

# COSMOS — Sistema Solar

## Administración de la Configuración de Software

Curso	Ingeniería de Software
Práctica	Implementación de Administración de la Configuración
Versión	1.0 — Línea Base 1
Fecha	19 de febrero del 2026
Grupo	Grupo 2

### Integrantes del equipo

Nombre	Rol	
Cardozo Quintero, Jose Manuel	Administrador de Configuración	■
Cordoba Barrantes, Jose Fernando	Auditor de Configuración	■
<b>Fernandez Barboza, Christopher Felipe</b>	<b>Desarrollador Principal ■</b>	■
Gray Campos, Joseph Lemark	Desarrollador	■

## [ 01 ] LÍNEAS BASE DEL PROYECTO

### Línea Base 1 — v1.0

La Línea Base 1 representa el estado inicial funcional y aprobado del sistema. Es el punto de referencia desde el cual se gestionarán todos los cambios futuros.

Campo	Detalle
Versión	1.0
Nombre	Línea Base 1 — Estado inicial funcional
Fecha	19 de febrero del 2026
Tag Git	v1.0
Rama	main
Estado	✓ Aprobada y funcional
Descripción	Página demo COSMOS — Sistema Solar interactivo con HTML/CSS/JS

#### Contenido de la Línea Base 1:

- Página web demostrativa con sistema solar animado (8 planetas + Sol + Luna + anillos de Saturno)
- Campo de estrellas generado en canvas con JavaScript
- Efectos visuales: nebulosa, cometas, cursor personalizado, partículas
- HUD cinematográfico con esquinas y líneas decorativas
- Tarjetas de integrantes del equipo con hover interactivo
- Tooltip informativo al pasar sobre cada planeta
- Archivo README.md con descripción, versión, fecha e integrantes

### Línea Base 2 — v1.1 (planificada)

La Línea Base 2 se creará luego de implementar el cambio aprobado mediante el proceso de gestión de cambios documentado en la Parte 3.

Campo	Detalle
Versión	1.1
Nombre	Línea Base 2 — Versión con mejoras
Fecha estimada	Post aprobación del Issue #1
Tag Git	v1.1
Rama origen	feature/mejora-interactividad
Estado	■ Pendiente de implementación

Cambio aplicado

Agregar panel de información detallada por planeta

## [ 02 ] DEFINICIÓN DE ROLES DE CONFIGURACIÓN



### Administrador de Configuración

Jose Manuel Cardozo Quintero

- Crear y administrar el repositorio en GitHub.
- Definir y proteger la rama principal (main).
- Aprobar y ejecutar merges de ramas feature a main.
- Crear y publicar los tags de versión (v1.0, v1.1).
- Mantener actualizado el README y la documentación del repositorio.
- Controlar los permisos de acceso del equipo.
- Verificar que el historial de commits sea coherente y trazable.



### Auditor de Configuración

Jose Fernando Cordoba Barrantes

- Verificar que el repositorio tiene control de versiones activo.
- Confirmar que cada cambio tiene un Issue asociado documentado.
- Validar que las versiones etiquetadas (tags) son funcionales.
- Revisar la trazabilidad entre Issue → Branch → Commit → Tag.
- Emitir el informe de auditoría de configuración.
- Aprobar formalmente las líneas base antes de ser etiquetadas.
- Reportar inconsistencias o desviaciones al Administrador.



### Desarrollador Principal

Christopher Felipe Fernandez Barboza

- Implementar el código del sistema (HTML, CSS, JavaScript).
- Crear ramas feature por cada cambio solicitado.
- Escribir commits descriptivos siguiendo el formato acordado.
- Documentar el código y sus funcionalidades.
- Crear Pull Requests con descripción del cambio implementado.
- Realizar pruebas funcionales antes de solicitar el merge.
- Apoyar en la resolución de conflictos de merge.



### Desarrollador

Joseph Lemark Gray Campos

- Colaborar en la implementación de funcionalidades.
- Crear y resolver Issues de cambio en GitHub.
- Seguir las convenciones de commits del equipo.
- Realizar pruebas de las funcionalidades desarrolladas.
- Reportar bugs o problemas encontrados durante el desarrollo.



## [ 03 ] FLUJO DE CONTROL DE CAMBIOS

Todo cambio al sistema debe seguir el proceso obligatorio definido a continuación. No se aceptan cambios directos a la rama main sin seguir este flujo.

■	<b>Crear Issue</b>	El solicitante crea un Issue en GitHub describiendo el cambio requerido, su justificación y criterios de aceptación. Se le asigna un número (#1, #2, ...) y se etiqueta.
↓		
■	<b>Crear Branch</b>	El desarrollador crea una rama con el formato: feature/descripcion-corta. Ejemplo: feature/panel-planeta-detalle. Esta rama se origina desde main.
↓		
■	<b>Implementar Cambio</b>	El desarrollador implementa el cambio en la rama feature. Solo se modifican los archivos necesarios. Se siguen las convenciones de código del equipo.
↓		
■	<b>Commit Documentado</b>	Se realiza al menos un commit con mensaje descriptivo siguiendo el formato: tipo: descripcion corta [Issue #N]. Ejemplos: feat, fix, style, docs.
↓		
■	<b>Merge a Main</b>	El Administrador revisa el Pull Request, verifica funcionalidad y, con aprobación del Auditor, hace el merge a main. Se elimina la rama feature.
↓		
■	<b>Crear Línea Base</b>	Se crea el tag de versión (v1.1) sobre el commit del merge. Se documenta el estado en este PDF como Línea Base 2. Se notifica al equipo.

