

**UNIDAD NACIONAL DE PROTECCIÓN**

CONSTITUCIÓN DEL PROYECTO

**ECOSISTEMA UNP**

Bogotá D.C julio de 2024

# Resumen Ejecutivo

## Descripción del proyecto

El Ecosistema de Información de la Unidad Nacional de Protección (UNP) es la proyección de un entorno digital y de gestión de la información que permitirá la recopilación, el análisis y la gestión de los datos producidos por la entidad en el cumplimiento de su misionalidad y su administración. Este se caracteriza por ser un entorno integrado y adaptativo de personas, procesos, políticas y tecnologías que, en su interacción, garantizan el acceso oportuno, relevante, preciso y completo de la información. De manera que contempla cómo —en términos operativos— esta se crea, distribuye, utiliza y se gestiona dentro de la entidad a partir de sus sistemas, subsistemas, módulos, funciones y los roles de usuarios.

Es, adicionalmente, una solución compleja o un grupo de soluciones que por medio de recursos tecnológicos (aplicaciones web, móviles, de escritorio, de consola y API), humanos e infraestructurales, así como de la mejora continua y funcional de los procedimientos, busca una gestión y aprovechamiento en tiempo real de los datos y la información para la toma de decisiones oportunas y la optimización de los recursos.

## Caracterización del proyecto.

|  |  |
| --- | --- |
| Título Del Proyecto | Ecosistema de Información de la Unidad Nacional de Protección (EI-UNP) |
| Enviado Por | Andrés Mauricio Soto Taborda | Versión N° | 01 |
| Fecha de la propuesta | 19 de julio de 2024 | Tiempo de ejecución  | 18 meses |
| Fecha Inicio Proyectada | 01 de agosto de 2024 | Fecha finalización proyectada | 1. de mayo de 2026
 |

## Líder del Proyecto

|  |  |  |
| --- | --- | --- |
| Nombre | Cargo | Departamento |
| Andrés Mauricio Soto Taborda | Líder del Proyecto | Oficina Asesora de Planeación e Información (OAPI) |

## Equipo Proyecto

|  |  |
| --- | --- |
| Cargo | Departamento |
| Líder de Proceso (1 persona) | Oficina Asesora de Planeación e Información (OAPI) |
| Líder de Calidad (1 persona) | Oficina Asesora de Planeación e Información (OAPI) |
| Colíder Equipo desarrollo  | Subdirección Especializada de Protección (SESP) / 1 personas. |
| Colíder Equipo desarrollo  | Subdirección de Evaluación del Riesgo (SER) / 1 persona. |
| Colíder Equipo desarrollo  | Subdirección de Protección (SP) / 1 persona. |
| Líder Equipo Infraestructura | Oficina Asesora de Planeación e Información (OAPI) / Grupo de Gestión tecnológica. |
| Colíder de Desarrollo General  | Oficina Asesora de Planeación e Información (OAPI) / Grupo de Gestión tecnológica. |

## Objetivo principal.

Desarrollar un ecosistema de información en la Unidad Nacional de Protección (UNP) que modernice la infraestructura tecnológica, promueva la interoperabilidad, garantice la seguridad de la información y cumpla con el marco normativo. Este ecosistema permitirá una gestión del cambio eficiente de los procesos internos y la integración con sistemas externos.

## Alcance del Proyecto

El Ecosistema de información de la UNP comprenderá el desarrollo de roles, funciones, módulos, subsistemas y sistemas que abarcará los procesos, procedimientos y tareas necesarias para asegurar la misionalidad y visión de la UNP, incluyendo asimismo aquellos proceso, procedimientos y tareas trasversales a la entidad. A continuación, se desglosan los alcanzables de acuerdo con la magnitud del proyecto.

* **Desarrollo y Modernización Tecnológica:**
* Diseñar, desarrollar e implementar las aplicaciones web mencionadas en el corto plazo (servicio al ciudadano, asesoría poblacional, asignación, análisis de riesgo, pre-mesa y subcomisión técnica).
* Actualizar y modernizar la infraestructura tecnológica existente para garantizar la escalabilidad, seguridad y eficiencia del ecosistema.
* Integrar sistemas y subsistemas existentes para lograr una visión unificada y coherente de la información.
* **Promoción de la Interoperabilidad:**

* Establecer protocolos y estándares para la comunicación entre diferentes módulos y sistemas dentro de la UNP.
* Facilitar la interoperabilidad con otras instituciones y entidades relacionadas con la protección y seguridad.
* **Seguridad de la Información:**
* Implementar medidas de seguridad robustas para proteger la integridad, confidencialidad y disponibilidad de los datos.
* Cumplir con las normativas y estándares de seguridad, como la norma ISO 27001.
* **Gestión Eficiente de Procesos Internos:**
* Optimizar los flujos de trabajo internos mediante la automatización y digitalización de procesos.
* Facilitar la toma de decisiones basada en información verídica y confiable.
* **Integración con Sistemas Externos:**
* Establecer interfaces y conexiones con sistemas externos relevantes para la UNP.
* Facilitar la colaboración y el intercambio de información con otras entidades.
* **Mejora Continua y Evaluación de Resultados:**
* Establecer indicadores clave de desempeño (KPIs) para medir el éxito del ecosistema.
* Realizar evaluaciones periódicas para identificar áreas de mejora y ajustar el sistema según sea necesario.

