

The background is a gradient of dark blue and purple. On the left side, there are several concentric circles and a large circular scale with numerical markings from 140 to 260. Some of the circles have arrows indicating a clockwise direction. The overall aesthetic is technical and modern.

ADMINISTRACION Y GESTION DE PROYECTOS.

Iniciamos con 3 premisas disparadoras...

1- QUE ES UNA IDEA?

2- Que es un proyecto?

3- Que es la Gestión de
Proyectos?

1-¿Qué es una idea?

Una idea es el punto de partida de todo proyecto. Es una necesidad, una propuesta o una solución que busca generar valor. Ejemplo: “¿Y si hacemos una app para organizar partidos de fútbol entre amigos?”

2-¿Qué es un proyecto?

Es un conjunto de actividades **temporales y únicas** que se realizan para cumplir un objetivo específico. Tiene un inicio, un desarrollo y un cierre. Ejemplo: “Diseñar, programar y lanzar esa app en 3 meses.”

3-¿Qué es la gestión de proyectos?

Es el proceso de **planificar, organizar, ejecutar, controlar y cerrar** un proyecto de forma eficiente. Implica coordinar personas, recursos, tiempos y objetivos para lograr resultados concretos.

“¿Por qué no basta con trabajar más?”

“La productividad no depende solo del esfuerzo, sino de cómo se organiza ese esfuerzo.”

Historia del Tren Transcontinental (EE.UU., siglo XIX)

✓ Contexto inicial

- Antes de 1869, viajar de costa a costa en EE.UU. podía tomar **meses** y costar cerca de **\$1.000 dólares** (equivalente a unos \$20.000 actuales).
- Las rutas eran peligrosas: por tierra (montañas, desiertos etc.) o por mar (vía Panamá o alrededor de Sudamérica).

✓ Proyecto visionario

- En 1862, durante la Guerra Civil, el presidente **Abraham Lincoln** firmó el **Pacific Railroad Act**, que autorizó la construcción de una vía férrea que conectara el este y el oeste del país.

✓ Se asignaron dos compañías:

- **Union Pacific Railroad**: construía desde Omaha, Nebraska hacia el oeste.
- **Central Pacific Railroad**: construía desde Sacramento, California hacia el este.

Historia del Tren Transcontinental (EE.UU., siglo XIX)

✓ Desafíos técnicos y humanos

- El terreno era extremadamente difícil, especialmente para el Central Pacific, que debía atravesar la **cordillera Sierra Nevada**.
- Se usaron explosivos como **nitroglicerina** para abrir túneles en la roca.

✓ La mano de obra fue diversa:

- Central Pacific: mayoritariamente **trabajadores chinos**, que enfrentaron condiciones brutales.
- Union Pacific: **inmigrantes irlandeses** y **veteranos de guerra**.

✓ Problemas organizativos

- Al principio, se pensaba que **trabajar más horas** aumentaría la productividad.
- Sin embargo, esto llevó a:
 - Alta **accidentalidad**
 - Baja **eficiencia**
 - Falta de **planificación**

HISTORIA (“cuando solo el esfuerzo no alcanza”)



La experiencia del Transcontinental fue clave para el nacimiento de la gestión de proyectos moderna: se comprendió que **el esfuerzo sin organización no basta**.

HISTORIA



Los administradores del Proyecto se dieron cuenta que.



=



=>



Baja productividad.

Falta de Planeamiento.

Baja Eficiencia.



A PARTIR DE ESTE CONTEXTO HISTÓRICO, PODRÍAMOS
PLANTEARNOS LAS SIGUIENTES PREGUNTAS...

- 1- Qué concluimos de este proyecto sobre la importancia de planificar?
- 2- ¿Qué similitudes hay entre este proyecto y los de software actuales?
- 3- ¿Cómo influye la gestión en el éxito de un proyecto, más allá del esfuerzo?

1: “¿QUÉ CONCLUIMOS DE ESTE PROYECTO SOBRE LA IMPORTANCIA DE PLANIFICAR?”

