

DIRECCIÓN DE REGULACIÓN, PLANEACIÓN, ESTANDARIZACIÓN Y NORMALIZACIÓN (DIRPEN)

GUÍA CON RECOMENDACIONES PARA LA ELABORACIÓN DE ESPECIFICACIONES DE REQUERIMIENTOS

COORDINACIÓN DE REGULACIÓN ESTADÍSTICA

Diciembre de 2024

CONTENIDO

Introducción	3
1. Objetivo y alcance	4
Objetivo	4
Alcance	4
2. Conceptos básicos	5
3. Importancia de las especificaciones de requerimientos	7
4. Aspectos fundamentales	9
4.1. Tipos de requerimientos	9
4.2. Características de los requerimientos	17
4.3. Tipos de técnicas para la toma de requerimientos	19
5. Buenas prácticas y recomendaciones	23
6. Actividades para elaborar las especificaciones de requerimientos	32
Bibliografía	38

Introducción

El Departamento Administrativo Nacional de Estadística (DANE), como autoridad estadística nacional según la Ley 2335 de 2023, ejerce la regulación en materia estadística y, entre sus funciones, orienta, coordina y regula la producción de estadísticas oficiales en el marco del Sistema Estadístico Nacional (SEN). En línea con este rol, el DANE expidió los Lineamientos para el Proceso Estadístico en el SEN (versión 2.0) (DANE, 2020), basados en el Modelo Genérico del Proceso Estadístico (GSBPM) de la Comisión Económica de las Naciones Unidas para Europa (UNECE), versión 5.1.

Estos lineamientos proporcionan directrices claras para la generación de una operación estadística, incluyendo la necesidad de diseñar especificaciones de requerimientos para el desarrollo de sistemas, aplicativos o software destinados a la producción de información estadística. Dichas especificaciones son esenciales para prevenir errores en el desarrollo, proporcionando una descripción detallada de la lógica, los datos y los criterios de prueba necesarios para que el equipo de desarrollo cuente con la información suficiente y pueda garantizar la calidad del sistema mediante pruebas técnicas (Lozano, 2016). Estos documentos cubren, entre otros aspectos, los requisitos para sistemas de información, consulta, captura o transferencia de datos, validación y consistencia, y plataformas de difusión, que se utilizan en diversas operaciones estadísticas.

La presente guía tiene como objetivo orientar y establecer recomendaciones para la correcta elaboración del documento de especificaciones de requerimientos. Se estructura en seis partes: la primera expone el objetivo y alcance del documento; la segunda presenta los conceptos básicos relacionados con la guía; en la tercera se aborda la importancia de las especificaciones en las operaciones estadísticas; la cuarta parte expone los aspectos clave para su comprensión; la quinta parte ofrece recomendaciones, buenas prácticas, y en la parte final se relacionan las actividades necesarias para su elaboración.

1. Objetivo y alcance

Objetivo

Proporcionar recomendaciones y buenas prácticas estandarizadas para la elaboración de las especificaciones de los requerimientos que se desarrollan en los sistemas, los aplicativos o los software utilizados en las operaciones estadísticas producidas por las entidades del SEN y así facilitar el desarrollo de las fases del proceso estadístico y contribuir con la calidad de la información estadística.

Alcance

Este documento es aplicable a la elaboración de las especificaciones de requerimientos demandadas por los desarrolladores que diseñan y construyen los sistemas, los aplicativos o los software requeridos por parte de los responsables de la operación estadística en las fases de recolección o acopio, procesamiento, análisis y difusión de las operaciones estadísticas, tanto de fuentes primarias como secundarias, producidas por las entidades que conforman el SEN.

2. Conceptos básicos

Análisis (proceso estadístico): fase del proceso estadístico en la que se examina la consistencia y coherencia de la información consolidada y se generan los productos definidos en el diseño (DANE, 2023).

Análisis de requerimientos: el análisis de requerimientos es la fase inicial de todo proceso de software, la cual está conformada por tareas que llevan a determinar las necesidades o condiciones a cumplir en un producto nuevo o de un ajuste de uno existente (DNP, 2020).

Aplicativo móvil: programa que se puede descargar y al que se accede directamente desde un dispositivo móvil (DANE, 2023).

Difusión: es la fase del proceso de producción estadística en la que se pone a disposición pública los resultados de las operaciones estadísticas. Ésta incluye las actividades relacionadas con la elaboración de la documentación técnica de soporte y las acciones para facilitar el acceso a esta información (DANE, 2023).

Información estadística: es el conjunto de resultados y la documentación que los soporta, los cuales se obtienen de las operaciones estadísticas y que describen o expresan características sobre un elemento, un fenómeno o un objeto de estudio (DANE, 2023).

Metadatos: es la información necesaria para el uso y la interpretación de las estadísticas. Los metadatos describen la conceptualización, la calidad, la generación, el cálculo y las características de un conjunto de datos estadísticos (DANE, 2023).

Microdatos: corresponde a los datos sobre las características asociadas a las unidades de observación que se encuentran consolidadas en una base de datos (DANE, 2023).

Operación estadística: es la aplicación del conjunto de procesos y actividades que comprende la identificación de necesidades, diseño, construcción, recolección o acopio, procesamiento, análisis, difusión y evaluación, la cual conduce a la producción de información estadística sobre un tema de interés nacional y territorial (DANE, 2023).

Recolección o acopio (proceso estadístico): fase del proceso estadístico en la se ejecutan todas las acciones planeadas, diseñadas y construidas, en las fases anteriores con el fin de obtener los datos que permitirá generar la información estadística que satisfaga las necesidades identificadas (DANE, 2023).

Procesamiento (proceso estadístico): fase del proceso estadístico en la que se consolidan, integran, procesan y depuran los datos, de acuerdo con lo establecido en el diseño (DANE, 2023).

Procesamiento de datos: tratamiento realizado sobre los datos a fin de obtener información de acuerdo con un conjunto de reglas dado (DANE, 2023).

Proceso estadístico: conjunto sistemático de actividades encaminadas a la producción de estadísticas, entre las cuales están comprendidas: la detección de necesidades de información, el diseño, la construcción, la recolección, el procesamiento, el análisis, la difusión y la evaluación (DANE, 2023).

Sistema Estadístico Nacional (SEN): es el conjunto articulado de componentes que garantizan la producción y la difusión de las estadísticas oficiales a nivel nacional y territorial que requiere el país, de manera organizada y sistemática. Sus componentes son las entidades y las organizaciones productoras de información estadística y responsables de registros administrativos, los usuarios, los procesos y los instrumentos técnicos para la coordinación, así como las políticas, los principios, las fuentes de información, la infraestructura tecnológica y el talento humano necesario para su funcionamiento (DANE, 2023).

Software ofimático: Comprende cualquier programa informático o paquete de aplicaciones que sirven para perfeccionar, optimizar, mejorar, crear, almacenar, manipular y transmitir digitalmente la información necesaria en una oficina y realizar tareas y lograr objetivos básicos (DANE, 2023).

3. Importancia de las especificaciones de requerimientos

La especificación de requerimientos es una pieza clave en el éxito de cualquier proyecto de desarrollo de sistemas, aplicaciones o software. Su importancia radica en su capacidad para garantizar que los productos finales cumplan con las expectativas y las necesidades de los usuarios, además de ser el punto de partida para la planificación y la ejecución de las siguientes fases del ciclo de vida de un proyecto. En este sentido, una especificación clara y bien definida es esencial para la alineación de todos los involucrados en la operación estadística, desde los ingenieros de sistemas hasta los equipos responsables de la temática o el contenido que el sistema debe gestionar (Sommerville, 2016; DNP, 2020).

Una de las principales razones por las que las especificaciones de requerimientos son tan críticas es porque permiten identificar posibles brechas en los requisitos en las primeras etapas del proyecto de desarrollo de sistemas, aplicaciones o software. Este proceso de revisión y ajuste temprano evita que se descubran deficiencias cuando el proyecto está en fases más avanzadas, donde corregir errores tiende a ser mucho más costoso tanto en términos de tiempo como de recursos. De hecho, modificar los requisitos en fases avanzadas puede incrementar los costos significativamente, dado que implica revisar partes sustanciales del desarrollo, ajustar funcionalidades y, en muchos casos, realizar retrabajos innecesarios (Lozano, 2016; Northware. 2024).

Además, una especificación de requerimientos bien elaborada sirve como un punto de referencia central durante todo el desarrollo del sistema. Este documento proporciona una guía clara y común sobre qué se espera del sistema. Sin este punto de referencia, es común que surjan malentendidos o interpretaciones erróneas acerca de las funcionalidades que debe tener el sistema, lo que puede derivar en entregables que no cumplen con las expectativas del usuario o que no son técnicamente viables.

La claridad y la especificidad en la definición de los requerimientos también son aspectos cruciales para evitar sobrecostos y retrasos en el proyecto. Un requerimiento ambiguo puede llevar a que los desarrolladores interpreten de manera incorrecta lo que se debe construir, resultando en un sistema que no cumple con las necesidades del usuario o que requiere ajustes continuos, lo cual afecta tanto al cronograma como al presupuesto previsto. Por ejemplo, la falta de claridad en las historias de usuario, una herramienta común en el desarrollo ágil de software puede provocar que se construyan funcionalidades innecesarias o inadecuadas, lo que genera más trabajo y eleva los costos del proyecto (Wong, 2017).

