**IF7100 - INGENIERÍA DE SOFTWARE**

## Plan de Gestión de la Configuración del Software

**Proyecto: IntelTaskPJ – Sistema de Gestión de Tareas y Permisos para el Poder Judicial**

**Fecha: 27/06/2025**

**Autor: Gabriela Valverde T.**

**Versión: 1.0**

**1. Introducción**

**1.1. Propósito**

Establecer los procedimientos, herramientas y responsabilidades para controlar los elementos de configuración del sistema IntelTaskPJ. Esto incluye el manejo de versiones, control de cambios, auditorías y documentación asociada al ciclo de vida del software.

**1.2. Alcance**

Este plan aplica a todos los artefactos del proyecto IntelTaskPJ: código fuente, scripts de base de datos, documentación técnica, prototipos, reportes y cualquier componente que forme parte del producto final entregado.

**2. Elementos de Configuración**

| **Elemento de Configuración** | **Descripción** | **Responsable** |
| --- | --- | --- |
| Código fuente | Archivos del backend (.NET) y frontend (Angular) | Desarrolladora |
| Script de base de datos | Archivo SQL con estructura y datos de configuración inicial | Desarrolladora |
| Documentación de requerimientos | Documento con todos los RF del sistema | Analista de requerimientos |
| Prototipos | Vistas de pantallas, diseño en Bootstrap | Desarrolladora |
| Planes de prueba | Casos de prueba, checklist y evidencia de validación | Tester |

**3. Gestión de Versiones**

**3.1. Herramientas de Control de Versiones**

* **GitHub**: Repositorio centralizado para código, historial de versiones y colaboración.
* **Google Drive / Notion**: Almacenamiento de documentación, control de cambios en archivos Word y PDF.

**3.2. Numeración de Versiones**

Se utilizará el formato Versión Mayor.Menor.Patch, por ejemplo:

* **1.0.0** – Primera versión estable
* **1.1.0** – Nueva funcionalidad agregada
* **1.1.1** – Corrección menor o ajuste no funcional

**4. Control de Cambios**

**4.1. Procedimiento de Solicitud de Cambio**

* Solicitud a través del formulario de cambio.
* Evaluación por el Comité de Cambios (en este caso, liderado por la gerente del proyecto).
* Aprobación documentada en el sistema de control (planificación, checklist de cambios).
* Aprobación documentada en el sistema de control.

#### 4.2. Evaluación de Cambios

* Impacto en cronograma.
* Recursos requeridos.
* Riesgos asociados.
* Aprobación formal por parte del cliente.

**5. Gestión de Configuraciones**

**5.1. Identificación de Configuraciones**

Para garantizar que el sistema IntelTaskPJ pueda ejecutarse correctamente en diferentes entornos, se identifican los siguientes elementos de configuración críticos:

| **Elemento** | **Descripción** |
| --- | --- |
| appsettings.json | Archivo de configuración de variables de entorno (cadena de conexión, JWT, etc.). |
| Startup.cs / Program.cs | Configuración de servicios, middleware y políticas de seguridad. |
| Script script.sql | Script con la estructura y datos iniciales de la base de datos. |
| Archivo .csproj | Archivo del proyecto .NET que contiene referencias a librerías y SDK. |
| angular.json y package.json | Configuraciones del frontend Angular (build, dependencias, scripts). |
| .env o archivos de entorno local | Variables sensibles para entornos locales, como claves privadas o rutas locales. |

Además, se definen convenciones para mantener la uniformidad:

* **Prefijos de campos en la BD:**
  + CT\_ para texto
  + CN\_ para numérico
  + CF\_ para fechas
  + CB\_ para booleanos
* **Nombres de carpetas frontend/backend:**
  + BackEnd/ (proyecto .NET Core)
  + FrontEnd/ (proyecto Angular)

**5.2. Auditoría de Configuraciones**

* **Frecuencia:** Al finalizar cada fase del ciclo de vida del proyecto (Diseño, Desarrollo, Pruebas).
* **Responsable:** Gerente del Proyecto (Gabriela Valverde T.)
* **Procedimiento:**
  1. Verificación de que todos los archivos necesarios están presentes y versionados.
  2. Validación de que las rutas de base de datos, APIs y claves estén correctamente referenciadas.
  3. Revisión de carpetas del repositorio, builds y estructura de archivos.

