



**El futuro  
es de todos**

**DNP**  
Departamento  
Nacional de Planeación

**GUIA PLAN MAESTRO DE PRUEBAS**

**Oficina de Tecnologías y sistemas de Información  
Grupo de Gestión de Sistemas de Información**

**Departamento Nacional de Planeación**

**Bogotá, junio 2020**

## Control de Versiones

Versión	Fecha	Descripción	Autores
0	30-07-2020	Elaboración del documento	Clara Inés Martínez Pineda

**Derechos de Autor:** La elaboración de este documento y sus diferentes componentes estuvo a cargo del Grupo de Gestión de Sistemas de Información de la Oficina de Tecnologías y Sistemas de Información del Departamento Nacional de Planeación, DNP, razón por la cual los Derechos de Autor y en lo particular los derechos patrimoniales de este documento y su contenido pertenece exclusivamente al DNP. Por lo tanto, su uso y reproducción por terceros, está sujeto a la autorización expresa de la Oficina de Tecnologías y Sistemas de Información, OTSI del DNP en cumplimiento de la Ley 23 de 1982 y demás que la modifican o adicionan. Siendo así, este documento está protegido por Derechos de Autor y no puede ser copiados, ni reproducidos, ni distribuidos por personas o Entidades diferentes al DNP.

## TABLA DE CONTENIDO

1. OBJETIVO .....	4
2. PROPÓSITO .....	4
3. TÉRMINOS Y DEFINICIONES .....	4
4. ALCANCE .....	4
5. CONTEXTO .....	4
6. ESTRATEGIA Y TIPOS DE PRUEBAS .....	5
6.1. Pruebas Funcionales .....	5
6.2 Pruebas No Funcionales .....	6
7. RECURSO HUMANO Y CRONOGRAMA .....	8
8. PLANIFICACION DEL PLAN MAESTRO DE PRUEBAS .....	9
9. ALCANCE DEL PLAN MAESTRO DE PRUEBAS .....	10
10. PROCESO GENERAL DE PRUEBAS, CRITERIOS DE INICIO Y SUSPENSIÓN DE PRUEBAS .....	11
11. AMBIENTES REQUERIDOS PARA PRUEBAS .....	11
12. RIESGOS .....	12
13. FINALIZACION DE LAS PRUEBAS .....	12
14. REFERENCIAS .....	13

	<b>GUIA PLAN MAESTRO DE PRUEBAS</b>	CÓDIGO: GUIA - PLAN MAESTRO DE PRUEBAS
		PÁGINA: 4 de 13

## 1. OBJETIVO

El presente documento contiene las directrices que la Oficina de Tecnologías y Sistemas de Información –OTSI- del DNP establece para la estrategia, planeación, ejecución y manejo de los resultados de la disciplina de pruebas para los Sistemas de Información que estén en etapa de desarrollo.

## 2. PROPÓSITO

El Plan Maestro de Pruebas es un documento que describe como planear, programar y controlar las actividades y el esfuerzo de la disciplina de pruebas. En este documento se documenta: el alcance, los objetivos, la estrategia y tipos de pruebas, el equipo humano requerido en términos de roles y responsabilidades, las herramientas que se utilizarán, la clasificación o categorización de los defectos, y los criterios para finalización del proceso de pruebas.

## 3. TÉRMINOS Y DEFINICIONES

**Plan de pruebas:** documento que indica de forma general la táctica y estrategia para la aplicación de las pruebas al sistema de información

**Caso de prueba:** Un conjunto de pasos y resultados esperados que se crean a partir de los requisitos del software que se va a probar.

**Suite de pruebas:** Es una colección de casos de prueba

**Pruebas de regresión:** una recopilación de casos de prueba que se utilizan para realizar una revisión a las funcionales ya entregadas y aprobadas.

## 4. ALCANCE

Esta guía sirve como insumo para la creación del Plan de Pruebas del nuevo Sistema de Información, y debe ser aplicada tanto por funcionarios como por contratistas y firmas externas contratadas por el DNP.

## 5. CONTEXTO

Los proyectos que construyen Sistemas de Información para el DNP están enmarcados dentro de los lineamientos y directrices publicados por la OTSI, sin perjuicio de aplicar también, siempre y cuando no entren en contradicción, las buenas prácticas de la industria.