## Cronograma e hitos

**Tiempo de ejecución:** 15 Meses

**Entregable**: Ecosistema digital de la UNP.

**Cronograma**: Anexo adjunto

## Estructura del proyecto

|  |  |
| --- | --- |
| Rol | Responsabilidad |
| Gerente  | Rol encargado de la supervisión, validación y verificación que el proyecto sea un sistema integrado donde la información de sus distintos grupos de trabajo se compile, almacene, analice y se utilice de manera eficiente; aprobación de los Hitos de entrega, y la correcta analítica de los datos. |
| Líder de Proyecto | Gestionar capacidad del equipo, gestión de usuario de red, asignación y arquitectura constante del desarrollo y programación, gestionar impedimentos de capacidad de las partes interesadas, garantiza la seguridad de la información, y el cumplimiento de la norma, gestionar usuarios de aplicación, permisos y roles correspondientes, cumplimiento de requisitos funcionales y no funcionales (Desarrolladores). |
| Equipo de Procesos y Calidad | Definición de roles, diseño flujo actual, diseño flujo propuesto o mejorado y descripción de actividades detalladas del flujo mejorado (incluye recomendaciones y observaciones), Generar y entregar Información real del proceso / actividad: descripciones, estadísticas, documentación existente y definición general actual del flujo, en cumplimiento a las normas de calidad. Diseño flujo funcional. |
| Coordinador del proceso / Área usuaria | Generar y entregar Información real del proceso / actividad: descripciones, documentación existente, aprobaciones del flujo actual y flujo funcional propuesto y final de documento.  |
| Equipo Desarrollo | Desarrollar proceso / actividad mejorada, en la herramienta y lenguajes de programación adaptados, para su desarrollo, elaboración de pruebas y puesta en producción. |
| Equipo Infraestructura | Garantizar las condiciones reales del ecosistema y soportar los requerimientos funcionales y No funcionales. La seguridad y el funcionamiento eficiente de los servidores, las redes y las telecomunicaciones. Esto implica cumplir con políticas de seguridad y niveles de seguridad establecidos. Además, realiza tareas de mantenimiento en equipos y respalda información. |
| DBA (Database Administrator) general  | El rol del DBA (Database Administrator) garantizar el diseño, implementar y mantener la base de datos, asegurando su rendimiento, seguridad y disponibilidad. colabora con el equipo para definir requisitos de datos. verifica la integración y seguridad.  |

## Necesidades e inversión

|  |  |
| --- | --- |
| Necesidad | Descripción |
| Recurso Humano | Conformado por personal desarrollador de software, ingenieros, asesores y demás personal idóneo de la Unidad Nacional de Protección |
| Recurso Tecnológico | Herramientas, dispositivos e infraestructura que se requiere para el desarrollo, las pruebas y despliegue durante todas las fases y en todos los niveles. |
| Recursos Físicos | Equipamiento y demandas infraestructurales que garantizan el durante y después del desarrollo del ecosistema de información. Deben responder a los requisitos funcionales y no funcionales. |
| Recursos Intangibles | * Sensibilización en Gobierno Digital
* Mesas de trabajo con el Ministerio de Tecnologías de la Información y las Comunicaciones de Colombia. MINTIC.
* Política de Infraestructura interna de la entidad.
* Cualquier sensibilización propia del desarrollo interno y políticas de la entidad.
 |

# Requisitos del Sistema

## Requisitos Funcionales:

Los requerimientos funcionales del sistema son declaraciones implementadas con características específicas para satisfacer las necesidades de los actores involucrados, así como permitir una interacción intuitiva entre todos sus componentes y la toma correcta de decisiones transformando los datos.

Imagen 1
Estructura de requerimientos funcionales



Fuente: Elaboración propia

A continuación, se puede visualizar los requerimientos funcionales en grandes conjuntos. Los principales y más relevantes para este proyecto son:

* **Servicio API:**

Un Servicio API (Interfaz de Programación de Aplicaciones) es un conjunto de reglas y protocolos que permite que diferentes aplicaciones de software interactúen entre sí. Estas interacciones pueden incluir la recuperación, creación, modificación y eliminación de datos. Además, es un «lenguaje» estándar para comunicar diferentes aplicaciones a través de la Internet, de manera que es usado en la convergencia e interoperabilidad entre sistemas**.**

* **Conectividad a la base de datos:**

La conectividad a una base de datos generalmente implica establecer una comunicación efectiva entre una aplicación (o sistema) y la base de datos para leer, escribir y manipular datos. También controla los roles de los usuarios y los permisos habilitados para estos según sean los casos. Implícitamente, es decir, no perceptible para el usuario del sistema, guía y enmarca los protocolos de seguridad y modificación de los datos.

### Acceso seguro:

### Acceder de manera segura a diferentes sistemas de información es crucial para proteger la integridad, confidencialidad y disponibilidad de los datos.

### Integridad de los datos:

### La integridad de los datos se refiere a la precisión, consistencia y fiabilidad de la información almacenada en una base de datos o sistema de información. Es fundamental para garantizar que los datos sean exactos y completos, y que reflejen de manera correcta el estado real de los objetos o entidades que representan.

### Reglas de negocio:

### Son directrices o restricciones que reflejan las políticas, procedimientos y operaciones específicas de una organización o dominio de negocio. Estas reglas son fundamentales porque definen cómo debe comportarse el sistema de software para cumplir con los requisitos y expectativas del negocio.

### Cálculos específicos:

### Realizar cálculos específicos que van más allá de simples operaciones aritméticas. Estos cálculos pueden involucrar lógica de negocio, reglas complejas y manejo de datos especializado.

### Flujo de información:

### El flujo de información se refiere al movimiento de datos o información dentro de un sistema, organización o proceso. Este concepto es fundamental en diversos contextos, desde la gestión empresarial hasta el diseño de sistemas de información y la comunicación interpersonal.

### Tiempo de ejecución:

### Es el tiempo total que lleva completar una tarea específica, desde que se inicia hasta que se termina. En el contexto del desarrollo de software, se refiere al tiempo que tarda un programa en ejecutarse completamente para realizar una función o procesamiento determinado.

### Reportes de salida:

### Se refieren a los documentos, informes o datos que se generan como resultado del procesamiento de información dentro del sistema. Estos reportes son fundamentales porque proporcionan información procesada y estructurada que es útil para la toma de decisiones, análisis, seguimiento de métricas, y otras actividades.

### Infraestructura:

### Se refiere al conjunto de componentes tecnológicos y recursos necesarios para soportar y operar eficientemente el sistema de información de una organización. Esta infraestructura es fundamental para facilitar el procesamiento, almacenamiento, análisis y distribución de datos e información dentro de la organización.

### Validaciones:

### Las validaciones en un sistema de información se refieren al proceso de verificar y garantizar que los datos ingresados en el sistema cumplan con ciertos criterios o reglas predefinidas. Estas validaciones son fundamentales para asegurar la integridad, consistencia y precisión de los datos almacenados y procesados dentro del sistema.

