



Procedimiento

Desarrollo de software



Unidad de Proyección Normativa
y Estudios de Regulación Financiera

Contenido

| | | |
|-----|---|----|
| 1. | Objetivo..... | 2 |
| 2. | Alcance..... | 2 |
| 3. | Productos esperados..... | 2 |
| 4. | Condiciones especiales del procedimiento | 2 |
| 4.1 | Identificación de Necesidades | 2 |
| 4.2 | Backlog Único (Scrum + Kanban) | 3 |
| 4.3 | Planificación (Scrum)..... | 3 |
| 4.4 | Ejecución (Scrum + Kanban) | 4 |
| 4.5 | Liberación y Soporte..... | 4 |
| 4.6 | Mejora Continua..... | 5 |
| 5. | Términos y definiciones..... | 5 |
| 6. | Definición del procedimiento | 9 |
| 7. | Documento referente | 16 |
| 8. | Datos de elaboración y control de cambios | 16 |

1. Objetivo

Desarrollar, implementar y mantener soluciones tecnológicas eficientes para la Unidad Administrativa Especial, Unidad de Proyección Normativa y Estudios de Regulación Financiera (URF), mediante un esquema híbrido (Scrum + Kanban) que integre priorización, desarrollo, pruebas, liberación y soporte, para avanzar en la transformación digital de la Entidad.

2. Alcance

Inicia con la recepción y registro de una necesidad tecnológica o de un incidente relacionado con fallas en los desarrollos de software. Cubre el análisis, desarrollo, mejora, mantenimiento y soporte de sistemas y soluciones tecnológicas. Finaliza con la liberación de una solución validada, el cierre formal de la solicitud o su descarte.

3. Productos esperados

Como resultado de la aplicación del presente procedimiento, se espera obtener:

- Solución tecnológica desarrollada, mejorada o ajustada, conforme a las necesidades del servicio.
- Backlog único actualizado y trazable, que integre funcionalidades nuevas, incidencias, mejoras y deuda técnica, con su respectiva priorización y control de estado.
- Gestión controlada de incidentes y cambios con registro, clasificación, atención y retroalimentación al backlog.

4. Condiciones especiales del procedimiento

Las siguientes condiciones especiales soportan la correcta ejecución del procedimiento bajo el marco ágil híbrido (Scrum + Kanban). Estas disposiciones definen los parámetros, límites operativos y criterios mínimos que deben cumplirse para asegurar la trazabilidad, control, calidad y sostenibilidad del proceso de desarrollo, liberación y soporte del sistema.

4.1 Identificación de Necesidades

- El procedimiento aplica a desarrollos, mejoras, soporte e incidentes relacionados con los sistemas administrados por el equipo técnico de la URF.

Los requerimientos tecnológicos que cumplan con esta característica deben presentarse formalmente haciendo uso de los formatos definidos para tal efecto, al correo electrónico de gsi-urf@urf.gov.co, con la siguiente información mínima:

- Descripción clara de la necesidad
 - Alcance esperado
 - Datos de contacto del solicitante
- La priorización de necesidades de soluciones tecnológicas se realizará con base en criterios de infraestructura y arquitectura tecnológica disponible.
 - Los cambios de alcance mayores o que impliquen impacto presupuestal o estratégico deben contar con aprobación de la instancia directiva correspondiente.

4.2 Backlog Único (Scrum + Kanban)

- Todas las solicitudes (funcionalidades nuevas, ajustes, bugs e incidentes) deben consolidarse en un único Backlog administrado por el Product Owner.
- Los incidentes en producción deben:
 - Registrarse mediante correo electrónico al buzón gsi-urf@urf.gov.co
 - Clasificarse según su criticidad
 - Priorizarse dentro del Backlog

4.3 Planificación (Scrum)

- Sólo se debe iniciar el desarrollo de funcionalidades previamente revisadas por el Product Owner.
- Ninguna historia de usuario podrá ingresar a un Sprint si no cumple con la definición de listo, es decir:
 - Criterios de aceptación definidos.
 - Alcance delimitado.
 - Revisión técnica preliminar por parte del profesional especializado (Ingeniero) del proceso de Gestión de la Información.