LA PLANIFICACIÓN ES ESENCIAL PARA LA EFICIENCIA Y LA SEGURIDAD. EN EL CASO DEL TREN TRANSCONTINENTAL, AL PRINCIPIO SE PRIORIZÓ EL ESFUERZO FÍSICO (MÁS HORAS, MÁS TRABAJADORES) SIN UNA ESTRATEGIA CLARA, LO QUE DERIVÓ EN:

- 1- ACCIDENTES FRECUENTES
- 2- RETRASOS
- 3- DESPERDICIO DE RECURSOS

CUANDO SE COMENZÓ A APLICAR UNA LÓGICA ORGANIZATIVA (DIVISIÓN DE TAREAS, USO DE TECNOLOGÍA, COORDINACIÓN ENTRE EQUIPOS), LA PRODUCTIVIDAD MEJORÓ NOTABLEMENTE. ESTO DEMUESTRA QUE **TRABAJAR MÁS NO SIEMPRE SIGNIFICA TRABAJAR MEJOR.**

2: “¿QUÉ SIMILITUDES HAY ENTRE ESTE PROYECTO Y LOS DE SOFTWARE ACTUALES?”

Tren Transcontinental	Proyecto de Software
Terreno difícil	Requisitos complejos
Mano de obra diversa	Equipos multidisciplinarios
Falta de planificación inicial	Desarrollo sin análisis previo
Uso de herramientas (explosivos, rieles)	Uso de frameworks, IDEs, metodologías
Coordinación entre compañías	Integración entre áreas (Dev, QA, UX, etc.)

AMBOS REQUIEREN **GESTIÓN DE RECURSOS, TIEMPOS, RIESGOS Y COMUNICACIÓN EFECTIVA** PARA ALCANZAR SUS OBJETIVOS.

3: “¿CÓMO INFLUYE LA GESTIÓN EN EL ÉXITO DE UN PROYECTO, MÁS ALLÁ DEL ESFUERZO?”

LA GESTIÓN PERMITE:

- 1- DEFINIR OBJETIVOS CLAROS
- 2- ASIGNAR ROLES Y RESPONSABILIDADES
- 3- OPTIMIZAR RECURSOS
- 4- PREVER RIESGOS
- 5- MONITOREAR AVANCES

SIN GESTIÓN, EL ESFUERZO PUEDE DISPERSARSE, DUPLICARSE O INCLUSO VOLVERSE CONTRAPRODUCENTE. EN EL CASO DEL TREN, LA FALTA DE PLANIFICACIÓN INICIAL GENERÓ ACCIDENTES Y BAJA PRODUCTIVIDAD. EN SOFTWARE, ÉSTO PUEDE TRADUCIRSE EN BUGS, SOBRECOSTOS O PRODUCTOS QUE NO CUMPLEN CON LO ESPERADO.

ACTIVIDAD GRUPAL (simulación): “Están a cargo de la construcción del tren Transandino. Tienen trabajadores, herramientas, presupuesto limitado y terreno difícil. *¿Cómo organizarían el proyecto para evitar accidentes y mejorar la productividad?*”

CADA GRUPO DEBE:

- 1- Definir 3 acciones de planificación (acción-porque es importante-quien la ejecuta)
- 2- Identificar 2 riesgos (riesgo-posible impacto-estrategia de mitigación)
- 3- Proponer una forma de medir el avance (indicador-que mide-como se registra)

1- POSIBLES ACCIONES DE PLANIFICACIÓN:

Acción de planificación	Explicación
Dividir el proyecto en tramos geográficos	Asignar equipos distintos a cada tramo (montaña, llanura, desierto) para trabajar en paralelo y especializar tareas.
Establecer un cronograma con hitos semanales	Definir metas intermedias (ej. "completar 10 km por semana") para monitorear el progreso y ajustar recursos.
Asignar roles específicos en cada equipo	Separar funciones: excavadores, colocadores de rieles, supervisores, etc., para evitar duplicación de tareas y mejorar la eficiencia.

2- POSIBLES RIESGOS:

Riesgo

**Condiciones
climáticas
extremas**

**Falta de
comunicación
entre equipos**

Explicación

Lluvias, nieve o calor intenso pueden retrasar la obra o poner en peligro a los trabajadores.