Para la elaboración, la creación y ejecución de casos de prueba se requiere como insumo principal los requerimientos funcionales y no funcionales, estos requerimientos deben estar aprobados y su contenido debe ser verificable, una vez están definidas las fechas de entrega de cada iteración, se deben comenzar a diseñar los casos de prueba.

El cronograma de pruebas se hará de acuerdo a los objetivos planeados para cada iteración y será actualizado y administrado por el gerente funcional de la dependencia.

## 6. ESTRATEGIA Y TIPOS DE PRUEBAS

El Plan Maestro de Pruebas debe indicar los tipos de pruebas que serán aplicados al Sistema de Información. Existen muchos tipos y niveles de pruebas, por lo cual es importante desde el inicio del proyecto que los líderes técnicos y funcionales de la dependencia definan los tipos de prueba que se ejecutaran a lo largo del ciclo de vida de desarrollo del Sistema de Información teniendo en cuenta las consideraciones particulares de cada Sistema de Información.

Con el objeto de generar entregables de manera ágil e incremental, es necesario que las funcionalidades desarrolladas se entreguen a los líderes funcionales y/o técnicos de la dependencia de acuerdo a las iteraciones definidas y de acuerdo con el Cronograma del Plan del Proyecto, a fin que se pueda abordar la mayor cantidad de pruebas y validar el correcto funcionamiento de los desarrollos vs requerimientos funcionales y no funcionales.

Se sugiere considerar el uso de pruebas automáticas para validar rápidamente que se mantenga la funcionalidad ya aprobada durante el ciclo de desarrollo del Sistema de Información.

La documentación relacionada con el proceso de pruebas, tales como, el plan maestro de pruebas, los casos de prueba, las fallas reportadas, los informes del resultado de las pruebas, y demás, deben ser registrados en la herramienta de Gestión de desarrollo de software, con el fin de dejar la trazabilidad de este proceso.

Los Sistemas de Información que sean entregados al DNP deben contemplar en el Plan Maestro de Pruebas con mínimo las siguientes tipos de pruebas:

### 6.1. Pruebas Funcionales

La aplicabilidad de las pruebas funcionales deben ser ejecutadas al inicio de cada ciclo de prueba de acuerdo a las entregas realizadas por el grupo de desarrollo.

Tipo de Prueba	Descripción de la Prueba	Producto a entregar
Pruebas Unitarias	<p>Las pruebas son elaboradas y ejecutadas por el perfil de desarrollo.</p> <p>Permite verificar la funcionalidad y estructura de cada componente individualmente del sistema una vez que ha sido codificado.</p> <p>Estas pruebas deben ser progresivas, ya que allí se pueden validar todos los métodos.</p>	Evidencia en la herramienta de Gestión de Desarrollo de Software del proyecto de pruebas unitarias debe ser cargado y ejecutado para evidenciar que las pruebas unitarias no presentan fallas.
Pruebas de Integración	<p>Las pruebas son elaboradas y ejecutadas por el grupo de desarrollo.</p> <p>Permite verificar el correcto ensamblaje entre los distintos módulos que componen el sistema desarrollado.</p>	Evidencia en la herramienta de Gestión de Desarrollo de Software o Informe de ejecución de las pruebas de integración en la carpeta "Documentación" subcarpeta Integración y Pruebas"

Pruebas de regresión	<p>Las pruebas son elaboradas y ejecutadas por el líder de pruebas.</p> <p>Las pruebas de regresión se realizarán durante las iteraciones de desarrollo y estabilización final para asegurar que la funcionalidad inicial este intacta junto con las nuevas características implementadas.</p> <p>Estas pruebas consisten en repetir las pruebas que ya fueron aplicados en las versiones o ciclos de prueba anteriores del sistema.</p>	<p>Evidencia en la herramienta de Gestión de Desarrollo de Software o Informe de ejecución de las pruebas de regresión en la carpeta “Documentación” subcarpeta Integración y Pruebas”</p>
Interfaz de usuario	<p>Las pruebas son elaboradas y ejecutadas por el líder de pruebas del proyecto.</p> <p>Permite verificar que la navegación a través de los elementos que se están probando reflejen las funciones de negocio.</p>	<p>Evidencia en la herramienta de Gestión de Desarrollo de Software o Informe de ejecución de las pruebas interfaz de usuario en la carpeta “Documentación” subcarpeta Integración y Pruebas”</p>
Pruebas de Aceptación de usuario	<p>Las pruebas son elaboradas y ejecutadas por el líder funcional del proyecto.</p> <p>Las pruebas de aceptación van dirigidas a validar que el sistema cumple los requisitos de funcionamiento esperado, de acuerdo a lo establecido en los requerimientos funcionales y en los criterios de aceptación del sistema de información, y conseguir la aceptación final del sistema por parte del usuario.</p> <p>Estas pruebas buscan diferencias entre la solución desarrollada y los requerimientos, con el fin de identificar fallas que se puedan generar entre la especificación funcional y el diseño del sistema</p>	<p>Evidencia en la herramienta de Gestión de Desarrollo de Software o Informe de ejecución de las pruebas de aceptación en la carpeta “Documentación” subcarpeta Integración y Pruebas”</p>

