

Gestión de Desarrollo del Software

TRABAJO PRÁCTICO FINAL

1. INTRODUCCIÓN

El presente trabajo práctico tiene como objetivo aplicar los conceptos de gestión de proyectos de software vistos durante el cuatrimestre. Los equipos realizarán la **gestión integral** del proyecto que desarrollarán en la materia Laboratorio IV.

Importante: Este trabajo consiste en **gestionar** el proyecto de Laboratorio IV, no en desarrollar software adicional. Aplicaremos técnicas de estimación, planificación, seguimiento y control sobre el proyecto real que están construyendo.

Requisito Institucional

Este informe de gestión será solicitado por la coordinación de la carrera como parte de la documentación requerida para la presentación del "Trabajo Final Integrador" (Ex PPS). Por lo tanto, es fundamental que la documentación sea completa, profesional y refleje fielmente el proceso de gestión llevado a cabo durante el proyecto.

La calidad y completitud de este documento no solo impacta en la evaluación de esta materia, sino que también forma parte del expediente académico necesario para la acreditación del Trabajo Final Integrador.

2. CONFORMACIÓN DE GRUPOS

- Los grupos deben estar compuestos por **3 a 4 integrantes**.
- Todos los integrantes deben participar activamente en las tareas de gestión y documentación.
- Los grupos serán los mismos que en Laboratorio IV.

3. CONTEXTO DEL PROYECTO A GESTIONAR

Los equipos gestionarán uno de los siguientes proyectos (según lo definido en Laboratorio IV):

Opción 1: Desarrollo del frontend para el proyecto desarrollado en Laboratorio III.

Opción 2: Desarrollo completo de un nuevo proyecto según los contenidos de Laboratorio IV.

La elección de la opción debe ser comunicada al docente en la **Semana 1**.

4. CRONOGRAMA GENERAL

Semana 1: Presentación y Definición

- Presentación del TP Final
- Conformación definitiva de grupos
- Elección de opción (1 o 2)
- Definición inicial de requisitos funcionales

Semanas 2-3: Estimación y Planificación

- Aplicación de técnicas de medición (PFA)
- Estimación con COCOMO y técnica adicional obligatoria
- Diseño de arquitectura (C4)
- Identificación de riesgos

CHECKPOINT 1 (Fin Semana 3): Entrega de estimaciones, arquitectura y riesgos

Semanas 4-5: Ejecución y Seguimiento

- Implementación de metodología SCRUM
- Ejecución de 2 sprints
- Seguimiento mediante JIRA/similar
- Gestión de cambios

CHECKPOINT 2 (Fin Semana 5): Revisión de JIRA/similar y avance de sprints

Semana 6: Cierre

- Finalización de documentación
- Análisis de desviaciones
- Presentación final

5. ENTREGABLES OBLIGATORIOS

Los equipos deberán presentar un **documento único en formato PDF** que contenga los siguientes apartados:

5.1 Portada y Descripción del Proyecto

Contenido:

- Nombre del proyecto
- Integrantes del equipo
- Misión y alcance del proyecto
- Opción elegida (1 o 2)
- Descripción general del sistema

5.2 Especificación de Requisitos Funcionales

Objetivo: Documentar entre **5 y 8 requisitos funcionales** siguiendo esta estructura:

Para cada requisito incluir:

- **Código:** RF-01, RF-02, etc.
- **Nombre descriptivo:** Ej. "Gestión de usuarios"
- **Descripción detallada:**
 - Contexto: Relación con otros requisitos
 - Objetivo: Propósito de la funcionalidad
 - Entradas y salidas de información
 - Pasos o procesos incluidos
 - Reglas de negocio aplicables
- **Prioridad:** Alta / Media / Baja
- **Notas adicionales:** Cualquier detalle relevante

Ejemplo de formato:

RF-01: Autenticación de Usuarios

Descripción:

El sistema debe permitir que los usuarios se autenticuen mediante email y contraseña para acceder a las funcionalidades según su rol.

