de Interfaces**:

* Fallas repetidas en integraciones clave que impiden el acceso a servicios.

 **Pruebas No Funcionales**:

* Problemas graves de seguridad o usabilidad.
* **Nuevos requisitos**: Cambios importantes que afecten el alcance de las pruebas deben detener las pruebas para replantear el enfoque.

## Criterios de reanudación

* Correcciones: Se realizarán pruebas de verificación para confirmar que las soluciones implementadas han abordado adecuadamente los casos fallidos. Esto asegurará que los cambios no hayan provocado de nuevos problemas en el sistema.
* Implementación de los nuevos de Requisitos: Las pruebas se reanudarán una vez que se hayan actualizado los casos de prueba y se haya atendidos los enfoque con los nuevos requisitos.

# Entregables

Establece que se entregará como parte de la ejecución del plan, por ejemplo: Documento de plan de pruebas, casos de pruebas, especificación de diseño de casos, logs de errores, reportes de incidencias, evidencias de pruebas, reportes emitidos por herramientas de pruebas y cualquier otro que se establezca

* Documento del Plan de Pruebas: Documento que incluye la estrategia de pruebas, el alcance, los criterios de suspensión y reanudación, así como los recursos y responsabilidades asignadas.
* Evidencias de Pruebas: Capturas de pantalla, que demuestre la ejecución de los casos de prueba y los resultados obtenidos.

# Recursos

## Requerimientos de entornos – Hardware

1. PC de Pruebas: Para el desarrollo y pruebas del Sistema de Gestión Documental Histórico para Ayuntamientos, se requiere una computadora con un procesador Intel Core i3 o superior, 8 GB de RAM para manejar múltiples tareas sin interrupciones, un almacenamiento de al menos 512 GB SSD para mejorar la velocidad de acceso a archivos, y sistema operativo Windows 10 Pro o Ubuntu 20.04. Estos recursos permiten realizar pruebas en condiciones reales simulando un entorno de trabajo diario en el ayuntamiento.
2. Servidor: Se recomienda un servidor dedicado para almacenar los documentos históricos y manejar el tráfico de usuarios. El servidor deberá contar con al menos un procesador Intel Xeon E5, 32 GB de RAM, y 2 TB de almacenamiento RAID para asegurar la redundancia y seguridad de los datos.
3. Acceso a Internet: Es esencial una conexión de 100 Mbps como mínimo, tanto para el acceso al sistema en línea como para sincronización de documentos entre los distintos ayuntamientos que accedan al sistema. Esto permitirá un manejo eficiente del sistema de gestión documental en tiempo real, sobre todo en la subida y descarga de archivos de gran tamaño.
4. Sensores de Temperatura y Humedad: Los archivos históricos deben ser monitoreados para asegurar su conservación en las condiciones correctas. Se requiere el uso de sensores de temperatura y humedad conectados a microcontroladores Arduino. Estos sensores proporcionarán datos en tiempo real para asegurar que las condiciones del archivo físico se mantengan dentro de parámetros adecuados.
5. Escáner de Documentos: Un escáner de alta resolución, capaz de procesar grandes volúmenes de documentos históricos en diferentes tamaños, como A3 y A4, será esencial para la digitalización de archivos. Se recomienda un escáner con capacidad de hasta 1200 DPI para asegurar la legibilidad y preservación digital de los documentos.
6. Impresora: Se necesita una impresora multifunción, preferiblemente láser, que permita imprimir reportes y documentos relacionados con los expedientes digitalizados.

## Requerimientos de entornos – Software

Lista de los requerimientos de software necesarios para completar las actividades de prueba, en la siguiente lista se detallan los requerimientos.