## 6.2 Pruebas No Funcionales

La Este tipo de pruebas se concentran en validar aspectos como: usabilidad, confiabilidad, desempeño, mantenibilidad etc. A continuación se describe este tipo de pruebas las cuales se aplican de acuerdo con las particularidades específicas de cada Sistema de Información, así mismo, pueden incluirse otros requerimientos no funcionales complementarios o atributos de calidad, estos atributos son los que se definan en el documento de [requerimientos no funcionales](#) realizado en la etapa previa al diseño de arquitectura..

Este tipo de pruebas se deben programar en el Cronograma del Plan del Proyecto, de tal manera que sean realizadas al menos por cada entrega de funcionalidades completas, se recomienda que este tipo de pruebas se implemente y apliquen desde el área de desarrollo, con el fin encontrar la menor cantidad de fallas en los demás ambientes.

Los Sistemas de Información que sean entregados al DNP deben tener y/o entregar el Plan de Pruebas con mínimo las siguientes pruebas no funcionales:

Tipo de Prueba	Descripción de la Prueba	Producto a entregar
Requerimientos de accesibilidad y usabilidad.	<p>Las pruebas son elaboradas y ejecutadas por el líder técnico del proyecto.</p> <p>El sistema debe cumplir con la guías de usabilidad y accesibilidad de la política de Gobierno Digital</p>	Evidencia en la herramienta de Gestión de Desarrollo de Software o Informe de ejecución de las pruebas de accesibilidad y usabilidad en la carpeta “Documentación” subcarpeta Integración y Pruebas”
Requerimientos de desempeño	<p>Las pruebas son elaboradas y ejecutadas por el líder de pruebas del proyecto junto con el líder técnico y el arquitecto de aplicaciones.</p> <p>El sistema debe soportar el número de usuarios conectados simultáneamente que determine el área funcional, sin degradar su rendimiento.</p> <p><b>Pruebas de Rendimiento</b> Se realiza generalmente para medir la rapidez con la cual un sistema ejecuta una determinada función</p> <p><b>Pruebas de Carga</b> Se realiza generalmente para medir el comportamiento del Sistema de Información o de un componente de software bajo el incremento de factores de exigencia.</p> <p><b>Pruebas de Estrés</b> Este tipo de pruebas de rendimiento está orientado a enfrentar los programas a condiciones excepcionales, extremas y desfavorables para demostrar que no continúan dando el rendimiento consistente o satisfactorio que se espera.</p>	Evidencia en la herramienta de Gestión de Desarrollo de Software e Informe de ejecución de las pruebas de carga y rendimiento en la carpeta “Documentación” subcarpeta Integración y Pruebas””. el proyecto de pruebas de carga debe ser cargado y ejecutado en servidores de DNP.
Cobertura de código	Las pruebas son elaboradas y ejecutadas por el líder de pruebas y el líder técnico del proyecto.	Evidencia en la herramienta de Gestión de Desarrollo de Software o Informe de ejecución de las