Además de los costos directos asociados con la mala definición de los requerimientos, existen otros riesgos importantes que deben considerarse. Los sistemas que no cumplen con las expectativas del usuario pueden requerir no solo ajustes a nivel técnico, sino también cambios en las políticas o los procesos de negocio, lo que puede ser igualmente costoso e incluso provocar el fracaso del proyecto.

De hecho, la mala gestión de los requerimientos es una de las causas más comunes de fallos en proyectos de desarrollo de software, lo que ratifica la importancia de documentar y gestionar los requisitos de manera adecuada (QbD Group. 2024; Wong, 2017).

La especificación de requerimientos también actúa como un mecanismo para asegurar que el sistema cumpla con cualquier estándar, contrato o normativa relevante que haya sido impuesta. En muchos proyectos, especialmente aquellos relacionados con la administración pública o los sectores regulados, es fundamental que el sistema cumpla con ciertos criterios de calidad, seguridad y funcionamiento. Documentar los requerimientos en detalle asegura que se tomen en cuenta estas consideraciones desde las fases iniciales del proyecto y así evitar incumplimientos durante su implementación o funcionamiento (Sommerville, 2016; Lozano, 2016; Wong, 2017).

Finalmente, no debe subestimarse el valor de la especificación de requerimientos como una herramienta de comunicación. Dado que los proyectos de desarrollo de sistemas suelen involucrar a equipos multidisciplinarios (por ejemplo, expertos en el dominio del contenido, ingenieros de software, diseñadores de experiencia de usuario y gerentes de proyecto) y contar con un documento formal que exponga claramente y con precisión lo que se espera del sistema, es crucial que todas las partes tengan un entendimiento común de los objetivos del proyecto. Esto no solo previene malentendidos, sino que también facilita la colaboración efectiva entre equipos y asegura que las decisiones tomadas durante todo el proceso de desarrollo estén alineadas con los objetivos del proyecto y las necesidades del usuario (Northware, 2024; DNP, 2020).

4. Aspectos fundamentales

Los requerimientos son aquellas condiciones o capacidades que posee un sistema, software o aplicativo para resolver un problema o satisfacer una necesidad solicitada por un usuario. En el desarrollo de sistemas de información, la correcta definición de los tipos de requerimientos es crucial para asegurar que el sistema no solo cumpla con las funcionalidades esperadas, sino que también ofrezca un rendimiento adecuado y sea pueda mantener a lo largo del tiempo. A continuación, se describen los tipos de requerimientos.

4.1. Tipos de requerimientos¹

Los requerimientos se clasifican en dos categorías principales: requerimientos funcionales y requerimientos no funcionales. Además, existen otros requerimientos específicos que abordan necesidades particulares del proyecto, que se detallan más adelante.

Requerimientos funcionales: describen las acciones y las operaciones que el sistema debe realizar. Estos requerimientos definen las funciones que el sistema debe ejecutar para satisfacer las necesidades de los usuarios y describir lo que el sistema debe hacer. Los requerimientos funcionales incluyen interacciones entre el usuario y el sistema, así como entre el sistema y otros sistemas externos. Además, estos requerimientos son la base para el diseño y el desarrollo del sistema y proporcionan una visión clara y detallada de las funcionalidades.

Un buen requerimiento funcional detalla:

- Los servicios que ofrecerá el sistema.
- La justificación de su inclusión.
- La persona que lo propuso.
- Los criterios de aceptación específicos².

Estos requerimientos deben ser coherentes y no contradictorios para que los desarrolladores puedan implementarlos efectivamente. A continuación, se presenta un ejemplo de un requerimiento funcional:

¹ Basado en: UNAD; UC; UGR; DNP, 2020; Sommerville, 2016; Lozano, 2016; Wong, 2017.

² Criterios de aceptación: condiciones específicas que deben cumplirse para que una historia de usuario se considere completada. Estos criterios aseguran que la funcionalidad entregada cumple con los requisitos del usuario. "Los criterios de aceptación proporcionan una forma objetiva de validar que una historia de usuario está terminada y cumple con los requisitos definidos, asegurando calidad y cumplimiento." (DNP, 2020)

Tabla 1. Ejemplo de un requerimiento funcional

Descripción: el aplicativo de recolección de datos debe permitir la captura y el almacenamiento de información.	
Detalle específico	Descripción
 Captura de datos	- El aplicativo debe permitir la entrada de datos a través de cuestionarios vía web. - Los cuestionarios deben adaptarse a diferentes tipos de datos, tales como numéricos, texto, fechas, entre otros. - Se debe validar los datos en tiempo real para asegurar la calidad y la consistencia de la información ingresada.
 Almacenamiento de datos	- Los datos recolectados deben ser almacenados en una base de datos segura y accesible. - El aplicativo debe permitir la exportación de datos en formato Excel. - Debe contar con mecanismos de respaldo y recuperación de datos, con el fin de prevenir la pérdida de información.

Fuente: DANE, Regulación estadística.

Este tipo de requerimientos es fundamental para garantizar que el sistema cumpla con las expectativas de los usuarios y que las funciones esenciales sean entregadas de forma precisa.

Requerimientos no funcionales: son aquellos que no están directamente relacionados con lo que el sistema hace, sino con las características de calidad que el sistema debe cumplir³. Estos incluyen restricciones operativas, como el tiempo de respuesta, la seguridad, la usabilidad, la escalabilidad y la mantenibilidad. Mientras que los requerimientos funcionales describen qué hace el sistema, los no funcionales describen cómo debe hacerlo, asegurando que el sistema sea funcional, eficiente, seguro y adaptable.

Estos requerimientos son esenciales para garantizar que el sistema cumpla con los estándares de calidad esperados y que sea capaz de manejar las cargas de trabajo y así garantizar la protección de los datos y ofrecer una experiencia de usuario óptima. Asimismo, deben incluirse consideraciones sobre la capacidad del sistema para evolucionar con el tiempo y permitir futuras actualizaciones y mejoras.

Ejemplo de un requerimiento no funcional:

Tabla 2. Ejemplo de un requerimiento no funcional

Requerimiento no funcional: Seguridad de la información Descripción: el aplicativo de recolección de datos debe garantizar que no se pierda la información recolectada, así como la confidencialidad y disponibilidad de la información.	
Detalle específico	Descripción
 Registro y monitoreo	• Debe mantener un registro (accesos, modificaciones y eliminaciones de datos) de todas las actividades de los usuarios definidos. • Los registros deben ser monitoreados constantemente para detectar y responder a actividades no autorizadas.
 Cumplimiento normativo	• El aplicativo debe cumplir con todas las normativas en materia de protección de datos y seguridad de la información (Ley 79 de 1993).
 Respaldo y recuperación	• Deben implementarse procedimientos de respaldo y recuperación de datos para asegurar la disponibilidad y la integridad de la información en caso de fallos o incidentes de seguridad.

Fuente: DANE, Regulación estadística.

Al cumplir con los requerimientos no funcionales se asegura que el sistema pueda ser utilizado de manera efectiva en un entorno real, incluso bajo condiciones de alta demanda.

Otros requerimientos: además de los requerimientos funcionales y no funcionales, existen otros tipos de requerimientos que son igualmente importantes para el desarrollo exitoso de un sistema. Estos requerimientos se centran en aspectos específicos como la interacción con los usuarios, el rendimiento, la seguridad y el cumplimiento de normativas.

- Los *requisitos del usuario* describen las necesidades y las expectativas que los usuarios tienen del sistema, expresadas en un lenguaje comprensible para ellos. A diferencia de los requerimientos funcionales, estos no se centran en aspectos técnicos, sino en lo que los usuarios finales esperan obtener del sistema. Los requisitos del usuario son cruciales para asegurar que el sistema sea útil y valioso para quienes lo utilizarán.

Ejemplos de requisitos del usuario:

Tabla 3. Ejemplo requisito de usuario

Aplicativo de recolección de datos debe permitir la captura y el almacenamiento de información Requisito del usuario: facilidad en la navegación. Descripción: debe proporcionar un diligenciamiento de la encuesta eficiente y permitir a los encuestados navegar y capturar datos sin dificultad.	
Detalle específico	Descripción
 Recuperación de contraseñas	El sistema debe permitir a los usuarios recuperar sus contraseñas a través de un proceso sencillo y seguro, utilizando su correo electrónico registrado.
 Notificaciones y alertas	Los usuarios deben recibir notificaciones y alertas cuando finalice el diligenciamiento del cuestionario.
 Interfaz amigable	Debe ser clara y fácil de entender, con menús y botones bien organizados y etiquetados.

Fuente: DANE, Regulación estadística.

Capturar correctamente los requisitos del usuario es esencial para diseñar un sistema que realmente cumpla con las expectativas y proporcione una buena experiencia.

- Los *requisitos del sistema* son especificaciones técnicas más detalladas que describen cómo el sistema debe implementar los requisitos del usuario. Estos están dirigidos a los desarrolladores y los ingenieros y suelen abordar aspectos como el diseño del software, las arquitecturas subyacentes y los mecanismos de comunicación entre componentes del sistema.