###

### Dashboard:

### Una interfaz gráfica que muestra visualmente información clave, métricas y datos relevantes para facilitar la monitorización, análisis y toma de decisiones en tiempo real dentro de una organización. Los dashboards son herramientas importantes en el ámbito de la inteligencia empresarial (BI) y la gestión de datos, ya que permiten a los usuarios obtener una vista rápida y comprensible del estado actual y del rendimiento de diversos aspectos del negocio.

###

### Servicios geográficos:

### Los servicios geográficos se refieren a aplicaciones, datos y recursos relacionados con la geografía y la cartografía. Estos servicios permiten acceder, visualizar y analizar información geoespacial de manera eficiente.

## Requisitos No Funcionales:

En el desarrollo del proyecto, se identifica claramente los requisitos no funcionales, también conocidos como "requisitos de calidad" o "requisitos de atributos del sistema". Estos requisitos describen características y criterios esenciales que abarcan el rendimiento, la seguridad, la usabilidad y otros aspectos clave del sistema o software, más allá de su funcionalidad básica. Garantizando que el sistema sea eficiente, seguro, fácil de usar, mantenible, escalable y compatible, asegurando así una experiencia de usuario óptima y el cumplimiento de los estándares de calidad esperados.

Imagen 2
Esquema de requisitos no funcionales



### Rendimiento:

### El sistema debe estar en la capacidad de manejar y procesar gran número de registros diarios de manera eficiente.​

### El tiempo de respuesta para consultas y operaciones críticas no debe exceder los límites especificados.​

### Escalabilidad:

El sistema debe ser capaz de escalar para manejar un aumento en la carga de trabajo sin degradación significativa del rendimiento.​

Debe ser posible agregar recursos adicionales, como servidores o capacidades de almacenamiento, según sea necesario.​

### Disponibilidad:

El sistema debe estar disponible las 24 horas del día, los 7 días de la semana, o según políticas de funcionamiento con un tiempo de inactividad mínimo planificado para mantenimiento.​

Debe haber medidas de redundancia y tolerancia a fallos para garantizar la continuidad del servicio.

### Seguridad:

Se deben implementar controles de acceso y autenticación adecuados para proteger los datos sensibles.​

Los datos deben estar encriptados durante la transmisión y el almacenamiento.​

Debe haber protección contra amenazas como ataques de denegación de servicio (DDoS) y hacking.

### Fiabilidad:

El sistema debe ser robusto y resistente a fallos para evitar la pérdida de datos.​

Se deben implementar mecanismos de respaldo y recuperación para garantizar la integridad de los datos en caso de fallo del sistema.​

Errores de digitación implementar una recuperación

### Mantenibilidad:

El sistema debe ser fácil de mantener y actualizar sin interrupciones significativas en el servicio.​

Debe haber una documentación detallada y actualizada sobre la arquitectura del sistema, los procesos de mantenimiento y las dependencias del sistema.

### Integración:

Debe ser posible integrar el sistema con otros sistemas de la organización, como sistemas de gestión de recursos humanos o sistemas de contabilidad entre otros.​

Se deben proporcionar interfaces estándar y API para facilitar la integración con aplicaciones externas.

### Interfaz de Usuario:

La interfaz de usuario debe ser intuitiva y fácil de usar, especialmente para los usuarios que interactúan con grandes volúmenes de datos.​

Debe haber opciones de personalización para adaptarse a las preferencias del usuario.

### Cumplimiento Normativo:

El sistema debe cumplir con las regulaciones y normativas pertinentes, como GDPR, HIPAA, etc., en lo que respecta al manejo de datos personales y sensibles.​

Otros relevantes de acuerdo con el tipo de información y usuarios manejados. ​

Manejo y control de os documentos de Impresión

### Eficiencia de Los recursos:

El sistema debe utilizar eficientemente los recursos de hardware y software para minimizar los costos operativos.​ Deben implementarse políticas de gestión de recursos para optimizar el uso de CPU, memoria y almacenamiento.

### 8 requisitos de Hardware y Software:

El logro del objetivo y alcance definidos en este documento de proyecto depende crucialmente de la infraestructura disponible, que incluye la red institucional, los servidores con crecimiento progresivo, así como los recursos humanos, tecnológicos y físicos especificados. Con esta observación el cumplimiento de los entregables será posible, en los 15 meses proyectados.

# Diseño del Sistema

## Arquitectura del Sistema:

Imagen 3
Características de la arquitectura tecnológica

****

Fuente: Elaboración propia

## Diseño de la Base de Datos:

Una base de datos se define como una colección de datos interrelacionados. Los datos son hechos conocidos que pueden ser registrados y poseen un significado implícito. El propósito y uso de esta colección de datos se conoce como el enfoque de la base de datos.

Existen varias características que distinguen el enfoque de la base de datos del enfoque de escribir programas personalizados para acceder a datos almacenados en archivos. Aunque no profundizaremos en ellas, se menciona que, en la base de datos, un solo repositorio mantiene los datos definidos una vez y luego los acceden varios usuarios repetidamente mediante consultas, transacciones y programas de aplicación. Las principales características del enfoque de la base de datos se muestran en el siguiente esquema:

### Aspectos fundamentales para la base de datos

Se deben tener en cuenta dos aspectos fundamentales en el diseño de la base de datos:

### Base de datos en su tercera forma normal

La base de datos para el Ecosistema de Información (EI) de la UNP debe estar en su tercera forma normal en su primera etapa. En esta forma, todos sus atributos o campos dependen de la clave principal. Esta forma incluye la segunda y primera normal, por lo que ambas deben considerarse en el diseño de la base de datos. Aunque no son totalmente equivalentes, se pueden entender como la normalización de la base de datos; en otras palabras, una base de datos está normalizada cuando se encuentra al menos en su tercera forma normal.

### Diccionario de datos

Por ahora, se definirán los atributos de las tablas con los tipos de datos correspondientes. Cada tabla debe especificar la relación con otras tablas y su respectiva cardinalidad (de uno a uno, de uno a muchos, de muchos a uno), y si son entidades fuertes o débiles con respecto a la relación que tienen como entidades. También se debe especificar cuándo los valores van por defecto y pueden almacenar valores nulos, así como definir el propósito de la tabla.