Si los grupos que trabajan en distintos tramos no se coordinan, puede haber errores de alineación, duplicación o retrasos.

3- POSIBLES FORMAS DE MEDIR AVANCE:

Indicador	Ejemplo
Kilómetros de vía colocados por semana	Permite ver si se cumple el ritmo previsto y ajustar si hay demoras.
Cantidad de accidentes reportados	Sirve para evaluar si las condiciones de trabajo son seguras y si la planificación está funcionando.
Comparación entre avance real y cronograma previsto	Se puede usar un gráfico simple para visualizar si el proyecto está “adelantado”, “en tiempo” o “retrasado”.

Cierre de la actividad grupal (simulación)

“A lo largo de esta simulación, quedó claro que la planificación no es un lujo, sino una necesidad. Organizar tareas, prever riesgos y medir avances permite transformar el esfuerzo en resultados concretos. Así como en el siglo XIX se aprendió que trabajar más no siempre es trabajar mejor, hoy sabemos que la gestión inteligente es lo que convierte una idea en un proyecto exitoso.

La experiencia que nos deja esta actividad, nos prepara para comprender cómo ciertos pensadores sistematizaron estos aprendizajes, dando origen a la gestión moderna de proyectos.”

DE LA EXPERIENCIA AL MÉTODO...

Lo que podemos concluir:

- 1- Planificar evita retrasos y mejora la productividad.
- 2- Medir el avance permite ajustar recursos y tiempos.
- 3- Coordinar equipos es clave para lograr resultados.

Finalmente...

¿Cómo se transformó esta lógica en una disciplina formal? ¿Quiénes fueron los primeros en estudiar el trabajo como un sistema organizado?

- **Frederick Taylor** – Padre de la administración científica
- **Henry Gantt** – Creador del diagrama de Gantt

“La gestión de proyectos moderna nace cuando el esfuerzo se estudia, se organiza y se optimiza”.



DESCANSO.

HISTORIA

Frederick Taylor.

“Lo importante no es trabajar más duro, sino trabajar de manera más inteligente.”

Henry Gantt.

Estudió el orden de las tareas de construcción creando el diagrama de Gantt

1. Tareas
2. Secuencia
3. Duración
4. Avance.

FREDERICK TAYLOR

1) ¿Qué propone Taylor?:

- Estudiar científicamente el trabajo: No dejar que cada obrero trabaje “como le parezca”, sino que un **equipo especializado analice cómo se realiza cada tarea**.
- Crear una **oficina de métodos**: Un área dentro de la empresa que se encargue de definir los **mejores procedimientos para cada tarea**.
- Definir el “**método óptimo**”: Establecer cuál es la **forma más eficiente de hacer cada trabajo**, con base en estudios de tiempos y movimientos.
- Determinar una “**norma de producción**”: **Cuánto trabajo puede hacer un obrero en condiciones ideales**.
- **Premiar el rendimiento**: Si el obrero alcanza esa norma, debe recibir un **salario elevado como incentivo**.

FREDERICK TAYLOR

2) ¿Qué propone Taylor?:

- Que **el proceso de selección** de trabajadores no sea aleatorio ni intuitivo, sino **científico y sistemático**.
- Que **se evalúen las aptitudes individuales** para asignar a cada persona el puesto más adecuado.
- Que **se proporcione entrenamiento específico** para que el trabajador pueda desempeñarse de forma óptima.
- Que **todos los trabajadores tienen potencial**, si se los ubica en el rol correcto y se los capacita adecuadamente.

¿Qué rompe con la lógica anterior?

- Antes, se asumía que el trabajador debía adaptarse al puesto. Taylor invierte esa lógica: **el puesto debe adaptarse al trabajador**, siempre que se lo entrene correctamente.

FREDERICK TAYLOR

Preguntémonos antes de seguir...

1. ¿Qué ventajas tiene este enfoque en comparación con la asignación “por necesidad”?
2. ¿Cómo se aplica hoy en áreas como desarrollo de software o gestión de proyectos?
3. ¿Qué riesgos puede tener esta visión si se aplica de forma rígida?