Ejemplo de un requisito del sistema:

Tabla 4. Ejemplo requisito del sistema

Aplicativo de recolección de datos debe permitir la captura y el almacenamiento de información Requisito del sistema: cifrado de contraseñas. Descripción: el aplicativo debe garantizar la seguridad de las contraseñas de los encuestados a través del uso de cifrado avanzado.	
Detalle específico	Descripción
 Almacenamiento de contraseñas	El sistema debe utilizar cifrado AES-256⁴ para almacenar las contraseñas de los usuarios de manera segura en la base de datos.

Fuente: DANE, Regulación estadística.

⁴ Cifrado AES-256 (Advanced Encryption Standard) o Rijndael es un sistema de cifrado usado en criptología.

Este tipo de requisitos aseguran que el sistema sea desarrollado de manera segura y eficiente, respetando las normas técnicas y garantizando su estabilidad.

- Los *requisitos de rendimiento* especifican cómo debe comportarse el sistema bajo ciertas condiciones, asegurando que sea capaz de manejar la carga esperada y que los tiempos de respuesta sean adecuados. Estos requisitos son críticos en sistemas que manejan grandes volúmenes de datos o altas demandas de usuarios simultáneos.

Ejemplo de un requisito de rendimiento:

Tabla 4. Ejemplo requisito de rendimiento

Aplicativo de recolección de datos debe permitir la captura y el almacenamiento de información Requisito de rendimiento: tiempo de respuesta. Descripción: el aplicativo debe procesar y almacenar la información de los cuestionarios de forma rápida y eficiente.	
Detalle específico	Descripción
 Tiempo de funcionamiento	- El tiempo de carga inicial del aplicativo no debe exceder los 3 segundos en condiciones de red estándar. - La captura y el almacenamiento de datos deben completarse en menos de 1 segundo en el 98% de los casos.

Fuente: DANE, Regulación estadística.

Este tipo de requerimientos es fundamental para garantizar que el sistema sea ágil y responda a las necesidades de los usuarios en tiempo real.

- Los *requisitos de seguridad* detallan las medidas necesarias para proteger el sistema y los datos que maneja contra amenazas y vulnerabilidades. Estos requisitos son especialmente importantes en sistemas que manejan información sensible o crítica.

Ejemplo:

Tabla 5. Ejemplo requisito de seguridad

Aplicativo de recolección de datos debe permitir la captura y el almacenamiento de información Requisito de seguridad: protección de datos sensibles. Descripción: el aplicativo debe garantizar la protección de los datos sensibles capturados y así asegurar la integridad y la confidencialidad de la información, evitando accesos no autorizados.	
Detalle específico	Descripción
 Autenticación y autorización	- El aplicativo debe requerir autenticación de todos los usuarios (fuentes, encuestadores y supervisores) mediante credenciales únicas (nombre de usuario y contraseña). - Los roles y los permisos deben ser definidos para asegurar que los usuarios solo puedan acceder a la información y las funcionalidades de acuerdo con su rol.

Fuente: DANE, Regulación estadística.

La seguridad es esencial para asegurar la integridad, la confidencialidad y la disponibilidad de los datos, especialmente en un entorno donde las amenazas a la ciberseguridad son constantes.

- Los *requisitos de usabilidad* especifican cuán fácil y efectiva es la interacción de los usuarios con el sistema. Estos requisitos son fundamentales para mejorar la experiencia del usuario y garantizar que el sistema sea intuitivo y eficiente en su uso.

Ejemplo de un requisito de usabilidad:

Tabla 6. Ejemplo requisito de usabilidad

Aplicativo de recolección de datos debe permitir la captura y el almacenamiento de información Requisito de usabilidad: facilidad de uso. Descripción: el aplicativo debe ser sencillo de usar para las fuentes y permitir la captura de datos de manera eficiente y sin errores.	
Detalle específico	Descripción
 Accesibilidad	El aplicativo tiene accesibilidad web⁵ para usuarios con discapacidades, cumpliendo con los estándares AA de la Guía de Accesibilidad de Contenidos

⁵ Accesibilidad web: consiste en lograr que las páginas web sean utilizables por el máximo número de personas, independientemente de sus conocimientos o sus capacidades personales, e independientemente de las características técnicas del equipo utilizado para acceder a la web. Conceptos estandarizados, DANE, https://conceptos.dane.gov.co/conceptos/conceptos_catalogo.

	Web (Web Content Accessibility Guidelines - WCAG) en la versión 2.1, expedida por el World Web Consortium (W3C) ⁶ .
--	--

Fuente: DANE, Regulación estadística.

Un sistema fácil de usar no solo mejora la satisfacción del usuario, sino que también aumenta la eficiencia y reduce la necesidad de capacitación intensiva.

- Los *requisitos de mantenimiento* especifican cómo se gestionarán las actualizaciones, las correcciones de errores y las mejoras del sistema una vez que este esté en operación. Estos requisitos aseguran que el sistema pueda evolucionar y adaptarse a nuevos requisitos o cambios en su entorno operativo.

Ejemplo:

Tabla 7. Ejemplo requisito de mantenimiento

Aplicativo de recolección de datos debe permitir la captura y el almacenamiento de información Requisito de mantenimiento: facilidad de mantenimiento. Descripción: El aplicativo debe ser fácilmente mantenible, permitiendo actualizaciones y correcciones de errores de manera eficiente, sin afectar su funcionamiento.	
Detalle específico	Descripción
 Documentación	• El aplicativo debe estar documentado de forma clara y detallada que expliquen la lógica de las funciones y los módulos.
 Modular	• Su diseño es de forma modular, para que se actualicen los componentes individuales puedan modificar sin afectar el resto del aplicativo.

Fuente: DANE, Regulación estadística.

Un buen plan de mantenimiento es esencial para prolongar la vida útil del sistema y garantizar que pueda evolucionar sin problemas.

- Los *requisitos legales y regulatorios* aseguran que el sistema cumpla con todas las leyes, las regulaciones y las normas aplicables en su ámbito de operación. Estos requisitos son críticos en sectores donde la privacidad y la protección de datos son fundamentales.

⁶ Resolución 1519 de 2020 del Ministerio de Tecnologías de la Información y las Comunicaciones, por la cual se definen los estándares y las directrices para publicar la información señalada en la Ley 1712 del 2014 y se definen los requisitos materia de acceso a la información pública, accesibilidad web, seguridad digital y datos abiertos.

Ejemplo de un requisito legal:

Tabla 8. Ejemplo requisito legal

Aplicativo de recolección de datos debe permitir la captura y el almacenamiento de información Requisito legales y regulatorios: protección de datos personales Descripción: el aplicativo debe cumplir con las normativas colombianas de protección de datos personales y garantizar la privacidad y la seguridad de la información de los usuarios, así como las guías y las recomendaciones para el buen manejo de los datos en Colombia (Ley 79 de 1993), por la cual se regula la realización de los censos de población y vivienda en todo el territorio nacional. Guía para el uso y el aprovechamiento de Datos Abiertos en Colombia del Ministerio de Tecnologías de la Información y las Comunicaciones.	
Detalle específico	Descripción
 Aviso de privacidad	El aplicativo debe incluir un aviso de privacidad claro informando a los encuestados sobre la recolección, el uso y el tratamiento de sus datos personales.

Fuente: DANE, Regulación estadística.

El cumplimiento normativo es crucial para evitar sanciones y asegurar la conformidad con las regulaciones aplicables.

- Los *requisitos de portabilidad* especifican la facilidad con la que el sistema puede ser transferido de un entorno operativo a otro. Esto es esencial en proyectos donde se espera que el sistema pueda operar en diferentes plataformas o entornos.

Ejemplo de un requisito de portabilidad:

Tabla 9. Ejemplo requisito de portabilidad

Aplicativo de recolección de datos debe permitir la captura y el almacenamiento de información Requisito portabilidad: transferencia de datos. Descripción: el aplicativo permite la transferencia de datos de forma eficiente y segura entre diferentes plataformas que se utiliza en la operación estadística y garantizar la integridad y la disponibilidad de la información.	
Detalle específico	Descripción
 Interoperabilidad	El aplicativo debe ser compatible con otras plataformas utilizadas por los responsables de la operación estadística para realizar las importaciones y las exportaciones de datos necesarios en formatos estándar.

Fuente: DANE, Regulación estadística.

La portabilidad asegura que el sistema pueda ser adaptado a nuevos contextos sin necesidad de un rediseño completo.

4.2. Características de los requerimientos⁷

Para definir y documentar adecuadamente las especificaciones de requerimientos, es esencial que estos cumplan con una serie de características. Estas características aseguran que los requerimientos sean útiles, comprensibles y factibles de implementar. A continuación, se describen las principales características de los requerimientos:

Completo: un requerimiento es completo cuando proporciona toda la información necesaria y suficiente para que sea entendido por todos los involucrados en el proyecto; no requieren detalles adicionales o aclaraciones y abarcan todos los aspectos relevantes para su implementación.

Comprobable: un requerimiento es comprobable cuando puede ser evaluado mediante pruebas objetivas. Esta característica está relacionada con la verificabilidad, pero pone énfasis en que se pueda ejecutar una prueba directa, ya sea a través de un análisis, inspección o demostración.