1. Lenguaje de Programación: El sistema estará desarrollado principalmente en Laravel 11 para el backend, aprovechando sus características de seguridad, escalabilidad y manejo eficiente de bases de datos.
2. Base de Datos: El sistema utilizará MySQL o PostgreSQL como motor de base de datos relacional, asegurando una gestión eficiente de los documentos digitalizados y metadatos históricos. La base de datos deberá estar optimizada para manejar grandes volúmenes de datos, con funciones de búsqueda avanzadas.
3. Integración de APIs: Para el manejo de documentos escaneados y su vinculación al sistema, se utilizará la API de OCR (Reconocimiento Óptico de Caracteres) para convertir los documentos escaneados en texto editable, mejorando la accesibilidad y búsqueda dentro del sistema.
4. Seguridad: El sistema contará con un esquema de seguridad robusto, utilizando encriptación AES de 256 bits para los documentos almacenados y TLS/SSL para las conexiones. Los usuarios tendrán autenticación de dos factores (2FA) para proteger el acceso a la plataforma.
5. Monitoreo y Reportes: Los sensores de temperatura y humedad estarán conectados al sistema mediante un módulo IoT, permitiendo generar alertas automáticas si las condiciones no son adecuadas. Además, el sistema ofrecerá reportes detallados sobre el uso de documentos y la conservación de los archivos físicos.

## Herramientas de pruebas requeridas

1.Herramientas para pruebas unitarias:

Laravel ofrece soporte integrado para pruebas unitarias usando PHPUnit, que es el estándar en la comunidad PHP. Algunas herramientas útiles son:

* PHPUnit: Es la biblioteca principal para pruebas unitarias en Laravel. Viene preconfigurada en Laravel.

1. Pruebas de integración

* Laravel BrowserKitTesting: Permite simular interacciones HTTP en tu aplicación.

1. Pruebas de desempeño:

* JMeter: Herramienta ampliamente utilizada para pruebas de carga.
* Xdebug: Aunque es un depurador, puede ayudarte a medir el tiempo que tarda tu código en ejecutarse.

4.Pruebas de interfaz de usuario:

* Laravel Dusk: Herramienta integrada en Laravel para realizar pruebas de navegador automatizadas. Simula clics, formularios, interacciones del usuario, etc.

## Personal

* Gerente/líder de pruebas de software: Axel Arzate Neri tiene la capacidad de supervisar todo el proceso de pruebas, coordinando al equipo de manera eficiente.
* Líder de proyecto: Omar Yoval Aviles posee las habilidades técnicas necesarias para ejecutar casos de prueba de manera eficiente y detectar cualquier defecto en el sistema.
* Encargado de Base de Datos: Miguel tiene la capacidad de gestionar la base de datos de forma óptima.
* Encargado de Seguridad: Carlos está capacitado para garantizar que se cumplan todas las normas de seguridad en el sistema.

## Entrenamiento

Entrenamiento en el Sistema de Gestión Documental:

* Familiarizar al grupo con la funcionalidad y características del sistema.
* Capacitación sobre cómo trabaja del sistema, gestión de usuarios y roles, manejo de documentos y el módulo de monitoreo de temperatura.

# Planificación y organización

## Procedimientos para las pruebas

Se empleará una combinación de metodologías ágiles y en cascada para lograr una ejecución flexible y organizada de las pruebas. En los desarrollos iniciales del sistema, se seguirá un enfoque en cascada para pruebas unitarias y de integración.

1. Pruebas Unitarias: En esta fase, cada módulo del sistema será probado de forma independiente para asegurar que funcione según lo esperado.
2. Pruebas de Integración: Una vez que los módulos individuales han sido probados, se evaluará la interacción entre ellos para garantizar que trabajen en conjunto sin errores.
3. Pruebas Funcionales: Se realizarán pruebas en el sistema completo para asegurar que todas las funcionalidades cumplen con los requisitos especificados.
4. Pruebas de Seguridad: Se implementarán pruebas que evalúen la seguridad del sistema, verificando la protección de los datos y los mecanismos de autenticación.
5. Pruebas de Rendimiento y Carga: Estas pruebas evaluarán la capacidad del sistema para manejar grandes volúmenes de datos y múltiples usuarios simultáneamente.
6. Pruebas de Sistema: Evaluara el sistema completo en su conjunto. Su objetivo es verificar que todos los componentes del software funcionen correctamente.

**Planificación de Pruebas**:

El plan de pruebas detallado incluirá el siguiente objetivo: Es asegurar que el sistema de gestión documental funcione sin errores, que las funcionalidades operen según lo esperado y que los datos históricos estén protegidos de forma adecuada.