Contexto: Este requisito es base para RF-02 (Gestión de permisos) y RF-03 (Registro de actividad).

Objetivo: Garantizar acceso seguro y diferenciado según perfil de usuario.

Entradas:

- Email (string, formato válido)
- Contraseña (string, mínimo 8 caracteres)

Salidas:

- Token de sesión (JWT)
- Datos del usuario autenticado
- Mensaje de error en caso de credenciales inválidas

Proceso:

1. Usuario ingresa credenciales en formulario de login
2. Sistema valida formato de email y longitud de contraseña
3. Sistema consulta BD para verificar credenciales
4. Si son válidas: genera token y redirige al dashboard
5. Si son inválidas: muestra mensaje de error

Reglas de negocio:

- Máximo 3 intentos fallidos antes de bloqueo temporal (15 minutos)
- Las contraseñas se almacenan encriptadas (bcrypt)
- El token tiene vigencia de 24 horas

Prioridad: ALTA

5.3 Medición Funcional - Análisis de Puntos de Función (PFA)

Objetivo: Aplicar la técnica de Análisis de Puntos de Función sobre los requisitos definidos.

Incluir:

1. **Tabla de conteo** con las siguientes columnas:
 - Tipo de función (EI, EO, EQ, ILF, EIF)
 - Descripción
 - Complejidad (Baja/Media/Alta)
 - Puntos asignados
2. **Cálculo de Puntos de Función no ajustados**
3. **Factor de Ajuste:**
 - Listar las 14 características generales del sistema
 - Asignar valores de 0 a 5 según influencia
 - Calcular FA
4. **Puntos de Función Ajustados**

5.4 Estimaciones de Esfuerzo, Tiempo y Costo

Objetivo: Estimar el proyecto utilizando **dos técnicas de estimación obligatorias:**

Técnica Obligatoria 1: COCOMO

Incluir:

1. **Conversión de PF a KLOC** (Líneas de código)
 - Usar factor de conversión según lenguaje
 - Indicar el lenguaje del proyecto y el factor utilizado
2. **Aplicación del modelo COCOMO:**
 - Indicar tipo de proyecto (Orgánico/Semi-acoplado/Empotrado)
 - Aplicar fórmulas de COCOMO Básico
 - Calcular:
 - Esfuerzo (personas-mes)
 - Tiempo de desarrollo (meses)
 - Personas promedio necesarias
3. **Justificación:**
 - Explicar por qué eligieron ese tipo de proyecto
 - Supuestos utilizados

Técnica Obligatoria 2 (elegir una)

Opción A - Estimación por Analogía:

- Comparar con proyectos similares previos
- Ajustar según diferencias identificadas

Opción B - Estimación por Tres Puntos:

- Escenario optimista
- Escenario más probable
- Escenario pesimista
- Calcular promedio ponderado: $(O + 4M + P) / 6$

Opción C - Planning Poker:

- Documentar sesión de estimación con el equipo
- Story points asignados a cada requisito
- Conversión a horas/días

4. Estimación de Costos:

- Definir costo por hora/persona (puede ser estimado/teórico)
- Calcular costo total del proyecto
- Desglosar por etapas si es posible

5.5 Arquitectura del Sistema - Modelo C4

Objetivo: Documentar la arquitectura del sistema utilizando el modelo C4 (Context, Containers).

Nivel 1: Diagrama de Contexto

Incluir:

- El sistema como caja central
- Actores/usuarios que interactúan con el sistema
- Sistemas externos con los que se integra
- Relaciones y flujos principales

Nivel 2: Diagrama de Contenedores

Incluir:

- Componentes principales del sistema (frontend, backend, base de datos, etc.)
- Otros contenedores relevantes según el proyecto
- Tecnologías utilizadas en cada contenedor
- Protocolos de comunicación entre componentes

Acompañar cada diagrama con:

- Breve descripción textual
- Tecnologías específicas
- Responsabilidades de cada componente

Herramientas sugeridas: Draw.io, Lucidchart, diagrams.net, PlantUML

5.6 Gestión de Riesgos

Objetivo: Identificar y analizar los principales riesgos del proyecto.