Consistente: los requerimientos no se contradicen entre sí ni con otros requerimientos del mismo sistema. La consistencia garantiza que todas las partes del sistema funcionen de manera coherente y que los requerimientos puedan integrarse sin conflicto.

Correcto: los requerimientos describen de manera precisa las funcionalidades o las características que el sistema debe ofrecer. Un requerimiento es correcto cuando refleja una necesidad real del usuario o del sistema y está alineado con los objetivos del proyecto para asegurar que se desarrolle el sistema deseado.

Estimable: un requerimiento se estima en términos de esfuerzo y tiempo necesarios para su implementación. Esto es crucial para la planificación del proyecto y la asignación de recursos.

Factible: un requerimiento es factible cuando puede implementarse dentro de las restricciones de tiempo, presupuesto y tecnología disponibles. Si un requerimiento no es factible, es necesario replantearlo o ajustar las expectativas del proyecto.

⁷ Estas características son la síntesis de las propuestas por: UNAD; DNP, 2020; Wong, 2017; QbD Group. 2024; Northware, 2024.

Independiente: los requerimientos son independientes para desarrollarse de manera autónoma. Esto permite mayor flexibilidad en el desarrollo y facilita el trabajo en paralelo de diferentes equipos.

Modificable: los requerimientos son modificables, cambiándose de forma sencilla y consistente cuando sea necesario. Para esto, es importante que cada requerimiento esté identificado de manera única, evitando duplicidades y asegurando que cualquier modificación se aplique de manera controlada.

Necesario: un requerimiento es necesario si, al omitirlo, el sistema sufre una deficiencia significativa en su funcionalidad o en la calidad del producto. Esto asegura que solo se incluyan requerimientos que aporten valor y sean imprescindibles para el sistema.

Negociable: los detalles de los requerimientos son flexibles y negociables entre las partes interesadas. Esto permite ajustar los requerimientos en función de las restricciones del proyecto o cambios en las prioridades.

No ambiguo: un requerimiento es no ambiguo cuando tiene una única interpretación posible. Todos los interesados deben llegar a la misma comprensión del requerimiento, lo que reduce los riesgos de malentendidos durante el desarrollo.

Pequeño: los requerimientos están desglosados en unidades de trabajo lo suficientemente pequeñas para ser gestionables. Esto facilita su implementación en ciclos cortos de desarrollo, como en *sprints*, y mejora el control del progreso.

Priorizado: los requerimientos se clasifican según su importancia y urgencia. Esta priorización ayuda a los equipos de desarrollo a enfocarse en los aspectos críticos del sistema y gestionar el tiempo de manera eficiente. Los requerimientos pueden ser clasificados como esenciales, deseados u opcionales.

Trazable: un requerimiento es trazable cuando su origen, su implementación y su relación con otros requerimientos o componentes del sistema son fácilmente identificables. La trazabilidad facilita el seguimiento de cada requerimiento a lo largo del desarrollo y garantiza que todos los aspectos del sistema estén alineados.

Valioso: todo requerimiento aporta un valor tangible al usuario final o al sistema. Si un requerimiento no genera un beneficio claro, es probable que no sea necesario para el éxito del proyecto.

Verificable: un requerimiento es verificable, lo que significa que debe poder comprobarse si ha sido cumplido o no. Esto puede hacerse a través de pruebas, análisis o demostraciones. Los criterios de aceptación son claros para poder validar que el requerimiento se ha implementado correctamente.

4.3. Tipos de técnicas para la toma de requerimientos⁸

La toma de requerimientos es un paso crucial en el desarrollo de sistemas, ya que asegura que las necesidades y expectativas de los usuarios y otras partes interesadas sean comprendidas y documentadas correctamente. Para lograr este objetivo, existen diversas técnicas que permiten capturar requerimientos de manera eficiente y asegurar que todos los aspectos relevantes sean considerados. A continuación, se describen algunas de las técnicas más comunes y útiles para la recolección de requerimientos:

La **entrevista** es una técnica clásica para la toma de requerimientos que consiste en conversaciones estructuradas⁹ o semi-estructuradas¹⁰ con los usuarios y otras partes interesadas del proyecto. Durante esta, se exploran las necesidades, las expectativas y los problemas específicos que los usuarios enfrentan con el sistema actual o las funcionalidades deseadas en un nuevo sistema.

Ventajas

- Permite una comprensión profunda de los requerimientos.
- Fomenta la interacción directa con los interesados.
- Posibilita la clarificación inmediata de dudas.

Limitaciones

- Es una técnica que puede ser costosa y consumir mucho tiempo.
- Los usuarios pueden no siempre ser capaces de articular claramente sus necesidades.

⁸ Northware, 2024; Wong, 2017.

⁹ Son entrevistas en las que todas las preguntas están predefinidas y se hacen en el mismo orden a todos los entrevistados.

¹⁰ Son entrevistas que combinan preguntas predefinidas con la flexibilidad de explorar temas emergentes durante la conversación.

Un **taller** de requerimientos es una sesión colaborativa donde se reúnen varios interesados para discutir, priorizar y validar los requerimientos del sistema. Estos talleres permiten la participación de diferentes roles¹¹ dentro del proyecto y facilitan el consenso entre equipos multidisciplinarios.

Ventajas

- Fomenta la colaboración y la toma de decisiones en grupo.
- Ayuda a resolver conflictos y priorizar requerimientos.
- Proporciona una visión integral del sistema desde diferentes perspectivas.

Limitaciones

- Puede ser complicado coordinar a todos los interesados para asistir al taller.
- La falta de moderación adecuada puede derivar en sesiones desorganizadas o improductivas.

La **observación** implica monitorear directamente el uso del sistema actual o el entorno de trabajo de los usuarios para identificar necesidades y problemas de primera mano. Esta técnica es útil cuando los usuarios tienen dificultades para expresar sus requerimientos o cuando las interacciones con el sistema no son completamente explícitas.

Ventajas

- Captura requerimientos que los usuarios pueden no ser conscientes de expresar.
- Ofrece una perspectiva objetiva de cómo se utiliza el sistema actual.

Limitaciones

- La observación puede ser invasiva y generar cambios en el comportamiento de los usuarios.
- Puede no ser efectiva en sistemas complejos donde las interacciones no son evidentes.

La **encuesta** es una técnica útil para recolectar datos de un gran número de usuarios o interesados de manera eficiente. Consisten en cuestionarios estructurados que pueden incluir preguntas abiertas o cerradas para obtener tanto información cualitativa como cuantitativa.

Ventajas

- Permite recopilar información de una gran cantidad de usuarios en poco tiempo.

¹¹ Roles: Actores que interactúan con el sistema y tienen interés en su desarrollo y uso. Estos pueden incluir usuarios finales, desarrolladores, testers, y responsables del diseño de la operación estadística. Por lo anterior, identificar y comprender los roles es crucial para redactar historias de usuario efectivas.

- Es una técnica económica y fácil de implementar.

Limitaciones

- Las respuestas pueden carecer de profundidad y contexto.
- No permite la interacción directa para aclarar dudas o explorar detalles.

El **análisis de documentación** implica revisar la información existente, como manuales, informes, contratos y especificaciones previas, para extraer requerimientos relevantes. Esta técnica es útil cuando se tiene acceso a documentos históricos o normativos que deben ser considerados en el desarrollo del sistema.

Ventajas

- Proporciona una base sólida de información preexistente.
- Asegura el cumplimiento de normativas y requisitos previos.

Limitaciones

- Los documentos pueden estar desactualizados o no reflejar completamente las necesidades actuales.
- Puede ser difícil interpretar algunos aspectos sin el contexto adecuado.

La creación de **prototipos** consiste en desarrollar versiones preliminares del sistema o de algunas de sus funcionalidades para que los usuarios puedan interactuar con ellos y brindar retroalimentación. Los prototipos permiten visualizar cómo funcionará el sistema final y ayudan a clarificar los requerimientos.

Ventajas

- Los usuarios pueden ver y experimentar el sistema antes de su desarrollo completo.
- Ayuda a identificar problemas y ajustar requerimientos de manera temprana.

Limitaciones

- Puede ser costoso y tomar tiempo desarrollar prototipos.
- Si no se gestiona bien, los usuarios pueden confundir el prototipo con el sistema final.

Las **historias de usuario** son descripciones breves y centradas en el usuario que capturan las funcionalidades específicas que el sistema debe cumplir. Las historias de usuario permiten comprender los requerimientos desde la perspectiva del usuario final y se utilizan comúnmente en enfoques ágiles.

Estos son los tipos de historias de usuario:

- *Épicas*: son historias grandes que representan una funcionalidad general o un grupo de funcionalidades. Generalmente, las épicas se dividen en historias más pequeñas para facilitar su desarrollo.
- *Historias de usuario*: son descripciones concisas de una funcionalidad específica que el sistema debe ofrecer desde el punto de vista del usuario.
- *Tareas*: son actividades más detalladas que deben completarse para implementar una historia de usuario. Las tareas suelen involucrar acciones técnicas específicas.

Ventajas

- Proporciona una visión clara desde la perspectiva del usuario.
- Facilita la priorización y la gestión de requerimientos en equipos ágiles.

Limitaciones

- Puede ser insuficiente para sistemas complejos si no se complementa con otros tipos de documentación.