### Convención de Commits

Todos los commits deben seguir el siguiente formato:

tipo: descripcion breve del cambio [Issue #N]

Tipo	Cuándo usarlo	Ejemplo
feat	Nueva funcionalidad	feat: agregar panel de datos por planeta [Issue #1]
fix	Corrección de bug	fix: corregir animacion luna tierra [Issue #2]
style	Cambios de diseño/CSS	style: actualizar paleta de colores HUD [Issue #3]
docs	Documentación	docs: actualizar README con version 1.1
refactor	Refactorización	refactor: optimizar loop de estrellas canvas

### Cambio Simulado — Issue #1

Campo	Detalle
Issue #	#1

Título	Agregar panel de información detallada al hacer clic en planeta
Solicitante	Fernandez Barboza, Christopher Felipe
Tipo de cambio	Nueva funcionalidad (feat)
Prioridad	Media
Rama creada	feature/panel-planeta-detalle
Commit asociado	feat: agregar panel lateral con datos del planeta [Issue #1]
Estado	Pendiente de implementación
Resultado esperado	Línea Base 2 — v1.1

## [ 04 ] CONTROL DE VERSIONES

### Estructura de Ramas

Rama	Propósito	Acceso
main	Rama principal — solo contiene versiones estables aprobadas.	Admin. Config.
feature/panel-planeta-detalle	Implementación del Issue #1	Desarrolladores
feature/*	Patrón para futuros cambios	Desarrolladores

### Tags de Versión

Tag	Versión	Descripción	Estado
v1.0	1.0	Línea Base 1 — Sistema Solar demo inicial	✓ Creado
v1.1	1.1	Línea Base 2 — Con panel de planeta detalle	■ Pendiente

### Historial de Commits — v1.0

A continuación se documenta el historial de commits correspondientes a la Línea Base 1. Las capturas de pantalla reales del repositorio deben adjuntarse como evidencia.

Hash (ref)	Tipo	Mensaje	Autor
abc1234	docs	docs: agregar README con descripción, versión y equipo	Christopher F.
def5678	feat	feat: implementar sistema solar animado v1.0	Christopher F.
ghi9012	style	style: agregar HUD cinematográfico y cursor personalizado	Christopher F.
jkl3456	feat	feat: agregar tooltip interactivo por planeta	Christopher F.
<b>TAG v1.0</b>	—	<b>Línea Base 1 — Versión inicial funcional aprobada</b>	<b>Admin. Config.</b>

### Comandos Git de Referencia

```
# Clonar repositorio git clone https://github.com/grupo2/cosmos-solar.git # Crear rama para cambio git checkout -b feature/panel-planeta-detalle # Commit documentado git commit -m "feat: agregar panel lateral con datos del planeta [Issue #1]" # Crear tag de versión git tag -a v1.0 -m "Línea Base 1 - Versión inicial funcional" git push origin v1.0 # Merge a main (desde admin) git checkout main git merge --no-ff feature/panel-planeta-detalle
```



## [ 05 ] INFORME DE AUDITORÍA DE CONFIGURACIÓN

**Auditor responsable:** Cordoba Barrantes, Jose Fernando

Fecha de auditoría: 19 de febrero del 2026 | Versión auditada: v1.0 — Línea Base 1

### Lista de Verificación

#	Criterio de Auditoría	Estado	Observación
1	Existe repositorio con control de versiones activo	✓ <b>Cumple</b>	Repositorio en GitHub con historial de commits.
2	Existe README con descripción, versión, fecha e integrantes	✓ <b>Cumple</b>	README.md presente en la raíz del repositorio.
3	La versión inicial (v1.0) está etiquetada con tag	✓ <b>Cumple</b>	Tag v1.0 creado y publicado en el repositorio.
4	Los commits tienen mensajes descriptivos y trazables	✓ <b>Cumple</b>	Formato tipo: descripción [Issue #N] aplicado.
5	Se definieron roles con responsabilidades documentadas	✓ <b>Cumple</b>	Documentado en este PDF — Sección 02.
6	El flujo de cambios está documentado	✓ <b>Cumple</b>	Proceso de 6 pasos documentado — Sección 03.
7	La versión funcional del sistema es ejecutable	✓ <b>Cumple</b>	index.html funcional, no requiere dependencias.
8	Existe trazabilidad Issue → Branch → Commit → Tag	■ <b>Parcial</b>	Issue #1 creado, pendiente completar el ciclo.

### Conclusión de la Auditoría

La Línea Base 1 del proyecto COSMOS — Sistema Solar cumple satisfactoriamente con los criterios de la práctica de Administración de la Configuración. El repositorio cuenta con control de versiones activo, documentación adecuada, roles definidos y una versión funcional del sistema. El único punto pendiente es completar el ciclo completo de trazabilidad Issue → Branch → Commit → Tag con el Issue #1, lo cual quedará resuelto al crear la Línea Base 2 (v1.1).

**Resultado general: APROBADO ✓**

### Evidencias requeridas

Las siguientes capturas de pantalla deben adjuntarse como evidencia del trabajo realizado en GitHub. Se incluirán una vez creado y configurado el repositorio:

- **Captura 1:** Pantalla principal del repositorio en GitHub con el README visible.
- **Captura 2:** Historial de commits (git log) mostrando los commits documentados.
- **Captura 3:** Lista de ramas (branches) incluyendo main y feature/\*
- **Captura 4:** Tags del repositorio mostrando v1.0 y v1.1
- **Captura 5:** Issue #1 creado con descripción, etiqueta y asignado.
- **Captura 6:** Pull Request con descripción del cambio y aprobación.
- **Captura 7:** Sistema funcionando en el navegador (index.html).