### Parámetros

Por otra parte, la base de datos del EI de la UNP debe contar con los siguientes parámetros:

1. Todos los nombres, tanto para tablas como los atributos de las tablas, se escribirán en minúscula. Por ningún motivo se deberá escribir un nombre en mayúscula o con una mayúscula intermedia. Esto no implica que los campos ingresados no puedan contener mayúsculas según las normas ortográficas definas para el idioma español.

2. Las tablas de la base de datos del EI se nombrarán al inicio con la abreviatura o nomenclatura «eco<número>», subseguida de un guion bajo (\_) y los siguientes patrones:

|  |  |
| --- | --- |
| **Base**  | **Abreviatura**  |
| Unidad Nacional de Protección  | eco00  |
| Dirección General  | eco01  |
| Oficina Asesora de Planeación e información  | eco02  |
| Secretaria General  | eco03  |
| Subdirección de evaluación del Riesgo  | eco04  |
| Subdirección de Protección  | eco05  |
| Subdirección Especializada de Seguridad y Protección  | eco06  |
| Subdirección de Talento Humano  | eco07  |
| Oficina Asesora Jurídica  | eco08  |
| Oficina Control Interno  | eco09  |

*a) Abreviatura del grupo de trabajo (al que le corresponde a cada uno un sistema o aplicación)*

|  |  |
| --- | --- |
| **Grupo**  | **Abreviatura**  |
| GCE  | eco01\_gce  |
| GSC  | eco02\_gsc  |
| GGIM  | eco02\_ggi  |
| GPIGI  | eco02\_gpi  |
| GGA  | eco03\_gga  |
| GP  | eco03\_gdp  |
| GT  | eco03\_gdt  |
| GC  | eco03\_gdc  |
| GCB  | eco03\_gcb  |
| GCDI  | eco03\_gcd  |
| GAG  | eco03\_gag  |
| GCV  | eco03\_gcv  |
| GGD  | eco03\_ggd  |
| GAR  | eco03\_gar  |
| GSCAV  | eco03\_gsc  |
| GAEP  | eco04\_gap  |
| CTAR  | eco04-cta  |
| CEREEM  | eco04\_cer  |
| CTARC  | eco04\_ctc  |
| CERREMC  | eco04\_cec  |
| GI  | eco05\_gdi  |
| GHP  | eco05\_ghp  |
| GSE  | eco05\_gse  |
| GARO  | eco05\_gar  |
| GCTV  | eco05\_gct  |
| GDMP  | eco05\_gdm  |
| GRVP  | eco05\_grv  |
| GCDEP  | eco05\_gcd  |
| GURP  | eco05\_gur  |
| GRAERR  | eco06\_gra  |
| GISFM  | eco06\_gis  |
| GA  | eco06\_gda  |
| GCSP  | eco06\_gcs  |
| GGVT  | eco06\_ggv  |
| GETH  | eco06\_get  |
| GPS  | eco06\_gps  |
| GES  | eco07\_ges  |
| GN  | eco07\_gdn  |
| GRC  | eco07\_grc  |
| GBSST  | eco07\_gbs  |
| GCA  | eco07\_gca  |

*b) Abreviatura de otras aplicaciones del ecosistema*

|  |  |
| --- | --- |
| Aplicación  | Abreviatura  |
| SISTEMA  | eco00\_sis  |
| SIG (Geodata)  | eco00\_geo\*  |
| ANÁLISIS CONTEXTO  | eco00\_aco  |

3. Las tablas que sirven para dar soporte a los grupos de trabajo de la UNP se definen con «eco00\_sis». Así, por ejemplo, eco00\_sis\_registro (para indicar el registro del caso) o eco00\_sis\_basicospersona (para los datos básicos de la persona que pasará de solicitante a evaluado y de evaluado a beneficiario).

4. Todas las tablas externas y no generadas por el EI, pero que son importantes para su funcionamiento se nombran con el prefijo «ext» (externas): ext\_municipio (puesto que esta información es generada por el DANE)

5. Las tablas que se utilicen en varios grupos, pero no en todos, así como necesarias para el funcionamiento del EI en alguna parte de este se definen con el prefijo «bas» después de «eco00». Así, eco00\_bas\_tipogenero para la tabla que contiene los géneros.

7. No puede existir tabla sin relación. Cada tabla debe tener como mínimo una relación con otra tabla.

8. Las tablas básicas, del sistema y de grupos que son susceptibles de actualizar información por diferentes usuarios deben siempre tener las siguientes columnas: usuario\_creacion (usuario que creo el registro o la fila), fecha\_creacion (fecha y hora en la que el usuario creo el registro), usuario\_actualizacion (usuario que actualizo el registro) y fecha\_actualizacion (fecha de la última vez que se actualizó el registro por parte de un usuario).

9. Todas las tablas anteriores (punto ocho) deben tener o estar asociadas a una tabla de historial (history). Esto se puede lograr con el uso de disparadores (triggers) que se activen cuando se realicen las acciones INSERT, UPDATE y DELETE (casos muy excepcionales con esta última acción).

10. Los nombres de las tablas deben propender a manejar solo dos guiones bajos. Rara vez y solo que el contexto lo amerite, puede manejarse más de tres guiones. Por ejemplo: eco00\_bas\_basicospersona es preferibles siempre a eco01\_bas\_basicos\_persona o eco00\_bas\_datos\_basicos\_persona.

11. Los atributos de las tablas compuestos por tres o más nombres o palabras se deben manejar de la siguiente manera: a) primera palabra completa; b) inicial de la segundo o tercera palabra; y c) última palabra completa. Si la palabra es muy extensa se puede abreviar. Así, por ejemplo, primer\_ntercero para (primer nombre de tercero, en la tabla de tercero de contacto).

12. Los atributos boléanos deben manejarse siempre como boléanos. Asimismo, al momento de nombrar deben tener un verbo rector que permita identificarlos: acepta\_mpreventivas, por ejemplo, para indicar que la persona aceptó medidas preventivas por parte de la Policía Nacional.