1- ¿Qué ventajas tiene este enfoque en comparación con la asignación “por necesidad”?

- **Mayor eficiencia:** Al estudiar científicamente las tareas, se reduce el desperdicio de tiempo y recursos.
- **Mejor aprovechamiento del talento:** Se asigna a cada persona al puesto más adecuado según sus aptitudes, no según urgencias o disponibilidad.
- **Capacitación dirigida:** El entrenamiento específico permite que el trabajador alcance su máximo rendimiento.
- **Motivación por reconocimiento:** El sistema de incentivos premia el esfuerzo y los resultados, lo que puede aumentar el compromiso.
- **Reducción de errores:** Al definir métodos óptimos, se estandariza el trabajo y se minimizan las improvisaciones.

2- ¿Cómo se aplica hoy en áreas como desarrollo de software o gestión de proyectos?

- **Roles definidos por perfil técnico y habilidades blandas:** Por ejemplo, no todos los programadores son buenos para testing o análisis funcional; se asignan tareas según fortalezas.
- **Metodologías ágiles:** Aunque no son tayloristas, también buscan eficiencia mediante roles claros, mejora continua y adaptación.
- **Oficinas de procesos o PMO:** Cumplen funciones similares a la “oficina de métodos”, definiendo estándares, buenas prácticas y herramientas.
- **Capacitación continua:** Se invierte en formación técnica y metodológica para mejorar el rendimiento del equipo.
- **Evaluación de desempeño y métricas:** Se usan KPIs, velocity, burndown charts, etc., para medir productividad y ajustar procesos.

3- ¿Qué riesgos puede tener esta visión si se aplica de forma rígida?

- **Deshumanización del trabajo:** Si se ve al trabajador solo como una pieza del sistema, se pierde la dimensión creativa y emocional.
- **Falta de flexibilidad:** En entornos dinámicos como el software, los métodos óptimos pueden cambiar rápidamente; aplicar reglas fijas puede ser contraproducente.
- **Desmotivación por exceso de control:** Si todo está normado y medido, puede generar estrés o sensación de vigilancia constante.
- **Ignorar la colaboración:** El enfoque taylorista es más individualista; en proyectos modernos, el trabajo en equipo y la comunicación son clave.
- **Resistencia al cambio:** Si se impone un método sin participación del equipo, puede haber rechazo o baja adopción.

FREDERICK TAYLOR

“Estas reflexiones nos permiten entender que, aunque el enfoque de Taylor propone mejoras claras, su éxito depende de algo más profundo: la voluntad de aplicarlo correctamente. Y ahí es donde aparecen los verdaderos desafíos...”

FREDERICK TAYLOR

3) ¿Qué plantea Taylor?:

- Que el **verdadero cambio** ocurre cuando **los obreros aplican los métodos científicos** en su trabajo diario.
- Que el **fracaso** en la implementación **no suele ser culpa** de los trabajadores, sino **de los dirigentes que se resisten al cambio**.
- Que **muchos jefes o supervisores no abandonan sus métodos tradicionales**, lo que impide que la ciencia del trabajo se traduzca en mejoras reales.

¿Qué implica la unión de: ?

1. **Estudiar científicamente el trabajo:** analizar tareas, tiempos, movimientos, herramientas.
2. **Seleccionar científicamente al trabajador:** **ubicarlo en el puesto adecuado y entrenarlo.**

Rta: Lograr que el trabajador capacitado ejecute el método optimizado. **Solo cuando el trabajador adecuado aplica el método adecuado**, se alcanza el ideal de productividad científica.

FREDERICK TAYLOR

Preguntémonos antes de seguir...

1. ¿Por qué creemos que los dirigentes se resisten a cambiar sus métodos?
2. ¿Qué ejemplos actuales conocemos donde la resistencia al cambio impide mejoras?
3. ¿Cómo se puede facilitar la aplicación de métodos científicos en equipos de trabajo?

1- ¿Por qué creemos que los dirigentes se resisten a cambiar sus métodos?