El **brainstorming** es una técnica creativa utilizada en las primeras etapas de la toma de requerimientos para generar ideas sobre las posibles funcionalidades del sistema. Se trata de una sesión abierta donde los participantes proponen ideas sin restricciones, que luego son organizadas y evaluadas.

Ventajas

- Genera una amplia variedad de ideas y perspectivas.
- Fomenta la creatividad y la colaboración.

Limitaciones

- Puede generar muchas ideas no prácticas o irrelevantes.
Necesita moderación adecuada para evitar desviaciones del objetivo principal.

5. Buenas prácticas y recomendaciones¹²

Para garantizar el éxito en el desarrollo de sistemas, aplicativos o software que apoyen una operación estadística, en el contexto del SEN, es imprescindible comenzar con una correcta elaboración de las especificaciones de requerimientos. Este proceso asegura que las necesidades y las expectativas del equipo temático u operativo se traduzcan en instrucciones claras y precisas para los ingenieros encargados del diseño y el desarrollo. Las especificaciones actúan como un puente que une la visión conceptual de la operación estadística con la implementación técnica, facilitando una comunicación fluida entre los equipos y asegurando que el producto final sea funcional y alineado con los objetivos planteados.

El proceso de especificación no es simplemente una lista de funcionalidades, dado que es un análisis detallado y exhaustivo que comienza con la identificación de las necesidades reales y continúa con la formulación de requerimientos específicos. Estas actividades no solo ayudan a evitar malentendidos y contradicciones durante el desarrollo, sino que también previenen defectos costosos que, de no ser abordados adecuadamente, pueden surgir más adelante. A través de una conceptualización precisa y un análisis meticuloso, el proceso de elaboración de requerimientos asegura que el sistema, aplicativo o software cumpla con las expectativas tanto técnicas como funcionales, minimizando los riesgos de error y optimizando la eficiencia en su desarrollo y uso dentro de la operación estadística.

A continuación, se presentan las recomendaciones y las buenas prácticas clave para la elaboración de especificaciones de requerimientos en una operación estadística, asegurando una comunicación clara y efectiva entre los responsables temáticos, operativos y los ingenieros de desarrollo.

Asegurar la participación continua de todos los interesados, incluyendo los equipos temáticos, operativos y de sistemas, durante todo el proceso de diseño del sistema, aplicación o software, para garantizar que sus necesidades y expectativas sean correctamente atendidas.

- Para asegurar que las expectativas de los usuarios finales se reflejen en las especificaciones, es fundamental organizar entrevistas y talleres que permitan identificar sus necesidades reales.
- La validación de los requerimientos puede realizarse mediante prototipos o simulaciones, lo que facilita la identificación temprana de ajustes antes de comenzar el desarrollo completo.

¹² Las siguientes recomendaciones y buenas prácticas surgen de experiencias manifiestas en la documentación del diseño de sistemas, aplicaciones y software dentro de las operaciones estadísticas en el SEN y de la siguiente bibliografía: UNAD; UC; UGR; DNP, 2020; ANT; Sommerville, 2016; Lozano, 2016; Wong, 2017; Northware.

- Durante el desarrollo, es importante incorporar la retroalimentación continua de los usuarios finales, manteniendo un registro de las modificaciones sugeridas y asegurando que se aborden en futuras iteraciones.
- La retroalimentación debería ser gestionada mediante revisiones regulares, fomentando la comunicación abierta entre los equipos temáticos, operativos y de sistemas para resolver dudas y ajustar los requerimientos en tiempo real.
- Cuando se detecte que el usuario no tiene claridad total sobre sus necesidades, proporcione un caso de éxito o un ejemplo concreto cercano al área temática del estudio estadístico. Esto ayudará a que el usuario visualice de manera práctica cómo podría funcionar la solución, reduciendo la ambigüedad y facilitando la toma de decisiones.

Utilizar un lenguaje natural, claro y preciso, complementado con material visual, para asegurar que las especificaciones sean fácilmente comprensibles y evitar malinterpretaciones entre los desarrolladores y otros interesados.

- Se recomienda redactar los requerimientos en un lenguaje comprensible para usuarios no técnicos, evitando el uso de jerga informática, términos especializados, y dobles o triples negaciones que puedan generar confusión.
- Es fundamental evitar términos ambiguos, sinónimos, homónimos y combinaciones como "X/Y", que podrían interpretarse de distintas maneras.
- Los encargados del diseño deberían proporcionar siempre valores medibles y claros en las especificaciones y realizar revisiones cruzadas formales entre los miembros del equipo para identificar y corregir posibles malentendidos o áreas ambiguas, garantizando que los desarrolladores no tengan que hacer suposiciones.
- Es recomendable organizar los requerimientos de manera jerárquica y numerada para facilitar su gestión y comprensión.
- En cuanto a los requerimientos no funcionales, estos deberían estar redactados con suficiente detalle, utilizando métricas claras y cuantificables, evitando términos vagos como "rápido" o "seguro", y vinculándolos a estándares de la industria.
- Se deberían incluir criterios de aceptación específicos que permitan verificar su cumplimiento durante las pruebas del sistema.

- El equipo de sistemas debería realizar talleres de capacitación sobre la redacción de especificaciones técnicas claras, para asegurar que tanto los ingenieros como los responsables de operaciones estadísticas puedan comunicar sus necesidades de manera efectiva.

Utilizar diagramas, esquemas y mapas mentales para visualizar las funciones principales del sistema, ya que estos facilitan la comprensión del flujo del sistema y las conexiones entre sus diferentes componentes por parte de los diferentes equipos responsables del diseño.

- Los equipos de diseño y sistemas deberían utilizar representaciones visuales, como diagramas de flujo y mapas mentales, para complementar la documentación de los requerimientos, ya que estos permiten organizar y estructurar de manera clara las ideas dispersas, consolidándolas en un esquema coherente durante la recolección de requerimientos.
- Las especificaciones de requerimientos deberían incluir diagramas de flujo de trabajo para ilustrar cómo interactúan las diferentes partes del sistema.
- El equipo de sistemas podrá incorporar diagramas UML (Unified Modeling Language) cuando sea necesario para representar visualmente la estructura del sistema.
- Las especificaciones de requerimientos deberían proporcionar material visual de apoyo, como diagramas simples que representen la arquitectura del sistema y prototipos visuales, para facilitar la comprensión de los requerimientos técnicos y funcionales por parte de todos los interesados y validar estos requerimientos con los usuarios antes de su desarrollo completo.

Involucrar a todos los interesados clave y documentar de manera clara los hallazgos, utilizando diversas técnicas para asegurar que se capturen todas las necesidades de forma precisa y completa.

- En la recolección de requerimientos, se recomienda utilizar diversas técnicas como entrevistas, talleres, encuestas y observación, para obtener una visión integral de las necesidades de los usuarios.
- Los responsables de esta tarea deberían documentar de manera clara y estructurada los hallazgos de cada técnica, asegurando que se capturen todos los detalles relevantes.
- Es esencial involucrar a todos los interesados clave, desde el equipo temático hasta los usuarios finales, para garantizar que se consideren todas las perspectivas y las necesidades.
- Cuando la recolección de requerimientos se realice de manera directa, es importante organizar sesiones con una agenda clara y objetivos definidos, asegurando la participación de todos los interesados.

Diferenciar claramente entre los requerimientos funcionales y no funcionales, asegurando que los servicios requeridos por el usuario estén bien definidos y detallados para garantizar una comprensión precisa de sus necesidades.

- Se recomienda documentar los requerimientos funcionales (qué debe hacer el sistema) y no funcionales (rendimiento, seguridad, escalabilidad) por separado, asegurando que ambos tipos de requerimientos sean específicos y medibles.
- En la descripción de las especificaciones se recomienda generar métricas cuantificables para los requerimientos no funcionales, lo que permitirá su verificación durante las pruebas.
- Todos los servicios requeridos deberían estar especificados de manera integral y sin contradicciones.
- En la generación de especificaciones se deberían utilizar descripciones detalladas de los servicios esperados y validar continuamente con el equipo temático para asegurar que los servicios se ajusten a las expectativas.

Actualizar continuamente las especificaciones de los requerimientos, revisándolos regularmente para reflejar los cambios en las prioridades, el entorno tecnológico o los recursos disponibles, asegurando que se mantengan alineados con las necesidades del proyecto.

- La entidad debería implementar un proceso formal de control de versiones que permita actualizar las especificaciones a lo largo del ciclo de vida del proyecto.
- El equipo de sistema debería asegurar que todos los interesados tengan acceso a las versiones más actualizadas de las especificaciones en cualquier momento.
- Los equipos deberían establecer puntos de control durante el desarrollo donde los equipos revisen y ajusten los requerimientos según el avance del proyecto.
- A través del proceso de diseño y desarrollo se debería realizar revisiones iterativas y colaborativas del documento de especificaciones, asegurando que todos los cambios queden documentados y sean comprensibles para los involucrados.

Validar continuamente los requerimientos con los interesados y establecer un control riguroso de cambios para garantizar que cualquier modificación sea gestionada de manera adecuada, manteniendo la coherencia con los objetivos del proyecto.