	<b>GUIA PLAN MAESTRO DE PRUEBAS</b>	CÓDIGO: GUIA - PLAN MAESTRO DE PRUEBAS
		PÁGINA: 8 de 13

	Este tipo de pruebas busca mejorar la calidad de software, ayuda a detectar de forma temprana código duplicado, código muerto, malas prácticas.	pruebas de cobertura en la carpeta “Documentación” subcarpeta Integración y Pruebas”
Requerimientos de escalabilidad	La arquitectura propuesta debe separar las capas lógicas de presentación, negocio y bases de datos permitiendo que sean desplegadas físicamente en diferentes servidores.	Evidencia en la herramienta de Gestión de Desarrollo de Software o Informe de ejecución de las prueba en la carpeta “Documentación” subcarpeta Integración y Pruebas”.
Requerimientos de seguridad	Las pruebas son elaboradas y ejecutadas por el líder de pruebas del proyecto y el agente de seguridad.  Al Sistema de Información le serán aplicadas inspecciones o escaneos a fin de verificar o detectar vulnerabilidades y/o debilidades que impliquen riesgos de seguridad, tales como inyección de código, seguridad de sitio web, La autenticación, los roles autorizados, la auditoria, datos encriptados, entre otras.	Evidencia en la herramienta de Gestión de Desarrollo de Software o Informe de ejecución de las prueba de seguridad en la carpeta “Documentación” subcarpeta Integración y Pruebas”
Requerimientos de vulnerabilidad	Las pruebas son elaboradas y ejecutadas por el líder de pruebas del proyecto y el agente de seguridad.  Se aplican para identificar y remediar posibles vectores de ataque y debilidades tecnológicas en redes, infraestructura, aplicaciones y otras áreas de TI.	Evidencia en la herramienta de Gestión de Desarrollo de Software o Informe de ejecución de las prueba de vulnerabilidad en la carpeta “Documentación” subcarpeta Integración y Pruebas”

## 7. RECURSO HUMANO Y CRONOGRAMA

En este ítem se especifican los recursos requeridos para la ejecución del proceso de pruebas. Cada Sistema de información debe contar con el recurso humano para la creación y ejecución de los casos de prueba tanto funcionales como no funcionales.

Es importante que, para desarrollo con metodologías ágil, el grupo de usuarios funcionales tenga conocimiento de la creación y ejecución de los casos de prueba.

La siguiente tabla muestra los recursos necesarios para desarrollar la disciplina de pruebas:



Rol	Responsabilidades Específicas
Líder funcional y/o líder técnico y/o supervisor del contrato	<ul style="list-style-type: none"> <li>Identificar y definir las pruebas funcionales y no funcionales necesarias para aplicar en cada entrega de software para pruebas.</li> <li>Estimar esfuerzo y organizar las tareas, identificar los riesgos de las pruebas.</li> <li>Efectuar seguimiento, control y análisis a los resultados y progreso de las pruebas en cada ciclo de pruebas.</li> <li>Revisar los casos de pruebas detallados.</li> <li>Ejecutar los casos de prueba de aceptación de usuario.</li> <li>Revisar los errores registrados en la herramienta de seguimiento de errores y supervisar su estado.</li> <li>Generar informes periódicos del estado de las pruebas.</li> <li>Resolver los problemas que bloqueen o atrasen las pruebas.</li> <li>Hacer un seguimiento de la programación de las pruebas.</li> </ul>
Líder de pruebas y/o líder técnico	<ul style="list-style-type: none"> <li>Definir las condiciones de prueba y los casos de prueba.</li> <li>Asegurar la correcta ejecución de la estrategia de pruebas.</li> <li>Crear y ejecutar las pruebas y registrar los resultados de dichas pruebas.</li> <li>Revisar los casos de pruebas detallados.</li> <li>Documentar los problemas encontrados durante la implementación</li> <li>Realizar la prueba de aceptación de la versión.</li> <li>Realizar las pruebas de regresión y de integración.</li> <li>Informar de los errores en la herramienta de gestión de desarrollo de software.</li> <li>Volver a probar los errores corregidos.</li> <li>Generar informes periódicos del estado de las pruebas y analizar los resultados.</li> </ul>

## 8. PLANIFICACION DEL PLAN MAESTRO DE PRUEBAS

Dentro del **Cronograma del Plan General del Proyecto**, se deben incluir los ítems relacionados con la ejecución de las pruebas funcionales y/o no funcionales.