Zona de confort: Aplican métodos que “siempre funcionaron” y temen que el cambio implique pérdida de control o resultados inciertos.

Falta de formación técnica: Algunos líderes no comprenden los beneficios de los métodos científicos o los ven como “tecnicismos innecesarios”.

Miedo a la pérdida de autoridad: Cambiar el método puede implicar delegar decisiones, escuchar al equipo o aceptar errores previos.

Inercia organizacional: Las estructuras rígidas y burocráticas dificultan la adopción de nuevas prácticas, aunque sean más eficientes.

Costos percibidos: Temen que capacitar, medir, analizar y rediseñar procesos implique tiempo y dinero que no están dispuestos a invertir.

2- ¿Qué ejemplos actuales conocemos donde la resistencia al cambio impide mejoras?

Ejemplo	Resistencia	Consecuencias
Escuelas que no incorporan tecnología educativa	Temor a perder el control del aula o desconocimiento técnico	Desaprovechan herramientas que podrían mejorar la participación y el aprendizaje
Empresas que no adoptan metodologías ágiles	Preferencia por jerarquías rígidas y planificación tradicional	Lentitud en la entrega de productos, baja adaptabilidad
Hospitales que no digitalizan sus procesos	Miedo al error tecnológico o pérdida de registros	Ineficiencia en turnos, diagnósticos y atención
Administraciones públicas que no usan datos para tomar decisiones	Cultura burocrática y desconfianza en lo técnico	Políticas poco efectivas y falta de transparencia

3- ¿Cómo se puede facilitar la aplicación de métodos científicos en equipos de trabajo?

- **Formación y sensibilización:** Mostrar con ejemplos concretos cómo los métodos científicos mejoran resultados (estudios de caso, simulaciones, comparativas).
- **Participación activa:** Involucrar al equipo en el diseño del método. Si sienten que lo construyen, lo adoptan con más compromiso.
- **Pequeñas victorias:** Aplicar mejoras en áreas puntuales y mostrar resultados medibles. Esto genera confianza en el enfoque.
- **Liderazgo ejemplar:** Los dirigentes deben aplicar primero el método en su propia gestión. El ejemplo arrastra más que la orden.
- **Feedback continuo:** Evaluar, ajustar y mejorar el método con base en datos y experiencias reales. Esto refuerza la lógica científica.

FREDERICK TAYLOR

4) ¿Qué propone Taylor?:

- Que el trabajo no debe recaer exclusivamente en los obreros ni en los dirigentes.
- Que debe haber una **división equilibrada de tareas y responsabilidades** entre ambos niveles.
- Los **dirigentes** deben planificar, analizar, definir métodos y supervisar.
- Los **obreros** deben ejecutar las tareas siguiendo los métodos definidos, pero también aportar desde su experiencia práctica.
- La **cooperación** entre ambos es esencial para que el sistema funcione.

¿Qué rompe con la lógica anterior?

- En modelos anteriores, los obreros eran vistos como simples ejecutores. Taylor propone que haya una **interdependencia**: cada parte tiene un rol clave y deben trabajar en conjunto.

FREDERICK TAYLOR

Preguntémonos antes de seguir...

- 1- ¿Qué ventajas tiene este modelo de cooperación?
- 2- ¿Cómo se aplica hoy en equipos de desarrollo o gestión?
- 3- ¿Qué pasa cuando la dirección no acompaña o los equipos no se sienten escuchados?

1- ¿Qué ventajas tiene este modelo de cooperación?

La propuesta de **Taylor rompe con la lógica verticalista** y plantea una **colaboración técnica y operativa** entre dirigentes y obreros. Las ventajas son:

Mayor eficiencia: Al dividir tareas según capacidades (planificación vs ejecución), se optimiza el uso de recursos.

Mejor calidad del trabajo: Los obreros aportan desde su experiencia práctica, lo que permite ajustar los métodos a la realidad.

Reducción de conflictos: La cooperación reemplaza la imposición, generando mayor compromiso y menos resistencia.