- Las funcionalidades nuevas se desarrollarán bajo modalidad Scrum mediante Sprints de 2–3 semanas.

4.4 Ejecución (Scrum + Kanban)

- Los incidentes, errores y ajustes menores se deben gestionar mediante flujo continuo Kanban.
- Se deben respetar los límites de trabajo en proceso (WIP limits) definidos en el tablero interno
- Los documentos técnicos y funcionales deben contener la siguiente información mínima:
 - Arquitectura
 - Base de datos
 - Manual técnico
 - Manual de usuario

4.5 Liberación y Soporte

- Los despliegues de producción deben contar con lo siguiente:
 - Validación previa en ambiente de pruebas o staging (cuando exista)
 - Checklist de liberación
 - Plan documentado de reversión (rollback)
- La liberación de funcionalidades debe contar con las siguientes condiciones:
 - Ejecución de pruebas funcionales por parte del QA
 - Coordinación y ejecución de UAT cuando aplique
 - Validación formal del Product Owner
 - Cumplimiento de la Definition of Done

4.6 Mejora Continua

- Las lecciones aprendidas deben documentarse en el instrumento de lecciones aprendidas dispuesto en el SMGI e incorporarse en la planeación de las siguientes vigencias, según su priorización.
- El scrum máster debe realizar retroalimentación al finalizar cada sprint para identificar oportunidades de mejora y presentar resultados del desarrollo de software en el Comité de tecnología y seguridad de la información.

5. Términos y definiciones

- **Arquitectura de software:** es la forma en que está estructurado el sistema (componentes, bases de datos, servicios, etc.).
- **Backlog:** es la lista organizada de tareas, mejoras o requerimientos pendientes por desarrollar en un sistema.
- **Bugs:** son errores del sistema.
- **Bus factor:** es el riesgo de que solo una persona tenga todo el conocimiento de un sistema.
- **Casos de uso:** son descripciones detalladas de cómo los usuarios interactúan con el sistema.
- **Criterios de aceptación:** son las condiciones que debe cumplir una funcionalidad para considerarse terminada y aprobada.
- **Dailies:** son reuniones cortas diarias para revisar avances y bloqueos.
- **Definition of Done:** (definición de terminado) son los requisitos que debe cumplir una tarea para considerarse finalizada.
- **Definition of Ready:** (definición de listo) son las condiciones que debe cumplir una tarea antes de comenzar a desarrollarse.
- **Deuda técnica:** son mejoras o ajustes pendientes en el sistema que se dejaron para después.

- **Despliegue:** es el proceso de publicar el sistema o una nueva versión para que los usuarios la utilicen.
- **Diccionario de datos:** es el documento que describe cada campo o dato del sistema y su significado.
- **Historias de usuario:** son descripciones simples de lo que un usuario necesita que el sistema haga y para qué lo necesita.
- **Incidentes:** son fallas o errores reportados en el sistema en uso.
- **Incremento:** es una mejora o funcionalidad nueva que se agrega al sistema en cada ciclo de desarrollo.
- **Kanban:** es un método visual de gestión del trabajo mediante un tablero con tareas en diferentes estados (pendiente, en proceso, terminado) que ayuda a controlar el flujo y evitar sobrecarga.
- **Lead time:** es el tiempo total desde que se solicita una tarea hasta que se entrega terminada.
- **Marco ágil:** es una forma moderna de gestionar proyectos que permite trabajar por etapas cortas, con entregas frecuentes y mejoras continuas.
- **Pipeline:** es una secuencia automatizada de pasos para construir, probar y publicar software.
- **Pipelines de CI/CD:** son procesos automáticos que permiten construir, probar y publicar el sistema.
- **Planning:** es la reunión de planificación donde se define qué tareas se trabajarán en el sprint.
- **Producción:** es el ambiente real donde el sistema es utilizado por los usuarios.
- **Product Backlog:** es la lista priorizada de funcionalidades y mejoras que se desean implementar en el sistema.