- El equipo de sistemas debería realizar validaciones regulares de los requerimientos con los interesados clave para asegurar que reflejan correctamente las necesidades y las expectativas.
- Se recomienda utilizar técnicas de revisión formal, como la comparación de interpretaciones entre los participantes, para identificar y corregir ambigüedades, asegurando que los requerimientos sean realizables y alineados con las expectativas de los usuarios.
- Los equipos a cargo del diseño deberían planificar revisiones formales periódicas de los requerimientos con los interesados, utilizando listas de verificación para garantizar que se revisen todas las áreas clave.
- Los equipos de sistemas deberían implementar un procedimiento formal de control de cambios para manejar modificaciones en los requerimientos de manera controlada y transparente.
- Es necesario que en el diseño los responsables mantengan un registro detallado de todas las versiones del documento de requerimientos, con la justificación de cada cambio, para asegurar trazabilidad.
- Los distintos equipos vinculados al diseño deberían garantizar que las especificaciones permitan la incorporación de cambios sin interrumpir el flujo de trabajo.
- Los distintos equipos vinculados al diseño deberían definir *sprints* o ciclos cortos para revisar y ajustar los requerimientos a medida que el desarrollo avanza.

Utilizar plantillas de documentación estandarizadas y mantener una matriz de trazabilidad de requerimientos, asegurando que cada requisito sea fácilmente rastreable y su progreso esté claramente documentado a lo largo del proyecto.

- La entidad debería utilizar plantillas estandarizadas para documentar los requerimientos y así facilitar la comprensión y el manejo de la información.
- Las plantillas deberían incluir secciones para la descripción, la justificación y los criterios de aceptación de cada requerimiento, así como una priorización clara de los mismos.
- Los responsables del diseño deberían establecer y mantener una matriz de trazabilidad que relacione cada requerimiento con su implementación, prueba y verificación, asegurando que todos los requerimientos sean rastreados y cumplidos adecuadamente a lo largo del ciclo de desarrollo.
- Se debería actualizar la matriz de trazabilidad continuamente, asegurando que los cambios en los requerimientos se reflejen en todas las fases del proyecto.

- Hay que asegurar que la trazabilidad permita identificar cualquier cambio en los requerimientos y su impacto en otros aspectos del sistema.
- Es necesario involucrar a todas las partes interesadas en el proceso de trazabilidad para garantizar que cada cambio sea correctamente documentado.
- Los equipos encargados del diseño deberían asegurarse de que cada cambio en los requerimientos sea revisado y aprobado por todos los interesados.
- Es necesario documentar cada decisión tomada respecto a los cambios de los requerimientos, con su justificación correspondiente.

Definir criterios claros de aceptación, realizar pruebas de concepto y aplicar una estrategia de verificación mediante pruebas y revisiones para asegurar el cumplimiento de los requerimientos.

- El equipo de diseño debería incluir criterios de aceptación específicos y medibles para cada requerimiento, asegurando que se pueda verificar su cumplimiento durante las pruebas.
- El equipo de diseño debería definir y validar los criterios de aceptación con los usuarios finales al inicio del proyecto, antes de proceder con la implementación.
- Se recomienda realizar pruebas de concepto o prototipos antes de la implementación completa, para validar la funcionalidad y los requerimientos, detectando problemas y ajustando el sistema desde etapas tempranas, asegurando que cumpla con las expectativas de los usuarios y los estándares definidos.
- El equipo de diseño debería utilizar prototipos para visualizar los requerimientos y asegurar que todos los interesados comprendan cómo se implementarán las funcionalidades.
- Se recomienda definir casos de prueba específicos para cada requerimiento funcional y no funcional, asegurando que estos se verifiquen de manera objetiva.
- Los responsables del diseño deberían establecer un cronograma de pruebas que permita a los interesados revisar y validar cada funcionalidad a medida que se desarrolla.

Planificar para la evolución del sistema, asegurando la inclusión de procedimientos de mantenimiento que permitan adaptaciones y mejoras futuras de manera eficiente.

- El equipo de sistemas debería incluir en las especificaciones información sobre posibles evoluciones futuras del sistema, como mejoras de hardware, cambios en la operación estadística o nuevas funcionalidades.

- El equipo de sistemas debería definir estrategias de mantenimiento para asegurar que el sistema pueda adaptarse a futuras necesidades sin comprometer su funcionalidad actual.
- Se recomienda documentar de manera clara los procedimientos de mantenimiento preventivo y correctivo del sistema.
- La documentación debería incluir instrucciones detalladas y cronogramas de mantenimiento para asegurar la continuidad operativa del sistema a largo plazo.
- Los responsables deberían almacenar y asegurar el acceso centralizado a la documentación para facilitar su consulta y actualización por todos los interesados a lo largo del proyecto.
- Se sugiere centralizar la documentación del proyecto en un repositorio accesible para todos los interesados y así garantizar que tengan acceso a las versiones más recientes y aplicando una política clara de control de versiones.

Definir una priorización clara de los requerimientos para garantizar que los recursos se enfoquen en las necesidades más importantes y urgentes del proyecto.

- Desde el inicio, se debería establecer un proceso claro de priorización de los requerimientos, clasificando cada uno según su importancia y urgencia.
- En conjunto, los diferentes equipos, pueden utilizar el método MoSCoW (Must, Should, Could, Won't) para priorizar los requerimientos.
- Los equipos encargados del diseño deberían realizar talleres con los interesados para definir las prioridades en función de los objetivos de la operación estadística y los recursos disponibles.
- Se recomienda distinguir entre requerimientos obligatorios y opcionales para asegurar que los desarrolladores se centren en lo más importante.
- Hay que asegurar que los requerimientos críticos estén claramente marcados como "esenciales" y que los opcionales sean etiquetados como "deseables".
- Se recomienda incluir en el documento de especificaciones una justificación para cada prioridad asignada.

Diseñar especificaciones que permitan cierta flexibilidad, de manera que los cambios necesarios en el futuro no afecten el sistema de manera significativa.

- En el diseño se deberían definir márgenes para la evolución de los requerimientos, de modo que las especificaciones puedan adaptarse a cambios en las necesidades o en el entorno sin una reconstrucción completa del sistema.
- Mantener una trazabilidad de los cambios realizados en los requerimientos, asegurando que cualquier modificación esté documentada y justificada.
- En el diseño es necesario dejar espacios en las especificaciones para modificaciones que puedan surgir durante el proceso de desarrollo.
- En la documentación incluir una sección para futuros requerimientos o mejoras previstas que se podrían implementar en iteraciones posteriores.

Implementar herramientas especializadas para la gestión de requerimientos que faciliten el seguimiento, la colaboración entre los equipos, la trazabilidad, el control de versiones y la validación continua de los mismos.

- Los encargados del diseño del software deberían asegurarse de que todos los involucrados en el proyecto tengan acceso a la herramienta y puedan actualizar el estado de los requerimientos en tiempo real.
- Los equipos de sistemas pueden aplicar metodologías ágiles para gestionar los cambios en los requerimientos de manera más flexible y adaptativa.
- Los equipos de sistemas pueden usar herramientas como Scrum o Kanban para organizar la gestión de los requerimientos y facilitar la adaptación a cambios en el proyecto.

Implementar un control de versiones efectivo en la documentación de los requerimientos para asegurar el seguimiento preciso de los cambios realizados durante todo el proyecto y facilitar la revisión histórica de decisiones clave.

- La entidad debería mantener una política clara de control de versiones en la que se indique la fecha, el autor y la justificación de cada cambio.
- El equipo de sistemas debería asegurarse de que todos los interesados trabajen siempre con la versión más reciente del documento de especificaciones.
- El equipo de sistemas debería establecer un sistema de versionado con identificadores únicos que reflejen el estado actual de la solución y su nivel de madurez. Estos identificadores deben

acompañar cada cambio o mejora, permitiendo así rastrear el avance, documentar las distintas etapas (desarrollo, pruebas, preproducción, producción) y facilitar la identificación de hitos clave a lo largo del ciclo de vida del producto.

Utilizar una estructura documental estandarizada que asegure la coherencia y uniformidad en todo el proyecto, facilitando la comprensión y el manejo de la información por todos los equipos involucrados.

- En los documentos de las especificaciones se recomienda incluir un índice y un glosario en el documento para facilitar la navegación y la comprensión de términos técnicos.
- Los responsables de la documentación deberían revisar la estructura de cada documento regularmente para asegurar que cumple con los estándares de calidad establecidos en la entidad.
- Se recomienda concentrar la documentación del ciclo de vida de los requerimientos en un repositorio único (formatos, cuestionarios, planes, historias de usuario, mockups, matrices de pruebas) para facilitar la consulta, evitar redundancias, asegurar trazabilidad y contar con evidencia para auditorías, mantenimiento y desarrollo.
- Se recomienda adoptar herramientas digitales (por ejemplo, GLPI o Jira) para gestionar esta documentación de forma centralizada, mejorar la colaboración entre equipos, asegurar fácil acceso a la información y disponer de evidencia clara para control interno, auditorías y certificaciones.

Identificar y gestionar los riesgos asociados con los requerimientos de manera continua durante todo el ciclo de desarrollo y asegurar que se implementen estrategias de mitigación efectivas y oportunas.