Formato: Risk Register con las siguientes columnas:

ID	Riesgo	Categoría	Probabilidad	Impacto	Exposición	Estrategia	Responsable
R-01	Descripción	Técnico/Alcance/Equipo	Alta/Media/Baja	Alto/Medio/Bajo	P×I	Mitigar/Evitar/Transferir/Aceptar	Nombre

Ejemplo:

ID	Riesgo	Categoría	Probabilidad	Impacto	Exposición	Estrategia	Responsable
R-01	Problemas de integración entre frontend Angular y backend Spring Boot	Técnico	Media	Alto	Media × Alto = Alto	Mitigar: Realizar pruebas de integración tempranas. Definir contratos de API claros desde el inicio.	Juan Pérez

Incluir mínimo 5-8 riesgos que contemplen:

- **Riesgos técnicos:** Integración entre componentes, rendimiento, seguridad
- **Riesgos de alcance:** Cambios en requisitos, funcionalidades complejas
- **Riesgos de tiempo:** Retrasos, estimaciones incorrectas
- **Riesgos de equipo:** Disponibilidad de miembros, conocimiento técnico

Para cada riesgo definir:

- Estrategia de mitigación específica
- Acciones preventivas concretas
- Responsable de monitorear el riesgo

5.7 Implementación de SCRUM

Objetivo: Documentar la aplicación de la metodología ágil SCRUM durante el desarrollo.

5.7.1 Definición de Roles

Especificar claramente:

- **Scrum Master:** Nombre del integrante y responsabilidades asumidas
- **Product Owner:** Nombre del integrante y responsabilidades asumidas
- **Development Team:** Resto de los integrantes

5.7.2 Product Backlog

Incluir:

- Lista priorizada de requisitos/historias de usuario
- Estimación de cada ítem (horas, story points, o similar)
- Criterios de aceptación resumidos

5.7.3 Sprint 1 (Duración: 2 semanas aproximadamente)

Documentar:

Sprint Planning:

- Objetivo del sprint
- Historias/tareas seleccionadas del backlog
- Estimación de capacidad del equipo
- Distribución de tareas entre los miembros

Sprint Execution:

- Resumen de tareas completadas
- Captura de pantalla del tablero JIRA/similar al finalizar el sprint

Sprint Review:

- Funcionalidades completadas y demostradas
- Funcionalidades no completadas (si las hay) y razones

Sprint Retrospective:

- ¿Qué funcionó bien?
- ¿Qué se puede mejorar?
- Acciones de mejora para el próximo sprint

5.7.4 Sprint 2 (Duración: 1-2 semanas aproximadamente)

Documentar los mismos puntos que Sprint 1

5.7.5 Uso de JIRA/similar

Incluir:

- Capturas de pantalla que evidencien:
 - Product Backlog configurado
 - Tableros de cada sprint (To Do, In Progress, Done)
 - Burndown chart (si está disponible)
 - Asignación de tareas a miembros del equipo

5.8 Gestión de Cambios

Objetivo: Documentar **al menos 1 (una) Solicitud de Cambio** que haya surgido durante la ejecución del proyecto.