- **Product Owner:** (responsable del producto) es la persona encargada de definir qué se va a desarrollar, priorizar tareas y asegurar que el sistema cumpla con las necesidades del negocio.
- **QA (Quality Assurance):** (aseguramiento de calidad) es el proceso de verificar que el software funcione correctamente y cumpla con los requisitos definidos.
- **README:** es un documento inicial que explica cómo funciona y cómo usar un proyecto.
- **Reléase:** es una versión del sistema que se entrega a los usuarios con nuevas funcionalidades o mejoras.
- **Requerimientos funcionales:** son las funciones que el sistema debe realizar.
- **Requerimientos no funcionales:** son características de calidad del sistema como seguridad, velocidad o disponibilidad.
- **Retrospectiva:** es la reunión para analizar qué se hizo bien, qué mejorar y cómo optimizar el trabajo.
- **Review:** es la reunión donde se muestran los avances del sistema a los interesados.
- **Roadmap (hoja de ruta):** es una representación visual o plan que muestra la dirección, objetivos y etapas principales de un proyecto o producto a lo largo del tiempo.
- **Rollback:** es regresar a una versión anterior del sistema si ocurre un problema.
- **Scrum:** es una metodología ágil que organiza el trabajo en ciclos cortos llamados sprints para entregar mejoras continuas de un producto.
- **Scrum Master:** (facilitador del proceso ágil) es quien organiza las reuniones, elimina bloqueos y ayuda al equipo a trabajar de forma eficiente.

- **SDLC (Software Development Life Cycle):** (ciclo de vida del desarrollo de software) es el proceso completo desde la idea del sistema hasta su uso y mantenimiento.
- **Software:** es el conjunto de programas e instrucciones que permiten al hardware ejecutar tareas específicas, gestionar datos y facilitar la interacción del usuario en dispositivos electrónicos.
- **Sprint:** es un periodo corto de trabajo (generalmente de 2 a 3 semanas) en el que se desarrollan funcionalidades específicas.
- **Staging:** es un ambiente de pruebas previo a producción.
- **Testing:** (pruebas) es el proceso de revisar el sistema para detectar errores antes de usarlo en producción.
- **Throughput:** es la cantidad de tareas completadas en un periodo de tiempo.
- **UAT (User Acceptance Testing):** (pruebas de aceptación de usuario) es cuando los usuarios finales prueban el sistema para confirmar que funciona según sus necesidades antes de ponerlo en uso.
- **WIP limits (Work In Progress limits):** (límites de trabajo en proceso) es el número máximo de tareas que se pueden trabajar al mismo tiempo para evitar sobrecarga.

6. Definición del procedimiento

| Actividad | Periodicidad | Tiempo estimado actividad | PC | Responsable | Descripción | Salidas | |
|-----------------------------------|-----------------------------|---------------------------|---------|-------------|--|--|---|
| a) Identificar necesidades | | | | | | | |
| 1 | Registrar necesidad | Según la necesidad | 3 horas | | Líder de proceso | Registrar formalmente la necesidad en la herramienta definida para tal fin, incluyendo descripción, alcance y datos del solicitante. | Formato Épica funcional |
| 2 | Levantar requerimiento | Según la necesidad | 5 horas | | Profesional especializado grado 14 – Proceso Gestión de la Información/ Profesional especializado grado 22 (Ingeniero) – Proceso Gestión de la Información | Realizar reunión con el proceso solicitante, para identificar procesos afectados y el proceso de Gestión de la Información documenta historias de usuario con criterios de aceptación. | Historia de usuario diligenciada |
| 3 | Analizar viabilidad técnica | Según la necesidad | 6 horas | | Profesional especializado grado 22 (Ingeniero) – Proceso Gestión de la Información | Analizar la viabilidad técnica del requerimiento, el impacto en la arquitectura y riesgos del sistema y remitir concepto técnico al Coordinador del Grupo de | Concepto técnico mediante correo electrónico dirigido al Scrum master |