13. La comunicación entre las diferentes bases de datos se establecerá mediante el protocolo API (Interfaz de Programación de Aplicaciones).

Imagen 4
Estructura de las bases de datos



Fuente: Elaboración propia

00: UNP central

01: Dirección General

02: Oficina Asesora de Planeación e Información

03: secretaría general

04: Subdirección de Evaluación del Riesgo

05: Subdirección de Proteccion

06: Subdirección Especializada de Seguridad y Protección

07: Subdirección de Talento Humano

08: Oficina Asesora Jurídica

09: Oficina de Control Interno

# 9. Interfaces de Usuario: Mockups o prototipos de la interfaz

Ver documento anexo para detallar la interfaz prototipo. Es importante anotar que está en proceso de ajustes para el desarrollo del aplicativo web del Ecosistema de información.

# 10. Metodología de Desarrollo

## 10.1. Metodología: Proceso racional unificado (PRU)

Es una metodología de desarrollo de *software* orientado a objetos. Esta establece las fases, plantillas y ejemplos para todos los aspectos y fases de desarrollo del *software.* También se define como una herramienta de la ingeniería de *software* que combina los aspectos del proceso de desarrollo (fases definidas, técnicas y prácticas, etc.) con otros componentes de este (documentos, modelos, manuales, código fuente, etc.). Además, establece cuatro fases de desarrollo, cada una organizada en varias iteraciones separadas que deben satisfacer criterios definidos antes de emprender la subsiguiente fase.

Cuadro 1
Estructura del PRU en su dimensión del proceso

|  |  |
| --- | --- |
|  | **Fases** |
| **Disciplinas del proceso** | **Conceptualización** | **Elaboración** | **Construcción** | **Transición** |
| Modelación |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| Requerimientos |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| Análisis y diseño |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| Implementación |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| Testeo |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| Instalación |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

Nota: verde (prioridad alta), azul (prioridad media) y naranja claro (prioridad baja)
Fuente: Elaboración propia

Sobre esta metodología se destaca que su proceso de *software tiene tres características esenciales*: dirigido por los casos de uso, centrado en la arquitectura, y es iterativo e incremental. A continuación, se especifica aspectos relevantes de cada una de estas características.

## 10.2. Estrategias y técnicas de desarrollo:

Las principales estrategias y técnicas de desarrollo en el PRU implican la utilización de casos de uso, la arquitectura centrada y un proceso iterativo incremental. No obstante, también se contemplan la adaptación del proceso, el equilibrio de las prioridades, demostrar valor iterativamente, colaboración entre equipos y la elevación del nivel de abstracción. De momento se abordan solo las tres primeras, por su importancia y transcendencia durante el desarrollo.

## 10.3. Desarrollo dirigido por casos de uso:

Es una técnica de capturas de requisitos que fuerzan a pensar en términos de importancia para el usuario y no solo en términos de las funciones que se deben considerar. Un caso de uso se define como un fragmento de funcionalidad del sistema que proporciona al usuario un valor añadido, por lo que representan los requerimientos funcionales del sistema.

Cuando los casos de uso se enmarcan en el PRU especifican, además de los requisitos del sistema, la guía de su diseño, implementación y prueba. Así, constituyen un elemento integrador y una guía de trabajo según se observa en la imagen sobre este tema.

En ese orden, cumplen la función de definir la transacción entre los distintos artefactos generados en las diferentes actividades del proceso de desarrollo. Con base en los casos de uso se crean los modelos de análisis y diseño, posteriormente se genera la implementación que los lleva a cabo y ayuda a verificar la adecuada implementación de cada caso de uso en el proceso final.

Imagen 1
Integración del trabajo con PRU a partir de los casos de uso



Fuente: Elaboración propia

## 10.4. Centrado en la arquitectura:

La arquitectura de un sistema es la organización o estructura de sus partes más relevantes, lo que permite tener una visión común entre todos los involucrados (desarrolladores y usuarios), así como una perspectiva clara del sistema completo, necesaria para controlar el desarrollo. La arquitectura involucra los aspectos estáticos y dinámicos más significativos del sistema y está relacionada con la toma de decisiones que indican la forma de construcción de este. La arquitectura se ve influenciada por la plataforma software, sistema operativo, gestor de bases de datos, protocolos, consideraciones de desarrollo como sistemas heredados. Muchas de estas restricciones constituyen requisitos no funcionales del sistema.

Como el PRU presta atención al establecimiento temprano de una buena arquitectura que no se vea fuertemente impactada ante cambios posteriores durante la construcción y el mantenimiento, enfatiza asimismo en la interacción entre los casos de uso y la arquitectura. Los casos de uso deben encajar en la arquitectura cuando se llevan a cabo; y la arquitectura debe permitir el desarrollo de todos los casos de uso requeridos en la actualidad y el futuro. Esto provoca que tanto arquitectura como casos de uso deban evolucionar en paralelo durante todo el proceso de desarrollo de *software*.

## 10.5. Proceso iterativo e incremental:

El PRU propone tener un proceso iterativo e incremental en donde el trabajo se divide en partes más pequeñas, permitiendo generar un equilibrio entre casos de uso y arquitectura. Cada parte más pequeña se puede ver como una iteración (un recorrido más o menos completo a lo largo de todos los flujos de trabajo fundamentales) del cual se obtiene un incremento que produce un crecimiento en el producto.

Una iteración puede realizarse por medio de una cascada, como se muestra en la imagen a continuación. Esta iteración pasa por los flujos fundamentales (requisitos, análisis, diseño, implementación y pruebas). Existe igualmente una planificación de la iteración un análisis de la iteración y algunas actividades específicas de la iteración. Al finalizar se realiza una integración de los resultados con lo obtenido de las iteraciones anteriores. El proceso iterativo e incremental consta de una secuencia de iteraciones, que se pueden agrupar en uno o más casos de uso. Cada iteración aborda una parte de la funcionalidad total, pasando por todos los flujos de trabajo relevantes y refinando la arquitectura. Cada iteración se analiza cuando termina. Se puede determinar si han aparecido nuevos requisitos o han cambiado los existentes, afectando a las iteraciones siguientes. Durante la planificación de los detalles de la siguiente iteración, el equipo también examina cómo afectarán los riesgos que aún quedan al trabajo en curso. Toda la retroalimentación de la iteración pasada permite reajustar los objetivos para las siguientes iteraciones. Se continúa con esta dinámica hasta que se haya finalizado por completo con la versión actual del producto.