Formato de Solicitud de Cambio:

SOLICITUD DE CAMBIO #001

Fecha de solicitud: DD/MM/AAAA

Solicitante: [Nombre o rol]

CAMBIO PROPUESTO:

[Descripción clara y detallada del cambio solicitado]

JUSTIFICACIÓN:

[Razón por la cual se solicita este cambio]

IMPACTO EN EL PROYECTO:

1. Alcance:

- Requisitos afectados: [listar RF-XX]
- Nuevas funcionalidades requeridas: [si aplica]

2. Tiempo:

- Estimación original: [X horas/días]
- Nueva estimación: [Y horas/días]
- Impacto: +/- [diferencia]

3. Costo:

- Costo original: [\$XXX]
- Nuevo costo estimado: [\$YYY]
- Impacto: +/- [diferencia]

4. Recursos:

- Recursos adicionales necesarios: [si aplica]

5. Riesgos:

- Nuevos riesgos introducidos: [si aplica]

DECISIÓN:

☐ APROBADO ☐ RECHAZADO ☐ DIFERIDO

JUSTIFICACIÓN DE LA DECISIÓN:

[Explicar por qué se aprobó o rechazó el cambio]

Fecha de decisión: DD/MM/AAAA

Responsable de la decisión: [Nombre - Rol]

5.9 Cierre del Proyecto

Objetivo: Realizar un análisis comparativo entre lo estimado inicialmente y lo ejecutado realmente.

5.9.1 Comparación Estimaciones vs. Real

Tabla comparativa:

Métrica	Estimación Inicial	Valor Real	Desviación	% Desviación
Puntos de Función	[X PF]	[Y PF]	+/- [diferencia]	[%]
Esfuerzo (personas-mes)	[X]	[Y]	+/- [diferencia]	[%]
Tiempo (semanas)	[X]	[Y]	+/- [diferencia]	[%]
Costo total	[\$X]	[\$Y]	+/- [diferencia]	[%]
Requisitos planificados	[X]	[Y]	+/- [diferencia]	[%]
Requisitos completados	-	[Y]	-	[%]

5.9.2 Análisis de Desviaciones

Para cada desviación significativa (>20%), explicar:

- Causa raíz de la desviación
- ¿Se podría haber previsto?
- ¿Qué acciones se tomaron al detectarla?
- ¿Cómo impactó en el resultado final?

Ejemplo:

El esfuerzo real fue 40% superior al estimado debido a:

1. Subestimación de la complejidad de integración entre componentes
2. Problemas técnicos no contemplados en la estimación inicial
3. Necesidad de refactorizar módulos para adaptarlos a nuevos requisitos

Estas desviaciones se detectaron en el Sprint 1 y se ajustaron las estimaciones del Sprint

2. Se redujo el alcance eliminando RF-07 y RF-08 para cumplir con los plazos.

6. CHECKPOINTS DE CONTROL

Durante el desarrollo del TP se realizarán controles de avance en las siguientes instancias:

Checkpoint 1 - Fin de Semana 3

Entregables a revisar:

- Requisitos funcionales definidos y priorizados
- PFA completo
- Estimaciones (COCOMO + técnica adicional)
- Arquitectura C4 (niveles 1 y 2)
- Risk Register inicial

Modalidad: Revisión en clase o entrega parcial en plataforma (a definir)

Checkpoint 2 - Fin de Semana 5

Entregables a revisar:

- JIRA/similar actualizado con ambos sprints
- Sprint 1 completamente documentado
- Evidencia de avance en Sprint 2
- Solicitud de cambio documentada

Modalidad: Revisión de JIRA/similar en clase

7. ENTREGA FINAL

Fecha de Entrega

Semana 6 - Fecha exacta a confirmar en clase.

Formato de Entrega

Documento PDF único que contenga todos los apartados solicitados en la sección 5.

Nomenclatura del archivo:

TPFinal_Gestion_Grupo[Número]_[Apellidos].pdf

Ejemplo: TPFinal_Gestion_Grupo3_Fernandez_Lopez_Martinez.pdf

Medio de entrega: Plataforma educativa

Presentación Oral (Opcional/A confirmar)

Si se requiere presentación oral, tendrá las siguientes características:

- Duración: 15-20 minutos por grupo
- Contenido: Resumen ejecutivo del proyecto gestionado
- Foco en: Estimaciones, desviaciones, aprendizaje
- Todos los integrantes deben participar

8. CRITERIOS DE EVALUACIÓN

La evaluación contemplará los siguientes aspectos:

Documentación Técnica (35%)

- **PFA:** Correcta aplicación de la técnica, conteo detallado (10%)
- **Estimaciones:** COCOMO correctamente aplicado + técnica adicional bien fundamentada (10%)
- **Arquitectura C4 + Risk Register:** Diagramas claros, identificación completa de riesgos (15%)

Gestión con SCRUM (35%)

- **Definición de roles y planificación:** Roles claros, backlog priorizado (10%)
- **Ejecución y seguimiento en JIRA/similar:** Tableros actualizados, evidencia de trabajo continuo (15%)
- **Gestión de cambios:** Cambio bien documentado y analizado (10%)

Análisis y Cierre (20%)

- **Calidad del documento final:** Redacción, formato, completitud (10%)
- **Análisis de desviaciones:** Comparación estimado vs real, justificaciones sólidas (10%)

Presentación (10%)

- **Claridad expositiva:** Comunicación efectiva de decisiones de gestión
- **Capacidad de justificar:** Argumentación de elecciones realizadas

Nota: Los porcentajes son orientativos y pueden ajustarse según criterio docente.

9. CONSIDERACIONES IMPORTANTES

Coordinación con Laboratorio IV

- Este TP de Gestión se basa en el proyecto de Laboratorio IV, pero **son materias independientes**.
- La evaluación de Gestión se centra en **cómo gestionaron** el proyecto, no en el código final.
- Todas las consultas **técnicas** (Angular, Spring Boot, MySQL) deben dirigirse al docente de Laboratorio IV.

Responsabilidades de cada materia

Laboratorio IV es responsable de:

- Definir requisitos funcionales del sistema
- Enseñar las tecnologías del proyecto
- Evaluar la calidad del código y funcionalidad

Gestión de Desarrollo de Software es responsable de:

- Verificar la correcta aplicación de técnicas de gestión
- Evaluar la documentación de gestión del proyecto
- Realizar seguimiento del uso de SCRUM y JIRA/similar

Sobre el uso de herramientas

- **JIRA/similar:** Se asume que los equipos ya tienen conocimiento de la herramienta o lo adquirirán de forma autónoma.
- **Metodología SCRUM:** Se asume conocimiento previo.
- **PFA y COCOMO:** Estas técnicas serán vistas y practicadas en clase.

Trabajo en equipo

- Se espera **participación equitativa** de todos los integrantes.
- Cada integrante debe estar capacitado para explicar **cualquier aspecto** de la documentación.
- La distribución de tareas debe ser documentada en JIRA/similar.

Originalidad y fuentes

- La documentación debe ser **producción propia** del equipo.
- Se pueden consultar fuentes externas (libros, artículos, videos) pero deben citarse apropiadamente.

10. RECOMENDACIONES PARA EL TP

Gestión del tiempo

- **No dejar todo para el final:** La documentación de gestión requiere tiempo y reflexionar sobre la misma.
- **Actualizar JIRA/similar semanalmente:** Es más fácil mantenerlo al día que actualizarlo todo al final.
- **Documentar mientras se trabaja:** Es más fácil recordar decisiones y cambios si se documentan en el momento.

Estimaciones realistas

- **Sean conservadores:** Es mejor estimar de más que quedarse cortos.
- **Incluyan buffer:** Contemplan tiempo para imprevistos (20-30% adicional).
- **Ajusten si es necesario:** Si detectan desviaciones temprano, re-estimen.

Comunicación con el docente

- **Consulten dudas temprano:** No esperen a la última semana para resolver incertidumbres.
- **Participen en los checkpoints:** Son oportunidades de feedback para corregir el rumbo.
- **Aprovechen las revisiones:** Las devoluciones de los checkpoints son para mejorar la entrega final.