| Actividad | Periodicidad | Tiempo estimado actividad | PC | Responsable | Descripción | Salidas | |
|-----------------------------------|-------------------------------|---------------------------|---------|-------------|---|--|---------------------|
| | | | | | Gestión de la Información y Relacionamento con el Ciudadano | | |
| 4 | Definir prioridad estratégica | Según la necesidad | 2 horas | | Comité de tecnología y seguridad de la información | Evaluar alineación de la solicitud con roadmap institucional | Prioridad definida |
| 5 | Aprobar ingreso al backlog | Según la necesidad | 2 horas | | Comité de tecnología y seguridad de la información | Aprobar ingreso al backlog. | Backlog actualizado |
| b) Gestionar backlog único | | | | | | | |
| 5 | Registrar incidente | Según la necesidad | 4 horas | | Profesional especializado grado 22 – Proceso de Gestión de la Información (Ingeniero) | Registrar historias de usuario, incidentes o mejoras en el tablero único institucional. | Backlog actualizado |
| 6 | Priorizar backlog | Según la necesidad | 2 horas | | Profesional especializado grado 22 – Proceso de Gestión de la | Ordenar elementos según valor de negocio, urgencia, impacto institucional y dependencias técnicas. | Backlog priorizado |

| Actividad | Periodicidad | Tiempo estimado actividad | PC | Responsable | Descripción | Salidas |
|----------------------|-------------------------------|---------------------------|----------|------------------------|--|---|
| | | | | Información/ Asesor | | |
| 7 | Clasificar incidente | Según la necesidad | 4 horas | | Profesional especializado grado 22 – Proceso de Gestión de la Información (Ingeniero) | Evaluar severidad y categorizar incidentes (crítico, alto, medio, bajo). Reporte herramienta de registro de incidentes |
| 8 | Controlar límites WIP limits | Quincenal | 1 hora | | Asesor | Verificar que el tablero Kanban cumpla con los límites de trabajo en proceso definidos Limites controlados |
| c) Planificar | | | | | | |
| 9 | Realizar planeación de Sprint | Mensual | 5 horas | | Líder de proceso / Profesional especializado grado 22 (Ingeniero) – Proceso de Gestión de la Información | Definir objetivo del Sprint y seleccionar historias según capacidad disponible. Plan de Sprint definido |
| d) Ejecutar | | | | | | |
| 10 | Desarrollar funcionalidad | Según la necesidad | 60 horas | | Profesional especializado grado 22 (Ingeniero) – Proceso de | Desarrollar la funcionalidad, mediante la codificación de historias de usuario, la ejecución de pruebas Requerimiento desarrollado |

| Actividad | Periodicidad | Tiempo estimado actividad | PC | Responsable | Descripción | Salidas | |
|--------------------------------------|---------------------------|---------------------------|----------|-------------|---|--|-----------------------|
| | | | | | Gestión de la Información | unitarias y el registro de cambios en el repositorio. | |
| 11 | Revisar calidad técnica | Según la necesidad | 16 horas | | Profesional especializado grado 22 (Ingeniero) – Proceso de Gestión de la Información | Verificar estándares de código, arquitectura y seguridad antes de las pruebas. | Código validado |
| 12 | Revisar avances | Según la necesidad | 1 hora | | Asesor | Revisar avances, bloqueos y coordinación del trabajo con los integrantes del proceso de Gestión de la Información. En caso de existir bloqueos, se debe proceder con la actividad 13. | Listado de asistencia |
| 13 | Gestionar bloqueo | Según la necesidad | 3 horas | | Asesor | Gestionar con otros procesos para eliminar impedimentos. | Bloqueo resuelto |
| e) Realizar pruebas y validar | | | | | | | |
| 15 | Ejecutar prueba funcional | Según la necesidad | 12 horas | | Profesional especializado grado 22 – Proceso de Gestión de la Información | Validar cumplimiento de criterios de aceptación. | Reporte de pruebas |