Imagen 2
Iteración del PRU



Fuente: Elaboración propia

## Ciclo de Implementación.



# Gestión de Riesgos

|  |  |  |
| --- | --- | --- |
| **Riesgo Identificado** | **Evaluación y Probabilidad** | **Plan de Mitigación y Contingencias** |
| Ciberataques | Alta probabilidad debido a la sensibilidad de la información manejada. Impacto alto. | Implementar medidas de seguridad cibernética como firewalls, sistemas de detección de intrusos, cifrado de datos y autenticación multifactorial. Realizar auditorías de seguridad periódicas. |
| Fugas de Información | Probabilidad media-alta. Impacto alto. | Establecer políticas de acceso restringido y monitoreo de actividad. Capacitar a los empleados en prácticas de seguridad de la información. Implementar herramientas de prevención de pérdida de datos (DLP). |
| Acceso no autorizado | Probabilidad media. Impacto alto. | Utilizar controles de acceso basados en roles (RBAC), autenticación multifactorial y sistemas de registro y monitoreo de accesos. |
| Violación de la Privacidad | Probabilidad media. Impacto alto. | Cumplir con las leyes y regulaciones de protección de datos. Implementar políticas de privacidad estrictas y realizar evaluaciones de impacto sobre la privacidad. |
| Fallas del Sistema | Probabilidad media. Impacto alto. | Implementar redundancia en la infraestructura y sistemas de respaldo. Realizar pruebas de continuidad del negocio y recuperación ante desastres. |
| Obsolescencia Tecnológica | Probabilidad media. Impacto medio. | Mantener un programa de actualización tecnológica y planificar inversiones en nuevas tecnologías. Realizar evaluaciones periódicas de la infraestructura tecnológica. |
| Errores Humanos | Probabilidad alta. Impacto medio. | Capacitar continuamente al personal en el uso del sistema. Implementar procedimientos de verificación y auditoría para minimizar errores. |
| Capacitación Inadecuada | Probabilidad media-alta. Impacto alto. | Desarrollar y mantener programas de capacitación continuos y actualizados. Evaluar regularmente la competencia del personal, observar los resultados |
| Interoperabilidad | Probabilidad media. Impacto medio. | Establecer estándares y protocolos de interoperabilidad. Trabajar estrechamente con proveedores y socios para asegurar la compatibilidad y funcionalidad. |
| Resistencia al Cambio | Probabilidad alta. Impacto medio. | Desarrollar un plan de gestión del cambio que incluya la comunicación efectiva, la capacitación y la participación de los empleados en el proceso de implementación. |
| Modificar la información sin autorización  | Probabilidad media-alta. Impacto alto. | Desarrollar controles de acceso y autenticación multifactorial, sistemas de monitoreo y detección de intrusos, políticas de seguridad claras y capacitación continua. |
| Falla en suministro eléctrico y/o internet | Probabilidad media-alta. Impacto alto. | Para mitigar el riesgo de fallas en el suministro eléctrico y/o internet, se deben instalar sistemas de alimentación ininterrumpida (UPS) y generadores de respaldo para asegurar el funcionamiento continuo del sistema durante interrupciones eléctricas. |

# 13.Documentación

Durante todo el proceso, se debe desarrollar documentación técnica y de usuario para todos los componentes del sistema, lo cual es esencial para la gestión del conocimiento y facilita el mantenimiento del software. Esta documentación incluye la definición de requerimientos, especificaciones del sistema, detalles de cada módulo, prueba y mantenimiento, utilizando herramientas de gestión de configuración para su elaboración.

Para el desarrollo del código, la documentación es crucial ya que detalla las operaciones del sistema, permitiendo su reproducción o adaptación para mantenimiento, y garantiza la transferencia de conocimiento dentro del equipo. Los manuales de usuario final facilitan el uso del sistema, mejoran la capacitación, reducen el tiempo de entrenamiento y sirven como referencia para administradores y personal de soporte, detallando cómo operarlo.

## Manual del usuario.

El manual del usuario se encontrará dentro del ECOSISTEMA, incorporado en cada fase y perfil según corresponda.

|  |  |
| --- | --- |
| CARÁCTER | DESCRIPCIÓN |
| Fecha y versión | Incluye la versión actual del documento y la fecha en que se actualizó. Esto es crucial para asegurar que los usuarios y desarrolladores están utilizando la información más reciente. |
| Pre-requisitos de Instalación | El ecosistema de Información detalla todos los requisitos necesarios antes de instalar el sistema. Esto incluye el sistema operativo compatible, los navegadores soportados, configuraciones de seguridad necesarias y cualquier otro software o hardware que deba estar presente.  |
| Manual del Usuario | Ofrece instrucciones detalladas sobre cómo utilizar las principales funciones y opciones del sistema. Debe incluir capturas el flujo del proceso de para cada paso, ayudando a los usuarios a entender mejor la ruta. |
| Preguntas Frecuentes. | Contiene una lista de las preguntas más comunes que los usuarios pueden tener, junto con respuestas claras y precisas. Esta sección ayuda a resolver problemas comunes rápidamente sin necesidad de asistencia adicional. |

## Documentación técnica

El manual técnico debe contemplar los siguientes atributos.