**5.3. Control de Versiones y Baselines**

* Se establecerá una **baseline** al final de cada fase importante (por ejemplo, cuando se completa el módulo de tareas, o al terminar pruebas).
* Cada baseline incluirá:
  + Código estable en GitHub, etiquetado con tag (ej. v1.0.0-alpha).
  + Documentación correspondiente (requerimientos, pruebas, pantallas).
  + Script actualizado de base de datos.
  + Configuración de entorno funcional para pruebas o despliegue.

**6. Roles y Responsabilidades**

| **Rol** | **Nombre** | **Responsabilidades** |
| --- | --- | --- |
| Gerente de Proyecto | Gabriela Valverde T. | Aprobar cambios, organizar entregas y coordinar auditorías de configuración. |
| Desarrolladora Principal | Gaby Valverde | Gestionar versiones del código, scripts de base de datos y asegurar configuraciones consistentes. |
| Analista de Requerimientos | Gaby Valverde | Asegurar trazabilidad de requisitos, versiones de documentos y artefactos relacionados. |
| Tester | Jordy Mendoza | Validar configuraciones en los entornos de pruebas y mantener evidencia de ejecución. |

**7. Comunicación**

**7.1. Reuniones de Configuración**

| **Tipo de Reunión** | **Frecuencia** | **Participantes** | **Objetivo** |
| --- | --- | --- | --- |
| Revisión de configuración | Al finalizar cada fase | Gerente de proyecto, desarrolladora | Verificar integridad de versiones, estructura del repositorio y documentación. |
| Coordinación técnica | Según necesidad (ad hoc) | Desarrolladora, tester | Alinear criterios de pruebas, versiones activas y configuración del entorno. |
| Validación previa a entregas | Antes de entregar una versión | Todos los integrantes | Confirmar que todos los elementos estén listos y configurados correctamente. |

**7.2. Informes de Configuración**

| **Informe** | **Contenido** | **Responsable** | **Medio de Entrega** |
| --- | --- | --- | --- |
| Checklist de configuración | Verifica elementos críticos: script, credenciales, estructura | Desarrolladora | Compartido en Google Drive |
| Reporte de versiones liberadas | Versión entregada, cambios aplicados, artefactos incluidos | Gerente de Proyecto | Documento en Notion / Word |
| Bitácora de cambios en configuraciones | Registro de cambios técnicos importantes en estructura o settings | Desarrolladora / Analista | Google Sheets compartido |

**8. Herramientas y Entornos**

**8.1. Herramientas Utilizadas**

| **Herramienta** | **Uso en el Proyecto** |
| --- | --- |
| **GitHub** | Control de versiones del código fuente, revisión de cambios y colaboración del equipo. |
| **Visual Studio 2022** | Desarrollo del backend en .NET Core. |
| **Visual Studio Code** | Edición de archivos de configuración y scripts adicionales. |
| **Angular CLI** | Generación y administración del proyecto frontend. |
| **SSMS** | Gestión de la base de datos, ejecución de scripts y revisión de estructura. |
| **Trello** | Seguimiento de tareas y entregas por fase. |

**8.2. Entornos de Desarrollo, Pruebas e Implantación**

| **Entorno** | **Descripción** |
| --- | --- |
| **Desarrollo** | Ambiente local en cada computadora del equipo. Incluye servidor local de backend, Angular y BD local o en la nube. |
| **Pruebas** | Ejecutado en navegador, con roles simulados y datos reales. Usa scripts y configuraciones predefinidas. |
| **Implantación (demo)** | Versión ejecutable para exposición final, corriendo localmente o en servidor temporal proporcionado por el profesor. |

### 10. Aprobación

**Gerente de Proyecto:**

**Cliente/Parte Interesada:**