Todos los Sistemas de Información deben contemplar el siguiente ciclo de pruebas en sus entregas parciales y/o totales:

	<b>GUIA PLAN MAESTRO DE PRUEBAS</b>	CÓDIGO: GUIA - PLAN MAESTRO DE PRUEBAS
		PÁGINA: 10 de 13

- **Planificación de las pruebas:** ¿qué se va a probar?, ¿cómo se va a probar?, ¿quién lo va a probar? y ¿cuándo?
- **Análisis y diseño de las pruebas:** En esta fase se analizan los requerimientos y se diseñan los casos de prueba, incluyendo en esta actividad casos de prueba con diferentes escenarios.
- **Ejecución de las pruebas:** En esta fase se ejecutan los casos de prueba con el fin de validar que los requerimientos especificados se hayan implementado de la manera correcta.
- **Reporte de las fallas y recomendaciones:** En esta fase se detectan las fallas y se categorizan de acuerdo a su gravedad, una vez son solucionadas se vuelve a revisar para validar la solución a lo reportado.

Para lograr el propósito de las pruebas, es necesario:

- Establecer la lista de funcionalidades que se van a probar en cada entrega de software.
- Detallar los recursos requeridos para la ejecución de las pruebas en cada entrega de software.
- Establecer los tiempos de los ciclos de prueba.
- Diseñar, crear y ejecutar los casos de prueba.
- Gestionar las fallas.
- Hacer seguimiento a las fallas reportadas y gestionar los nuevos casos de prueba de acuerdo a los desarrollos relacionados para la siguiente iteración.

Este ciclo es iterativo hasta que se termine el proceso de desarrollo y se entreguen todas las funcionalidades, es allí cuando se programa el cierre de las pruebas.

- **Cierre de las pruebas:** Es la última fase del proceso de pruebas, se archivan toda la documentación generada tanto de casos de prueba, como errores, se realiza un acta de aceptación de cierre del proceso de pruebas, que debe ser firmada por el grupo funcional y técnico de la dependencia, y por el ingeniero de pruebas y arquitecto de aplicaciones de la OTSI-GGSI.

## 9. ALCANCE DEL PLAN MAESTRO DE PRUEBAS

Se deben planear y ejecutar tanto las pruebas funcionales como las no funcionales, y de acuerdo con el tiempo definido en el Cronograma del Plan General del Proyecto.

- En cada despliegue de versión se deben diseñar y ejecutar los casos de prueba que cubran el 100% de los ciclos de caminos básicos válidos, se pueden elaborar diferentes casos de prueba, bien sea para validar pasos para los caminos alternos y generar otros para escenarios de falla.
- Las pruebas unitarias e integrales deben incluir al menos 80% de las funcionalidades desarrolladas.
- Las pruebas de dicha iteración, Se aprobarán con un 100% de las pruebas ejecutadas, pero con un 90% de aceptación. Esto quiere decir el 90% de las pruebas deben ser exitosas y sin fallas. En el restante 10% pueden existir fallas leves. Si se presentan fallas graves se deben suspender las pruebas y esperar a que sea solucionado la falla.

	<b>GUIA PLAN MAESTRO DE PRUEBAS</b>	CÓDIGO: GUIA - PLAN MAESTRO DE PRUEBAS
		PÁGINA: 11 de 13

## 10. PROCESO GENERAL DE PRUEBAS, CRITERIOS DE INICIO Y SUSPENSIÓN DE PRUEBAS

A continuación, se nombran algunos hitos importantes para cuando se realicen despliegues en los ambientes de pruebas de acuerdo a las iteraciones parciales que vayan siendo entregadas por el grupo de desarrollo.

El Plan de Pruebas puede aplicarse a todo el Proyecto de Desarrollo de Software, o puede acotarse a una iteración o conjunto de casos, de acuerdo a la metodología que se pacte, y de acuerdo al plan de desarrollo del proyecto definido en el **Plan de Proyecto de TI**.

Una vez están los casos de prueba definidos y se entreguen una versión para pruebas funcionales, es importante definir de acuerdo al resultado de las pruebas, si estas se aceptan o se suspenden, es decir definir unos criterios de aceptación o de suspensión de las pruebas de acuerdo con los resultados del ciclo de pruebas:

- **Pruebas Superadas:** Todos los procesos se ejecutan sin errores, los procesos finalizan en un tiempo aceptable, las pruebas de carga demuestran que existe un grado satisfactorio de capacidad y se puede dar por aceptada la versión para ser liberada a producción cuando sea necesario.
- **Pruebas Suspendidas:** Cuando se detectan fallas graves que impiden que se siga ejecutando las pruebas, por ejemplo, No se cuenta con un ambiente de pruebas, no disponibilidad de la base de datos, La versión entregada no supera la validación general, ya que se presenta muchas fallas.