| Actividad | | Periodicidad | Tiempo estimado actividad | PC | Responsable | Descripción | Salidas |
|-----------|---|--------------------|---------------------------|----|--|---|-------------------|
| 16 | Ejecutar prueba técnica | Según la necesidad | 8 horas | | Profesional especializado grado 22 (Ingeniero) – Proceso gestión de información | Ejecutar pruebas unitarias e integración. | Evidencia técnica |
| 17 | Realizar pruebas de aceptación de usuario | Según la necesidad | 12 horas | | Profesional especializado grado 22 – Proceso gestión de información / Líder de proceso | Realizar pruebas con usuarios finales y registrar su aceptación. Si una vez realizadas las pruebas, el usuario manifiesta que no se cubrió el requerimiento, se debe regresar a la actividad 5 para revisar la solicitud inicial y en su defecto continuar con la actividad 18. Cuando el usuario acepte que se cubrió el requerimiento, se debe continuar con la actividad 19. | Acta UAT |
| 18 | Solicitar incremento funcional | Según la necesidad | 6 horas | | Líder de proceso | Solicitar incremento funcional. Una vez gestionado el incremento funcional, se | Acta UAT |

| Actividad | Periodicidad | Tiempo estimado actividad | PC | Responsable | Descripción | Salidas | |
|------------------------------|------------------------------------|---------------------------|----------|-------------|---|--|--|
| | | | | | debe proceder con la actividad 19 | | |
| f) Liberar y soportar | | | | | | | |
| 19 | Preparar despliegue | Según la necesidad | 8 horas | | Profesional especializado grado 22 (Ingeniero) – Proceso de Gestión de la Información | Preparar versión y ejecutar checklist de liberación. | Versión preparada |
| 20 | Desplegar versión | Según la necesidad | 4 horas | | Profesional especializado grado 22 (Ingeniero) – Proceso de Gestión de la Información | Implementar versión en ambiente de producción. | Versión desplegada en ambiente de producción |
| 21 | Monitorear estabilidad del sistema | Según la necesidad | 24 horas | | Profesional especializado grado 22 (Ingeniero) – Proceso de Gestión de la Información | Monitorear funcionamiento del sistema y detectar incidentes. | Registro de monitoreo |
| 22 | Gestionar incidente en producción | Según la necesidad | 48 horas | | Profesional especializado grado 22 (Ingeniero) – Proceso de | Analizar y resolver incidentes reportados. | Incidente cerrado |



| Actividad | Periodicidad | Tiempo estimado actividad | PC | Responsable | Descripción | Salidas | |
|---------------------------------------|-----------------------------|---------------------------|---------|-------------|---|--|--|
| | | | | | Gestión de la Información | | |
| g) Implementar mejora continua | | | | | | | |
| 23 | Realizar revisión de Sprint | Al finalizar cada Sprint | 2 horas | | Profesional especializado grado 22 (Ingeniero) – Proceso de Gestión de la Información | Presentar incremento funcional y recibir retroalimentación. | Retroalimentación registrada en la herramienta de incidentes |
| 24 | Analizar métricas | Mensual | 4 horas | | Asesor | Evaluar lead time, throughput y cumplimiento de compromisos. | Reporte de métricas |
| Fin del procedimiento | | | | | | | |

7. Documento referente

| Tipo | Nombre |
|-------------|--|
| Plan | Plan Estratégico de Tecnologías de la Información - PETI |

8. Datos de elaboración y control de cambios

| Control de cambios | | | |
|--------------------|---------|----------------|-------------------------|
| Fecha | Versión | Cód. Solicitud | Descripción del cambio |
| 2026-04-10 | 1 | TS-1148 | Creación del documento. |

| Elaboración, revisión y aprobación | |
|------------------------------------|--|
| Elaboración | |
| Nombre: | Camila Andrea Sánchez Amaya |
| Cargo: | Profesional especializado |
| Nombre: | Franklin González Sierra |
| Cargo: | Profesional especializado |
| Revisión | |
| Nombre: | Juan Stiven Rios Andrade |
| Cargo: | Asesor |
| Aprobación | |
| Nombre: | Paola Patricia Rodríguez Angulo |
| Cargo: | Subdirectora Jurídica y de Gestión Institucional |