Aprendizaje organizacional: El intercambio entre niveles permite mejorar continuamente los procesos.

Empoderamiento técnico: Los dirigentes dejan de ser meros jefes y se convierten en analistas y facilitadores del trabajo.

2- ¿Cómo se aplica hoy en equipos de desarrollo o gestión?

En contextos actuales, especialmente en tecnología y gestión de proyectos, esta lógica se traduce en prácticas como:

- **Scrum y metodologías ágiles:** El Product Owner (dirigente) define objetivos y prioridades, pero el equipo técnico decide cómo implementarlos.
- **DevOps:** La integración entre desarrollo y operaciones implica que todos colaboran en la mejora continua del producto.
- **Gestión participativa:** Los líderes facilitan procesos, pero escuchan activamente al equipo para ajustar métodos y herramientas.
- **Reuniones de retrospectiva:** Se analiza lo que funcionó y lo que no, con aportes de todos los niveles.

3- ¿Qué pasa cuando la dirección no acompaña o los equipos no se sienten escuchados?

Cuando se rompe la lógica de cooperación, aparecen consecuencias negativas:

Desmotivación: Los equipos sienten que su experiencia no vale, lo que reduce el compromiso.

Ineficiencia: Los métodos definidos por la dirección pueden estar desconectados de la realidad operativa.

Resistencia al cambio: Si no hay diálogo, cualquier mejora se percibe como imposición.

Rotación de personal: Los profesionales buscan entornos donde puedan aportar y crecer.

Estancamiento organizacional: Sin retroalimentación, los procesos no evolucionan.

Ejemplo actual: Empresas que imponen herramientas sin consultar al equipo técnico (como cambiar de Jira a otra plataforma sin capacitación ni consulta). El resultado suele ser caos y rechazo.

FREDERICK TAYLOR – PRINCIPIOS DE LA ADMINISTRACIÓN CIENTÍFICA

1. Estudio científico del trabajo

Analizar cada tarea para definir el método más eficiente. Crear una oficina de métodos que determine procesos óptimos y metas de producción.

2. Selección científica y entrenamiento

Ubicar a cada trabajador en el puesto adecuado según sus aptitudes, y capacitarlo para alcanzar su máximo rendimiento.

3. Aplicación práctica de la ciencia

Unir el estudio del trabajo con la selección del personal. El fracaso no es del obrero, sino del dirigente que no aplica los métodos científicos.

4. Cooperación entre dirigentes y obreros

Dividir el trabajo y la responsabilidad de forma equilibrada. La planificación y la ejecución deben estar conectadas por la colaboración.

Taylor no solo buscaba eficiencia, proponía un sistema donde el conocimiento técnico y la acción práctica se integran para mejorar el trabajo.



DESCANSO.

HENRY GANTT

- Henry Laurence Gantt (1861–1919) fue un ingeniero mecánico y consultor en gestión industrial.
- Colaboró con Frederick Taylor en el desarrollo de la administración científica, pero luego se enfocó en mejorar la planificación y el control del trabajo.
- Su principal aporte fue el diseño de una herramienta visual que permitiera ver el progreso de las tareas en el tiempo, facilitando la toma de decisiones.

“Lo importante no es solo definir qué se debe hacer, sino cuándo, en qué orden y con qué seguimiento.”

HENRY GANTT

Contexto histórico y necesidad

- A comienzos del siglo XX, los **proyectos industriales eran cada vez más complejos**: construcción, manufactura, obras públicas.
- Los **métodos de planificación eran lineales y poco visuales**, lo que dificultaba el seguimiento del avance.
- **Gantt propuso** una solución: un **diagrama de barras** que mostrara las tareas, su duración y su secuencia en el tiempo.
- Esta herramienta permitió:
 - Identificar **dependencias** entre tareas.
 - Detectar **retrasos** y ajustar recursos.
 - Mejorar la **comunicación** entre equipos.

El **diagrama de Gantt** se convirtió en **una herramienta clave para la gestión de proyectos**, y sigue vigente hoy en software como Trello, MS Project, Asana, entre otros.