|  |  |
| --- | --- |
| CARÁCTER | DESCRIPCIÓN |
| Pre\_rrequisitos | Sistema operativo de los servidores de aplicaciones y base de datos, marca y versión de la base de datos, marca y versión de los servidores de aplicaciones, navegador, configuraciones de seguridad, etc. |
| Frameworks | Nombres y versiones de los frameworks y estándares bajo los cuales está construido el sistema |
| Diagramas de casos de uso  | Presenta un diagrama detallado que muestra los diferentes casos de uso del sistema. Este diagrama ilustra las interacciones entre los usuarios (actores) y el sistema, destacando las funcionalidades principales y cómo se utilizan. Ayuda a comprender los requisitos funcionales y las expectativas de los usuarios. |
| Diagrama ERD | Muestra el modelo entidad-relación del sistema, destacando las entidades principales y sus relaciones. Este diagrama es fundamental para entender la estructura de la base de datos, facilitando el diseño, implementación y mantenimiento del sistema de gestión de datos. |
| Diccionario de Datos | Proporciona una lista detallada de los elementos de datos utilizados en el sistema, incluyendo nombres, tipos de datos, tamaños, restricciones y una breve descripción de cada uno. Este diccionario sirve como una referencia crucial para desarrolladores y analistas de datos. |
| Script de Instalación  | Contiene los scripts necesarios para instalar el sistema, incluyendo instrucciones paso a paso y comandos específicos. Estos scripts automatizan el proceso de instalación, asegurando que todos los componentes se configuren correctamente y minimizando errores manuales. |
| Diagrama de Componentes | Muestra un diagrama que describe los componentes principales del sistema y sus interacciones. Este diagrama ayuda a visualizar la arquitectura del sistema, identificando cómo los diferentes módulos se conectan y colaboran para proporcionar las funcionalidades del sistema. |

## Guías de instalación y administración

# Mantenimiento y Soporte

El mantenimiento y soporte son dos aspectos fundamentales para el Ecosistema de Información de la UNP, debido a que con ambos se garantizar su funcionamiento óptimo y la satisfacción del usuario final.

**Mantenimiento correctivo**: se realiza con el fin de reparar fallos o problemas que surgen durante el uso normal de las aplicaciones y herramientas del Ecosistema. Su objetivo es restaurar el funcionamiento normal lo más rápidamente posible.

**Mantenimiento preventivo:** consiste en acciones planificadas para evitar problemas futuros. Incluye actividades como revisiones periódicas, actualizaciones de software, y ajustes preventivos que pueden reducir la probabilidad de fallos.

**Soporte Técnico:** implica proporcionar asistencia directa a los usuarios para resolver problemas técnicos. Puede incluir desde la resolución de incidencias hasta la instalación y configuración inicial del usuario para optimizar el uso de los sistemas, incluyendo sus aplicaciones y herramientas. Es un componente clave en la satisfacción del usuario final y la operación eficiente.

**Soporte de usuario:** se refiere a brindar ayuda y orientación a los usuarios sobre cómo utilizar de manera efectiva las aplicaciones y herramientas del Ecosistema de Información. Esto puede incluir formación, guías de usuario y soporte para preguntas frecuentes.

Ecosistema de Información. También incluye la prevención de futuros fallos y la capacitación de Plan de mantenimiento.

Un plan de mantenimiento del Ecosistema de Información es crucial para garantizar que las aplicaciones y sistemas informáticos operen de manera eficiente, segura y confiable a lo largo del tiempo. Aquí alguna estructura básica que se puede adaptar.

* 1. **Evaluación Inicial:** incluye un análisis detallado del software y sus componentes para identificar áreas de mejora y posibles problemas.
	2. **Inventario de Recursos:** documentar todos los recursos asociados con el software, incluyendo hardware, licencias de software, y sistemas operativos compatibles.
	3. **Definición de Objetivos:** establecer objetivos claros para el mantenimiento, como mejorar la estabilidad, seguridad, y rendimiento del software.
	4. **Frecuencia de Mantenimiento:** determinar la frecuencia con la que se llevarán a cabo las actividades de mantenimiento preventivo y correctivo.
	5. **Actualizaciones de Software:** implementar regularmente actualizaciones de seguridad y versiones del software para mantenerlo actualizado.
	6. **Optimización de Rendimiento:** realizar ajustes y optimizaciones para mejorar el rendimiento del sistema.
	7. **Limpieza de Datos:** verificar y limpiar bases de datos y archivos temporales para evitar problemas de rendimiento.
	8. **Backup y Recuperación:** realizar copias de seguridad periódicas y pruebas de recuperación para proteger contra la pérdida de datos.
	9. **Gestión de Incidentes:** responder rápidamente a problemas y errores reportados por usuarios o detectados durante las pruebas.
	10. **Control de Versiones:** mantener un sistema de control de versiones para gestionar cambios en el software de manera ordenada.
	11. **Pruebas de Aceptación:** realizar pruebas exhaustivas después de implementar cambios para asegurar que el software funcione correctamente.

## Estrategia de soporte

Una estrategia de soporte eficaz es fundamental para asegurar que los usuarios puedan utilizar y mantener los productos o servicios de manera óptima. Aquí se presentan algunos de los elementos claves para desarrollar una estrategia de soporte efectiva:

**Definición de Objetivos y Metas:**

* 1. *Claridad en los Objetivos:* definir claramente los objetivos del equipo de soporte, como la satisfacción del cliente, tiempos de respuesta rápidos, resolución eficiente de problemas, etc.
	2. *Metas Medibles:* establecer metas cuantificables para evaluar el desempeño del equipo de soporte, como tiempos de resolución, índice de satisfacción del cliente, etc.

**Segmentación de Soporte:**

* 1. *Priorización de Casos:* implementar un sistema de priorización para gestionar casos según su urgencia e impacto en los usuarios.
	2. Canalización de Solicitudes: definir y promover canales de comunicación efectivos para recibir y gestionar solicitudes de soporte, como correo electrónico, sistemas de tiques, chat en vivo, etc.

**Personal y Recursos:**

* 1. *Capacitación del Personal:* proporcionar formación continua al personal de soporte técnico en productos, procedimientos y habilidades de comunicación.
	2. *Asignación de Recursos:* asegurar los recursos adecuados, como herramientas de soporte, software de gestión de tiques, acceso a bases de conocimiento, etc.

**Proceso de Soporte:**

* 1. *Flujos de Trabajo Claros:* establecer flujos de trabajo y procedimientos estándar para la gestión de solicitudes, desde la recepción hasta la resolución y cierre.
	2. *Escalabilidad:* Diseñar el proceso de soporte de manera que sea escalable para manejar aumentos en la demanda o problemas complejos.