Una vez se supera las fallas (**fallas graves**), se reanuda la ejecución de las pruebas, ya sea porque se corrige la falla reportada, o los equipos de desarrollo y de pruebas acuerdan que no es necesario corregir la falla inmediatamente y que podría solucionarse durante la siguiente iteración (**fallas leves**).

- **Fallas graves:** La información crítica es presentada erróneamente, información mal registrada en la base de datos, El error bloquea la generación o la realización de pruebas de una funcionalidad, incumplimiento de objetivos en funciones principales, este tipo de falla debe ser corregida inmediatamente.
- **Fallas leves:** Fallas en presentación de datos secundarios, generalmente son fallas de presentación.

## 11. AMBIENTES REQUERIDOS PARA PRUEBAS

Los ambientes de desarrollo – pruebas y producción debe estar por separado, es recomendable que el ambiente de pruebas sea lo más similar a producción, sin embargo, no siempre es posible debido a limitaciones financieras, pero debe estudiarse los requisitos mínimos de confiabilidad de estas pruebas respecto al entorno productivo.

En lo posible los proyectos deben implementar la metodología DEVOPS para el desarrollo de los Sistemas de Información, dado que permite tener la trazabilidad de los requerimientos, sus casos de pruebas, sus fallas, y hacer que los despliegues a producción sean más eficientes.

Las herramientas que se utilizaran para el proceso de las pruebas son las definidas dentro de los lineamientos de la Oficina de Informática y Sistemas de Información, estas son:

- DEVOPS la cual incluye seguimiento de errores, creación y seguimiento de casos de prueba, e informes de resultado del proceso
- Repositorio del proyecto en el DNP el cual debe contener los entregables definidos para las pruebas funcionales y no funcionales.
- Si el ciclo de pruebas no se genera en el repositorio de Gestión de Desarrollo de Software, se deben dejar toda la documentación y el resultado de las pruebas en la carpeta “Documentación subcarpeta Integración y Pruebas”.

## 12. RIESGOS

Los siguientes son los posibles riesgos que pueden afectar la ejecución exitosa del plan de prueba:

Riesgo	Estrategia de Mitigación
Agotar el tiempo de pruebas estimado para el proyecto, sin que se haya podido realizar parcial o totalmente el proceso de pruebas.	Tener los requerimientos aprobados, claros, objetivos y de fácil entendimiento.  Programación de fechas de entrega intermedias y final de producto. Indicando en cada iteración la funcionalidad incluida.
Incumplir alguno de los aspectos técnicos (software, plataforma)	Habilitar Ambiente de prueba con características similares al ambiente productivo.
Disponibilidad del personal para realizar la creación y ejecución de las pruebas.	Disponibilidad de personal con conocimientos especializados en alguna herramienta.  Capacitación de los usuarios en las herramientas de pruebas (funcional y técnica, pruebas manuales y automáticas)
Retrasos en el desarrollo del software	Verificar los motivos y verificar con el grupo funcional las alternativas y actualización de cronograma de proyecto.
Rotación de personal del área funcional de la dependencia	Capacitación de los nuevos usuarios en las funcionalidades solicitadas y en las herramientas de pruebas.

## 13. FINALIZACION DE LAS PRUEBAS

	<b>GUIA PLAN MAESTRO DE PRUEBAS</b>	CÓDIGO: GUIA - PLAN MAESTRO DE PRUEBAS
		PÁGINA: 13 de 13

El proceso de pruebas finaliza cuando están implementados todos requerimientos y se encuentran el 100 % de las pruebas ejecutadas y pasadas satisfactoriamente, no deben existir errores críticos.

Una vez finalizado el proceso de pruebas, se realiza un acta de aceptación por parte del líder funcional de la dependencia en donde se indica que el aplicativo cumple con lo establecido en los requerimientos funcionales y no funcionales.

- Es la última fase del proceso de pruebas, se archiva toda la documentación generada tanto de casos de prueba, como errores, se realiza un acta de aceptación de cierre del proceso de pruebas, que debe ser firmada por el grupo funcional y técnico de la dependencia, por el ingeniero de pruebas y arquitecto de aplicaciones de la OTSI-GGSI.

#### 14. REFERENCIAS

<https://mintic.gov.co/arquitecturati/630/w3-article-8841.html>