HENRY GANTT

¿Qué aportó Gantt?:

- Creación del diagrama de Gantt: herramienta visual para planificar y controlar proyectos.
- Permite ver:
 - Tareas
 - Duración
 - Secuencia
 - Avance

HENRY GANTT

¿Qué cambia con Gantt respecto a Taylor?:

Aspecto	Taylor	Gantt
Enfoque	Científico del trabajo	Visual y organizativo
Herramienta	Oficina de métodos	Diagrama de Gantt
Relación con el trabajador	Selección y entrenamiento	Seguimiento del avance
Aplicación	Producción industrial	Proyectos diversos

HENRY GANTT

Preguntas...

1. ¿Qué ventajas tiene visualizar el avance de un proyecto?
2. ¿Cómo se usa hoy el diagrama de Gantt en software?
3. ¿Qué limitaciones puede tener esta herramienta?

1- ¿Qué ventajas tiene visualizar el avance de un proyecto?

El aporte de **Gantt** fue revolucionario porque **introdujo una forma visual y temporal de representar el trabajo**. Las ventajas son:

- **Claridad en la planificación:** Permite ver qué tareas se hacen, cuándo y en qué orden.
- **Seguimiento del progreso:** Se puede comparar lo planificado con lo real y detectar retrasos.
- **Identificación de dependencias:** Ayuda a entender qué tareas dependen de otras y cuáles pueden hacerse en paralelo.
- **Comunicación efectiva:** Facilita que todos los involucrados (equipo, cliente, dirección) comprendan el estado del proyecto.
- **Gestión de recursos:** Permite asignar personas, herramientas y tiempos de forma más eficiente.

2- ¿Cómo se usa hoy el diagrama de Gantt en software?

Aunque Gantt nació en la era industrial, sigue vigente en entornos digitales, especialmente en:

Herramientas de gestión de proyectos: Como Microsoft Project, Trello (con extensiones), Jira, Asana, ClickUp, etc.

Planificación de entregas: Se usa para definir sprints, hitos, fechas de entrega y tareas clave.

Seguimiento de proyectos grandes: En desarrollos con múltiples equipos, Gantt permite coordinar tareas y visualizar el impacto de los retrasos.

Presentaciones ejecutivas: Se usa para mostrar el estado del proyecto a clientes o stakeholders de forma clara y visual.

3- ¿Qué limitaciones puede tener esta herramienta?

- **Rigidez ante cambios:** Si el proyecto cambia constantemente, **actualizar el Gantt puede volverse tedioso.**
- **Complejidad visual:** En **proyectos grandes**, el diagrama puede volverse **difícil de leer y mantener.**
- **Falsa sensación de control:** Puede parecer que todo está bajo control solo porque está “dibujado”, aunque no se esté ejecutando bien.
- **No muestra carga de trabajo:** No indica si una persona está sobrecargada o si hay recursos mal distribuidos.
- **No se adapta bien a metodologías ágiles:** En entornos ágiles, donde el cambio es constante, se prefieren tableros Kanban o burndown charts.

ACTIVIDAD GRUPAL (simulación): *“Están a cargo de desarrollar una app en 6 semanas. Deben organizar las tareas en un diagrama de Gantt.”*

CADA GRUPO DEBE:

- 1- Definir al menos 6 **tareas** con duración estimada.
- 2- Establecer **dependencias** entre tareas.
- 3- Representar el **avance** en una línea de tiempo (puede ser en papel, Excel, Trello, etc.).
- 4- Identificar qué pasa si una tarea se **retrasa**.