**Comunicación y Transparencia:**

* 1. *Comunicación Proactiva:* mantener a los usuarios informados sobre el estado de sus solicitudes y cualquier problema conocido.
	2. *Feedback del Cliente:* solicitar retroalimentación regular de los usuarios para identificar áreas de mejora en el servicio de soporte.

**Mejora Continua:**

* 1. *Análisis y Evaluación:* realizar análisis periódicos del desempeño del equipo de soporte y del proceso, utilizando métricas clave para identificar áreas de mejora.
	2. *Iteración y Adaptación:* ajustar la estrategia de soporte según los resultados obtenidos y las necesidades cambiantes del negocio y los usuarios.

**Gestión de Conocimiento:**

* 1. *Base de Conocimientos:* desarrollar y mantener una base de conocimientos actualizada con soluciones a problemas comunes, guías de usuario, y otra documentación relevante.
	2. *Autoservicio:* promover el autoservicio proporcionando a los usuarios acceso fácil a la base de conocimientos y recursos de ayuda.

**Cultura de Servicio**

* 1. *Orientación al Cliente:* fomentar una cultura organizacional que ponga al usuario en el centro, con un enfoque en la resolución rápida y efectiva de problemas.
	2. *Reconocimiento y Motivación:* reconocer y premiar el buen desempeño del equipo de soporte para mantener altos niveles de motivación y compromiso.

## Procedimientos de actualización y parches.

Los procedimientos de actualización y parches mantienen la seguridad y el rendimiento de los sistemas y software. A continuación, un conjunto de procedimientos recomendados:

**Gestión de Versiones y Parches:**

1. *Identificación de Necesidades:* mantener un registro de las actualizaciones disponibles y evaluar si son necesarias según las mejoras de seguridad, funcionalidad o compatibilidad que ofrecen.
2. *Planificación de* Actualizaciones*:* establecer un calendario para aplicar actualizaciones de manera regular, considerando períodos de baja actividad para minimizar el impacto en los usuarios.
3. *Priorización:* clasificar las actualizaciones según su criticidad (críticas, importantes, opcionales) y aplica primero las críticas que abordan vulnerabilidades conocidas.

**Pruebas Previa a la Implementación:**

1. *Entorno de Pruebas:* realizar pruebas exhaustivas de las actualizaciones en un entorno de prueba que simule el ambiente de producción para verificar la compatibilidad y estabilidad.
2. *Escenarios de Uso:* simular diferentes escenarios de uso del software para asegurar que las actualizaciones no introduzcan nuevos problemas o conflictos.
3. *Registro de Pruebas:* documentar los resultados de las pruebas realizadas, incluyendo cualquier problema detectado y las acciones correctivas tomadas.

**Implementación de Actualizaciones:**

1. *Programación y Notificación:* informar a los usuarios y equipos afectados sobre la fecha y hora de implementación de las actualizaciones planificadas.
2. *Proceso de Implementación:* seguir un procedimiento estructurado para aplicar las actualizaciones, que puede incluir la copia de seguridad de datos críticos antes de proceder.
3. *Monitorización en Tiempo Real:* supervisar la implementación en tiempo real para identificar cualquier problema inesperado y tomar medidas correctivas de inmediato.

**Gestión de Parches de Emergencia:**

1. *Respuesta Rápida:* establecer un protocolo para manejar parches críticos de emergencia, como los que abordan vulnerabilidades de seguridad graves.
2. Validación y Prueba acelerada: acelerar el proceso de prueba y validación de parches críticos para implementarlos lo antes posible sin comprometer la estabilidad del sistema.
3. *Comunicación Urgente:* comunicar de manera urgente y clara a los usuarios afectados sobre la necesidad y el impacto de los parches de emergencia.

**Evaluación Post-implementación:**

1. *Monitoreo continuo:* realizar monitoreo continuo después de la implementación para identificar cualquier efecto secundario o problema surgido como resultado de las actualizaciones.
2. *Retroalimentación de Usuarios:* solicitar retroalimentación de los usuarios sobre su experiencia con las actualizaciones para ajustar futuros procesos de implementación.
3. *Revisión y Mejora Continua:* revisar regularmente los procedimientos de actualización y parches para incorporar aprendizajes y mejoras basadas en las experiencias previas.

# Impacto del Proyecto

## Impacto esperado en la organización/comunidad.

El ecosistema de información implica la interacción e integración entre personas, tecnologías, infraestructuras y herramientas digitales para procesar, aprovechar y utilizar datos. Uno de sus objetivos trasversales es mejorar la toma de decisiones basada en información no solo disponible, sino también estructurada, verídica y confiable. Además, busca operar en tiempo real y a partir de flujos continuos de datos. En el marco de la norma ISO 27001, se debe garantizar la integridad y seguridad de los datos.

En el anexo correspondiente a este apartado, en una tabla de Excel, se encuentran en detalle las actividades principales, los indicadores, descriptores y metas relacionados con los resultados que se presentan a continuación.

## Beneficios a corto y largo plazo.

**2.1.1. Corto Plazo (entre tres y seis meses)**

Desarrollar e implementar la aplicación web de servicio al ciudadano y de asesor poblacional para facilitar la solicitud de protección en su ruta individual y colectiva.

Desarrollar e implementar la aplicación web de la Subdirección Especializada de Seguridad y Protección (SESP) en sus módulos de asesoría poblacional, asignación, análisis del riesgo, pre-mesa y subcomisión técnica, tanto en su ruta individual como colectiva.

Desarrollar e implementar la aplicación web de la Subdirección de Evaluación del Riesgo (SER) en sus módulos de asignaciones, análisis de riesgo y Cerrem, tanto en su ruta individual y colectiva.

Contar con la infraestructura básica para el desarrollo, las pruebas y el despliegue de las herramientas tecnológicas y aplicativos producidos durante los primeros tres meses.

**2.1.2. Mediano Plazo (entre siete y doce meses)**

implementación Fase 4 y 5 del desarrollo total.

**2.1.3. Largo Plazo (entre trece y veinte meses)**

Disponer de este desarrollo total por encima del 90% de las misionales de la Unidad Nacional de Protección