- Los relacionados con el proyecto deberían realizar un análisis de riesgo para identificar los requerimientos más vulnerables a cambios o problemas durante la implementación.
- Se recomienda establecer y documentar estrategias de mitigación de riesgos para cada requerimiento y explicar cómo se abordarán los posibles problemas.
- Se recomienda incluir un análisis de riesgos como parte de las revisiones periódicas planificadas del proyecto y ajustar las estrategias de mitigación, según sea necesario, durante todo el desarrollo.
- El equipo de diseño debería utilizar un enfoque proactivo para identificar problemas potenciales antes de que impacten el desarrollo.

6. Actividades para elaborar las especificaciones de requerimientos

La correcta elaboración de especificaciones de requerimientos es crucial para garantizar que los sistemas, aplicativos o softwares diseñados para una operación estadística cumplan con las necesidades y expectativas de los usuarios. Este proceso implica una serie de actividades organizadas que aseguran una adecuada comunicación entre los equipos temáticos, logísticos y técnicos. A continuación, se describen los pasos necesarios para desarrollar unas especificaciones claras y completas, ofreciendo instrucciones prácticas para todos los involucrados.

6.1. Identificación de necesidades y definición de objetivos y alcance

El primer paso es identificar las necesidades específicas del sistema o software dentro de la operación estadística. Esta actividad permite definir claramente los objetivos del proyecto y garantizar que se cubran todos los aspectos necesarios para el éxito del sistema. Los pasos clave incluyen:

Reuniones iniciales: el equipo de diseño o rediseño de la operación estadística se reúne con el equipo de sistemas para explicar las principales características metodológicas de la operación y las actividades que pueden ser automatizadas. Se definen los productos que se necesitan desarrollar para apoyar la operación.

Ejemplo:



El equipo de diseño de la operación estadística sobre salud mental se reúne con el equipo de sistema de la entidad para socializar su ficha metodológica e informar la necesidad de contar con un sistema que permita capturar datos sobre este tema de manera eficiente y precisa, ya que usan formularios en papel, lo cual es ineficiente y propenso a errores.

Los principales problemas son: la duplicación de datos, la falta de precisión y la dificultad para consolidar la información de diferentes fuentes.

Definición del alcance: se identifican los servicios que el sistema ofrecerá, las restricciones técnicas, su rendimiento, y la interoperabilidad con otros sistemas de la operación. También se definen los objetivos a alcanzar y se elabora un cronograma con responsables asignados.

Ejemplo:

Para definir los objetivos y el alcance, se puede responder lo siguiente:

¿Cuál es servicio del sistema?

Servicio que ofrece el sistema es capturar información sobre salud mental, historial médico, tratamiento y datos de pacientes. Estos datos se utilizan para estudios epidemiológicos y para informar políticas de salud pública.

¿Cuántos usuarios y cuál es el modo de acceso?

Alrededor de 150 usuarios, incluyendo personal de recolección de datos (supervisores y encuestadores), equipo de diseño de la operación estadística y equipo de sistemas necesitarán acceder al sistema desde la entidad y de forma remota.

¿Cuál es la funcionalidad?

El sistema permitirá capturar, almacenar y transmitir datos, así como realizar validaciones en tiempo real para garantizar la calidad de la información recopilada.

¿Cuáles son los objetivos del nuevo software de captura de datos?

Mejorar la eficiencia, la precisión y la calidad en la captura de datos con el fin de facilitar el procesamiento de datos y el análisis estadístico para la toma de decisiones en salud pública.

¿En cuánto tiempo se e desarrollaría el software y cuál es su presupuesto?

Sistema implementado en 6 meses con un presupuesto de \$10.000.000.

Documento de objetivos: se elabora un documento preliminar que describa los objetivos, el alcance y el cronograma del proyecto, lo que servirá de base para la siguiente fase de definición de requerimientos.

Ejemplo:

Objetivos

- Mejorar la eficiencia de la captura de datos a través de los cuestionarios, reduciendo el tiempo de recolección en un 70%.
- Fortalecer la calidad de los datos recolectados, validando su información en tiempo real.
- Facilitar la fase de procesamiento de la Operación estadística, así como su análisis.
- Garantizar que el software de captura de datos sea accesible tanto desde la oficina como de forma remota.

Alcance

Diseñar y desarrollar un sistema de captura para la operación estadística sobre salud mental que permita la recolección, almacenamiento y transmisión de datos, facilitando el procesamiento y análisis de resultados estadísticos. A través de este software se generarán informes y validaciones en tiempo real, y se capacitará a 150 usuarios en su uso. La entrega del sistema se realizará en un período de 6 meses, con un presupuesto de \$10.000.000.

6.2. Definición de requerimientos del usuario

Una vez establecidas las necesidades generales, se procede a definir los requerimientos del usuario. Esta actividad se centra en describir qué necesita el usuario que el sistema haga y por qué es importante (Alonso, Martínez, y Segovia; UC). Los pasos clave son:

Redacción clara y en lenguaje natural: los requerimientos deben estar redactados en un lenguaje que sea accesible para todos los involucrados, evitando el uso de tecnicismos innecesarios.

Descripción de funciones: se deben detallar todas las funciones que el sistema debe realizar, especificando su comportamiento en diferentes escenarios operativos. Se recomienda utilizar diagramas y tablas que muestren las interacciones y flujos de información.

Justificación de los requerimientos: cada requerimiento debe ir acompañado de una justificación que explique por qué es necesario para la operación. Esto facilita la validación y las futuras modificaciones si es necesario.

Priorización de requerimientos: se deben distinguir entre requerimientos obligatorios (esenciales para el funcionamiento del sistema) y deseables (aquellos que mejorarían el sistema, pero no son críticos).

Estandarización: se debe utilizar un formato estándar que incluya la descripción de las funciones, sus entradas y salidas, precondiciones, postcondiciones, y efectos secundarios si los hay. Asimismo, debe asegurarse de que todos los involucrados sigan este formato para una documentación coherente.

A continuación, se describe un solo ejemplo de la definición de un requerimiento del usuario para el nuevo software de recolección de datos para la operación estadística sobre salud mental.

Ejemplo:

Requerimiento funcional: validación de datos en tiempo real.

Descripción: el sistema debe validar los datos ingresados en el aplicativo en tiempo real para no incurrir en errores.

Justificación: garantizar la calidad y la precisión de los datos recolectados.

Requerimiento no funcional: seguridad de los datos.

Descripción: el sistema debe implementar medidas de seguridad para proteger la información sensible de los encuestados sobre salud mental.

Justificación: asegura la confidencialidad de los encuestados y da cumplimiento a la normatividad protección de datos. ((Ley 79 de 1993, Por la cual se regula la realización de los censos de población y vivienda en todo el territorio nacional. Guía para el uso y aprovechamiento de Datos Abiertos en Colombia del Ministerio de Tecnologías de la Información y las Comunicaciones.

6.3. Modelado del sistema

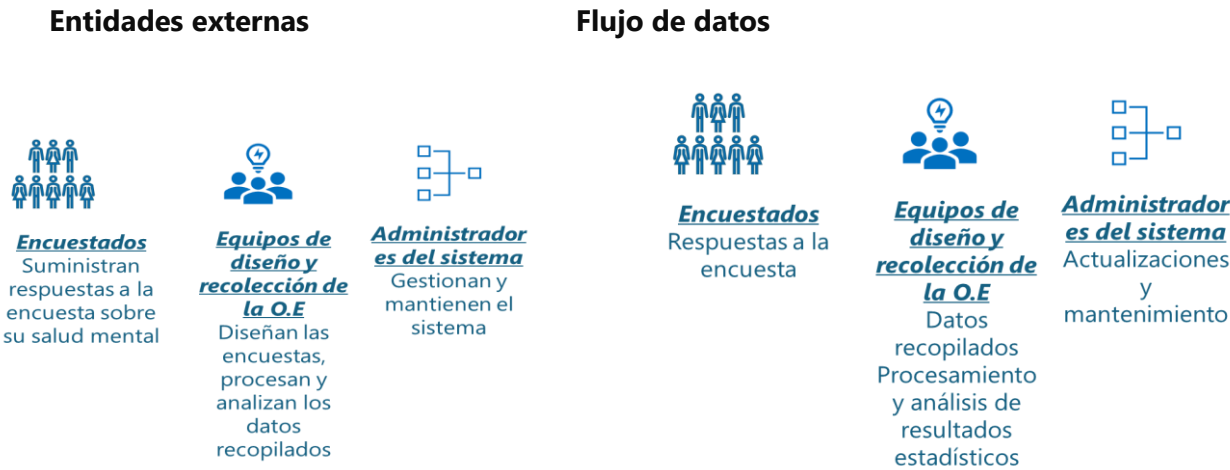
El equipo técnico debe crear un modelo gráfico del sistema para visualizar su estructura y comportamiento. Esto facilitará la comunicación entre el equipo de diseño de la operación estadística y el equipo de sistemas y permite asegurar que todos comprendan cómo funcionará el sistema en su entorno operativo. A continuación, se presenta tipos de diagramas:

Diagramas de contexto: se crea un diagrama de contexto que muestra cómo el sistema interactúa con otros sistemas y actores externos, facilitando la identificación de interacciones clave.