## Matriz de responsabilidades

Leyenda RACI

* R (Responsable): Quien realiza la tarea.
* A (Aprobador): Quien toma la decisión final o aprueba el resultado.
* C (Consultado): Quien debe ser consultado antes de tomar decisiones o ejecutar tareas.
* I (Informado): Quien debe ser mantenido al tanto del progreso y los resultados.

Esta matriz asegura que cada persona esté clara sobre su rol en cada actividad del plan de pruebas.

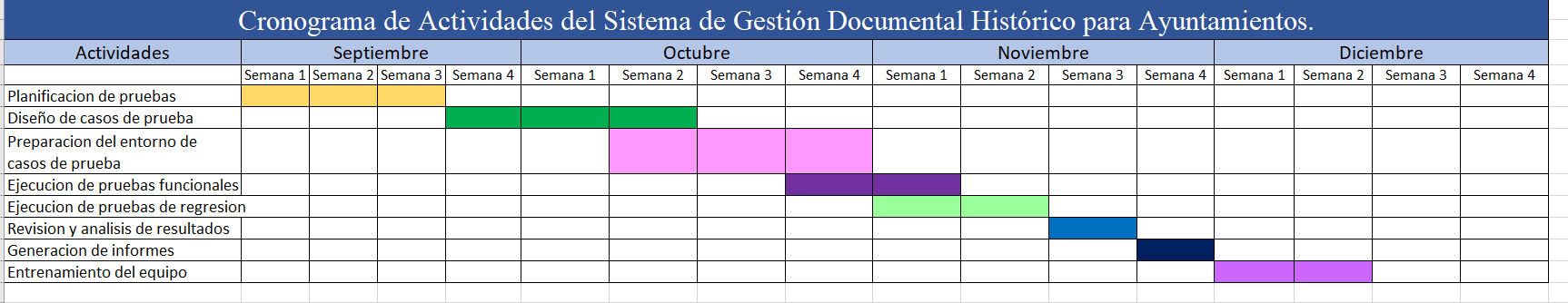
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Actividad | Axel Arzate Neri (Gerente/líder de pruebas de software | Carlos Flores Valdespino (Encargado de Seguridad) | Miguel Angel Lopez Villar (Encargado en Base de Datos) | Omar Yoval Avilés (Líder de Proyectos) |
| Planificación de Pruebas | R, A | C | I | I |
| Diseño de Casos de Prueba | R | C | C | I |
| Ejecución de Pruebas | C | R | A | I |
| Gestión de Defectos | R | R | C | I |
| Revisión de Resultados | R | R | I | C |
| Generación de Informes | R | C | I | A |
| Cierre de Pruebas | R | C | I | A |

|  |
| --- |
|  |

|  |
| --- |
|  |

## Cronograma

En el siguiente cronograma de actividades, se plantean el conjunto de actividades a desarrollar en un periodo de 4 meses, los cuales abarcan de septiembre a diciembre de 2024



## Premisas

* Disponibilidad de Recursos: Se considera que todos los recursos necesarios, incluyendo hardware, software y personal, estarán disponibles según lo planeado. La falta de recursos puede resultar en un retraso en las actividades de prueba.
* Metodología de Pruebas: Se adoptará una metodología ágil para la ejecución de pruebas, permitiendo una revisión continua y ajustes en los casos de prueba según la evolución del desarrollo.
* Compromiso del Personal: Se espera que todos los integrantes del equipo se comprometan y participen activamente en las actividades, respetando las fechas establecidas.
* Revisión Continua: Se asumirá que habrá revisiones periódicas del progreso de las pruebas y ajustes en el enfoque si es necesario.
* Accesibilidad del Sistema: Se garantiza que el sistema será accesible para los usuarios de prueba, con las credenciales y permisos necesarios para llevar a cabo las pruebas de funcionalidad y seguridad.

## Dependencias y Riesgos

Dependencias con Desarrollos

* Probabilidad: Alta
* Impacto: Alto
* Mitigación: Mantener una comunicación constante con el equipo de desarrollo para asegurarse de que los cambios y actualizaciones se notifiquen con antelación.
* Contingencia: Si se produce un retraso en el desarrollo, se debe ajustar el cronograma de pruebas.