Calidad de la documentación

- **Redacción clara y profesional:** Revisen ortografía y gramática.
- **Diagramas legibles:** Usen herramientas profesionales, no fotos de dibujos a mano.
- **Formato consistente:** Mantengan el mismo estilo en todo el documento.
- **Referencias cruzadas:** Vinculen secciones relacionadas (ej: "ver RF-03 en sección 5.2").

11. RECURSOS DE APOYO

Bibliografía recomendada

- **Software Engineering: A Practitioner's Approach** - Roger Pressman
- **Estimating Software Costs** - Capers Jones
- **The C4 Model for Software Architecture** - Simon Brown (<https://c4model.com>)
- **Scrum Guide** - Schwaber & Sutherland (<https://scrumguides.org>)

Herramientas sugeridas

- **JIRA/similar:** Gestión de proyectos ágiles (solicitar licencia educativa gratuita)
- **Draw.io / diagrams.net:** Diagramas C4 y arquitectura
- **Excel / Google Sheets:** Cálculos de PFA y COCOMO
- **Git / GitHub:** Control de versiones (deseable)

Plantillas disponibles

El docente proporcionará (o podrá consultar):

- Plantilla de especificación de requisitos
- Plantilla de Risk Register
- Plantilla de Solicitud de Cambio
- Ejemplo de cálculo PFA
- Ejemplo de aplicación COCOMO

12. PREGUNTAS FRECUENTES

¿Podemos cambiar de opción (1 o 2) después de la Semana 1? Solo con justificación válida y aprobación del docente. No se recomienda.

¿Qué pasa si no completamos todos los requisitos en Laboratorio IV? La evaluación de Gestión se centra en **cómo gestionaron**, no en cuánto completaron. Analizar por qué no se completó es parte del cierre.

¿Podemos usar otra herramienta en lugar de JIRA/similar? No. JIRA/similar es obligatorio para garantizar uniformidad en las revisiones.

Si estimamos 30 PF pero al final fueron 50 PF, ¿eso está mal? No necesariamente. Lo importante es **analizar por qué hubo esa desviación** en el cierre del proyecto.

¿Hay penalización por entregar fuera de término? Sí. Consultar con el docente la política de entregas tardías (típicamente 1 punto menos por día de retraso).

¿Podemos reutilizar documentación de años anteriores? No. Cada proyecto es único y debe tener documentación original.

¿Los checkpoints son obligatorios? Sí. No presentarse a un checkpoint sin aviso previo puede impactar en la evaluación.

ANEXO: CHECKLIST DE ENTREGA

Antes de entregar, verificar que el documento incluya:

- ☐ Portada con nombres de todos los integrantes
 - ☐ Índice o tabla de contenidos
 - ☐ 5.1 - Misión y alcance del proyecto
 - ☐ 5.2 - Especificación de 5-8 requisitos funcionales
 - ☐ 5.3 - PFA completo con todas las tablas
 - ☐ 5.4 - COCOMO + técnica adicional de estimación
 - ☐ 5.5 - Diagramas C4 nivel 1 y 2
 - ☐ 5.6 - Risk Register con mínimo 5 riesgos
 - ☐ 5.7 - Documentación completa de SCRUM (roles, 2 sprints, JIRA/similar)
 - ☐ 5.8 - Al menos 1 solicitud de cambio documentada
 - ☐ 5.9 - Análisis de desviaciones y lecciones aprendidas
 - ☐ Capturas de pantalla de JIRA/similar incluidas
 - ☐ Diagramas claramente visibles y legibles
 - ☐ Documento revisado ortográfica y gramaticalmente (no lo descuiden)
 - ☐ Formato PDF, nomenclatura correcta del archivo
 - ☐ Todas las secciones numeradas y referenciadas correctamente
 - ☐ Prolijidad en todo momento!
-

Estimados alumnos:

Recuerden que este TP es una oportunidad para aplicar conocimientos de gestión en un contexto real. Aprovechen la experiencia para aprender no solo técnicas, sino también a trabajar en equipo y gestionar proyectos bajo presión y con recursos limitados!.