Diagramas de flujo de datos y actividad: estos diagramas ayudan a describir cómo fluye la información a través del sistema y cómo se procesarán las diferentes funciones.

Ejemplo:

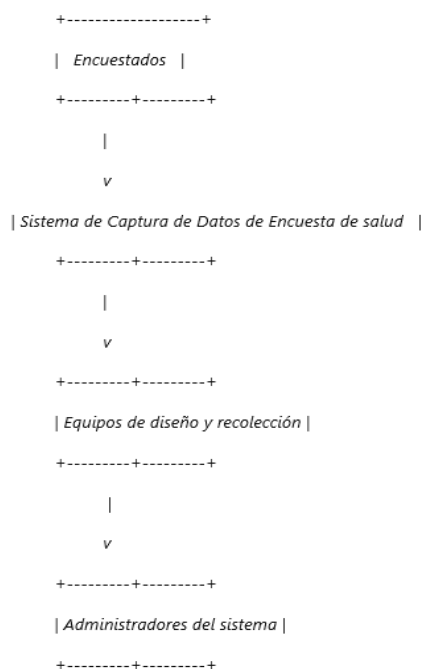
Imagen 1. Diagrama de contexto para un sistema de captura de datos de la Encuesta sobre Salud Mental



Fuente: DANE, Regulación estadística.

Ejemplo visual

Imagen 2. Ejemplo visual para un sistema de captura de datos de la Encuesta sobre Salud Mental



Fuente: DANE, Regulación estadística.

6.4. Establecimiento de requerimientos detallados

A partir de los requerimientos del usuario, se desarrollan los requerimientos detallados del sistema. Esta fase es crucial para asegurar que el equipo técnico tenga toda la información necesaria para implementar el sistema de manera eficiente. Las actividades incluyen:

Descripción de funciones técnicas: describir detalladamente las funciones del sistema, incluyendo sus entradas, salidas, restricciones de rendimiento y seguridad, y cualquier excepción que deba manejarse.

Definición de interacciones y protocolo: detallar cómo los diferentes componentes del sistema interactúan entre sí y con actores externos. Esto incluye especificar los mensajes que se intercambiarán y los protocolos de interacción.

Anotaciones gráficas: usar anotaciones gráficas como diagramas de secuencia y de entidad-relación para visualizar las interacciones y dependencias entre los componentes del sistema.

6.5. Validación de especificaciones

La validación es esencial para garantizar que los requerimientos satisfagan las necesidades identificadas y que el sistema pueda implementarse con éxito. Se pueden utilizar varias técnicas de validación, como:

Revisiones de requerimientos: un equipo con las personas que participan en el desarrollo de la operación estadística revisa sistemáticamente los requerimientos para identificar errores o inconsistencias.

Prototipos: se desarrollan prototipos funcionales que permiten a los usuarios experimentar con el sistema y validar su usabilidad y funcionalidad.

Casos de prueba: se diseñan casos de prueba a partir de los requerimientos del usuario para asegurar que todos los requerimientos sean verificables. Si un requerimiento es difícil de probar, podría ser ambiguo y requerir revisión.

6.6. Consolidación de los documentos de especificaciones

Finalmente, se elaboran los documentos que incluyan todos los aspectos establecidos en cada una de las actividades anteriores, que describan adecuadamente requerimientos del sistema y asegura que sea una referencia clara y útil durante todo el desarrollo.

Bibliografía

- Agencia Nacional de Tierras (ANT). (2021). Política para la gestión de la información. Obtenido de: <https://www.ant.gov.co/wp-content/uploads/2021/10/GINFO-Politica-011.pdf>
- Alonso, F., Martínez, L., y Segovia, F. (2005). Introducción a la ingeniería de software: modelos de desarrollo de programas. Madrid, España: Delta Publicaciones. Obtenido de: https://books.google.com.co/books?id=rXU-WS4UatYC&pg=PA82&dq=especificaciones+funcionales&hl=es&sa=X&ved=0ahUKEwiph8_S1NPoAhUnVt8KHblhBD0Q6AEILzAB#v=onepage&q=especificaciones%20funcionales&f=false
- Departamento Administrativo Nacional de Estadística (DANE). (2024). Procedimiento de Desarrollo y Mantenimiento de Sistemas de Información (Versión 5). Oficina de Sistemas. Obtenido de: <https://dane.isolucion.co>
- Departamento Administrativo Nacional de Estadística (DANE). (2024). Guía Metodológica Sistemas de Información (Versión 2). Oficina de Sistemas. Obtenido de: <https://dane.isolucion.co>
- Departamento Administrativo Nacional de Estadística (DANE), D. A. (03 de 10 de 2023). Ley 2335. Obtenido de Por la cual se expiden disposiciones sobre las estadísticas oficiales en el país. Obtenido de: <https://www.dane.gov.co/index.php/acerca-del-dane/informacion-institucional/normatividad/leyes/ley-de-estadisticas-oficiales-de-colombia>
- Departamento Administrativo Nacional de Estadística (DANE). *Sistema de consulta de conceptos estandarizados*. Obtenido de: https://conceptos.dane.gov.co/conceptos/conceptos_catalogo
- Departamento Administrativo Nacional de Estadística (DANE). (2020). Lineamiento para el proceso estadístico en el Sistema Estadístico Nacional. Obtenido de: <https://www.sen.gov.co/sites/default/files/migracion-files/sen/normatividad/Lineamientos Proceso Estad%C3%ADstico v2.pdf>
- Departamento Administrativo Nacional de Estadística (DANE) (2020). Norma Técnica de la Calidad del Proceso Estadístico.
- Departamento Nacional de Planeación (DNP). (2020). Guía para la elaboración y presentación de historias de usuario. Obtenido de: <https://colaboracion.dnp.gov.co/CDTI/Oficina%20Informatica/Sistemas%20de%20informaci%C3%B3n/Gu%C3%ADas%20Formatos%20Plantillas/Gu%C3%ADa%20para%20la%20Elaboraci%C3%B3n%20y%20Presentaci%C3%B3n%20de%20Historias%20de%20Usuario.pdf>
- Departamento Nacional de Planeación (DNP). (2020). Manual operativo para la implementación y mantenimiento de sistemas de información. Obtenido de: <https://colaboracion.dnp.gov.co/CDTI/Inversiones%20y%20finanzas%20pblicas/EPICO/M-TI-01%20Manual%20Op.%20Implementaci%C3%B3n%20y%20Mantenimiento%20de%20S.I.%20DNP.Pu.pdf>
- Departamento Nacional de Planeación (DNP). (2020). Guía para la elaboración del manual técnico y de operación del sistema. Obtenido de: <https://colaboracion.dnp.gov.co/CDTI/Oficina%20Informatica/Sistemas%20de%20informaci%C3%B3n>

[3n/Gu%C3%ADas%20Formatos%20Plantillas/Gu%C3%ADa%20para%20la%20Elaboraci%C3%B3n%20del%20Manual%20T%C3%A9cnico%20y%20de%20Operaci%C3%B3n%20del%20Sistema.pdf](#)

Lozano, Dairo. (2016). Guía rápida para especificaciones funcionales. SAP Community. Obtenido de: <https://blogs.sap.com/2016/05/17/ricefw-guia-rapida-para-especificaciones-funcionales-introduccion/>

Northware. (n.d.). Técnicas efectivas para la toma de requerimientos. Obtenido de: <https://www.northware.mx/blog/tecnicas-efectivas-para-la-toma-de-requerimientos/>

QbD Group. (2022). Validación de sistemas informatizados: Requisitos de usuario (URS). <https://qbdgroup.com/es-es/blog/validacion-de-sistemas-informatizados-requisitos-de-usuario-urs/#:~:text=%C2%BFQu%C3%A9%20son%20los%20Requisitos%20de,%E2%80%9D%2C%20la%20justificaci%C3%B3n%20del%20sistema>

Sommerville, Ian. (2011). Software Ingengering (9 ed.). Boston, USA: Pearson. Obtenido de: https://dinus.ac.id/repository/docs/ajar/RPL-Sommerville_-_Software_Engineering_9ed.pdf

Universidad Nacional Abierta y a Distancia (UNAD). (n.d.). Reproductor de OVA: Guía de identificación de requerimientos. https://repository.unad.edu.co/reproductor-ova/10596_35614/index.html

Universidad de La Salle. (2021). Guía Metodológica para la Toma de Requerimientos en Sistemas de Información Aplicando la Norma IREB. Obtenido de: https://ciencia.lasalle.edu.co/cgi/viewcontent.cgi?article=2066&context=sistemas_informacion_documentacion

Universidad de Cantabria (UC). (2008). Análisis de requisitos y especificación de una aplicación. Obtenido de: https://www.ctr.unican.es/asignaturas/Ingenieria_Software_4_F/Doc/M3_08_Especificacion-2011.pdf

Universidad de Granada (UGR). Especificación de requerimientos: diseño de bases de datos. Obtenido de: <https://elvex.ugr.es/idbis/db/docs/design/2-requirements.pdf>

Wong Durand, Sandra (2017). Análisis y requerimientos de software. Huancayo: Universidad Continental. Obtenido de: https://repositorio.continental.edu.pe/bitstream/20.500.12394/4281/1/DO_FIN_103_MAI_UC0939_2018.pdf