Dependencias con Otros Proyectos

* Probabilidad: Media
* Impacto: Alto
* Mitigación: Identificar las dependencias de otros proyectos y coordinar con los equipos correspondientes para ajustar los plazos.
* Contingencia: Establecer prioridades en las pruebas y documentar las áreas afectadas para su revisión posterior.

Disponibilidad de Recursos

* Probabilidad: Alta
* Impacto: Medio
* Mitigación: Planificar con anticipación la asignación de recursos y tener un equipo de respaldo en caso de que algún miembro no esté disponible.
* Contingencia: Redistribuir las tareas entre el equipo existente.

**Riesgos**

Restricciones de Tiempo

* Probabilidad: Alta
* Impacto: Alto
* Mitigación: Establecer un cronograma con márgenes para imprevistos y revisar periódicamente el progreso.
* Contingencia: Reducir el alcance de las pruebas en caso de que el tiempo se vuelva un problema.

Premisas que Resulten No Ser Ciertas

* Probabilidad: Media
* Impacto: Medio
* Mitigación: Realizar revisiones periódicas de las premisas y ajustarlas según sea necesario, basándose en el avance del proyecto.
* Contingencia: Documentar cualquier cambio y adaptar el plan de pruebas según la nueva información.

# Referencias

1. Sánchez 2015 Sobre Pruebas de Software, fundamentos y técnicas, disponible en: https://oa.upm.es/40012/1/PFC\_JOSE\_MANUEL\_SANCHEZ\_PENO\_3.pdf
2. Márquez, F. J., & Dorronsoro, J. R. (2019). Pruebas de software: técnicas, herramientas y metodologías. Ediciones Universidad de Salamanca. https://www.edicionesusal.es/
3. Pérez, J. A., & Rodríguez, A. (2018). Gestión de proyectos de software. McGraw-Hill. https://www.mcgraw-hill.com/
4. Pressman, R. S. (2015). Ingeniería del Software: Un Enfoque Práctico. McGraw-Hill. https://www.mcgraw-hill.com/
5. Gutiérrez, J. (2020). Pruebas de software: Manual práctico de testing. Editorial UOC. https://editorial.uoc.edu/
6. Mendoza, J. (2021). Metodologías ágiles y pruebas de software. Fundación Universitaria Iberoamericana. https://www.fundacioniberoamericana.org/

# Glosario

1. Aceptación: Proceso mediante el cual se determina si un sistema cumple con los requisitos y está listo para su entrega.
2. Automatización: Uso de herramientas para ejecutar pruebas sin intervención manual.
3. Caso de prueba: Documento que describe una condición o variable para verificar que una aplicación funciona correctamente.
4. Criterios de suspensión: Condiciones bajo las cuales se detienen las pruebas.
5. Criterios de reanudación: Condiciones necesarias para continuar las pruebas después de una suspensión.
6. Defecto: Error o falla en el software que impide que cumpla con los requisitos.
7. Entorno de prueba: Configuración de hardware y software donde se realizan las pruebas.
8. Estabilidad: Capacidad del software para funcionar sin fallas durante un período prolongado.
9. Errores: Inconsistencias o problemas en el software que deben corregirse.
10. Funcionalidad: Conjunto de características y capacidades que un sistema debe ofrecer.
11. Interfaz: Punto de interacción entre diferentes sistemas o componentes.
12. Matriz RACI: Herramienta que define roles y responsabilidades dentro de un equipo.
13. Pruebas de regresión: Evaluación para garantizar que nuevas modificaciones no afecten funciones existentes.
14. Pruebas de seguridad: Evaluación de la protección de datos y acceso no autorizado.
15. Pruebas funcionales: Evaluación del comportamiento del software en base a los requisitos.
16. Requisitos: Especificaciones que definen lo que debe hacer un sistema.
17. Riesgo: Posibilidad de que ocurra un evento negativo en el proyecto.
18. Sistema IoT: Infraestructura que permite la interconexión de dispositivos para recopilar y compartir datos.
19. Testeo: Proceso de evaluación de un sistema para identificar defectos y asegurar su calidad.
20. Validación: Proceso de comprobar que el sistema cumple con las necesidades y expectativas del cliente.