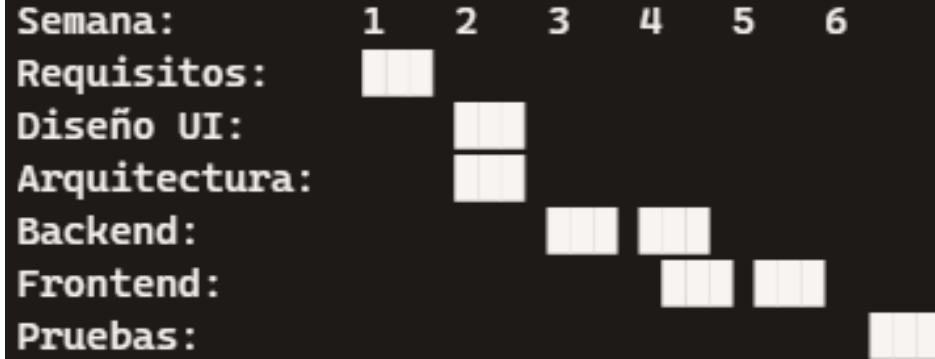
1- POSIBLES TAREAS DEFINIDAS Y DURACIÓN ESTIMADA

Nº	Tarea	Duración	Semana estimada
1	Relevamiento de requisitos	1 semana	Semana 1
2	Diseño de interfaz (UI/UX)	1 semana	Semana 2
3	Arquitectura y base de datos	1 semana	Semana 2
4	Desarrollo del backend	2 semanas	Semanas 3-4
5	Desarrollo del frontend	2 semanas	Semanas 4-5
6	Pruebas y ajustes finales	1 semana	Semana 6

2- POSIBLES DEPENDENCIAS ENTRE TAREAS

- Tarea 1 (Relevamiento) debe completarse antes de iniciar cualquier otra.
- Tareas 2 y 3 (Diseño y arquitectura) pueden hacerse en paralelo, pero ambas deben estar listas antes de comenzar el desarrollo.
- Tarea 4 (Backend) depende de la arquitectura.
- Tarea 5 (Frontend) depende del diseño y puede comenzar cuando el backend esté avanzado.
- Tarea 6 (Pruebas) depende de que el desarrollo esté completo.

3- POSIBLE REPRESENTACIÓN EN LÍNEA DE TIEMPO (DIAGRAMA DE GANTT SIMPLIFICADO)



4- ¿QUÉ PASA SI UNA TAREA SE RETRASA? (fin de la actividad)

Si el **relevamiento de requisitos** se retrasa una semana, todas las tareas posteriores se desplazan, y el proyecto podría extenderse a **7 semanas** o más.

Impacto posible:

- **Diseño y arquitectura** no pueden comenzar a tiempo.
- **Desarrollo** se retrasa, lo que afecta las **pruebas**.
- El equipo debe **reorganizar el cronograma**, buscar tareas paralelas o **reducir el alcance** para cumplir el plazo.



DESCANSO.

HISTORIA

“De los pioneros a la profesionalización”

Por lo estudiado hasta aquí, podemos ver como con el paso del tiempo, la **gestión de proyectos** pasó de ser una práctica empírica a una **disciplina formal**. Por este motivo, en 1969 se funda el **Project Management Institute (PMI)**, que establece estándares, certificaciones y buenas prácticas para gestionar proyectos en todo el mundo.

"La historia nos muestra que la **gestión de proyectos no nació como una disciplina formal**, sino como una respuesta a desafíos reales. A medida que los proyectos se volvieron más complejos, surgió la necesidad de **sistematizar el conocimiento y establecer buenas prácticas**. Este proceso de profesionalización permitió **transformar la experiencia acumulada en estándares aplicables** a cualquier industria."

HISTORIA

1950

**Era moderna gerencia
de proyectos**



1969

Fundación PMI
Project Management
institute.



"Durante las décadas previas a 1969, la gestión de proyectos evolucionó desde enfoques empíricos hacia modelos más estructurados. La creciente complejidad de los proyectos industriales y tecnológicos evidenció la necesidad de una organización que reuniera conocimientos, definiera estándares y profesionalizara la práctica. Así nació el PMI, marcando un antes y un después en la forma de gestionar proyectos en todo el mundo."

HISTORIA

“Con la fundación del PMI en 1969, la GESTION DE PROYECTOS dejó de ser solo una práctica industrial y pasó a tener un marco formal. Hoy, los proyectos se gestionan siguiendo un ciclo de vida estructurado y una matriz de procesos que permite integrar todas las áreas clave.”

HISTORIA- ¿QUÉ ES EL PMI?

- **Project Management Institute (PMI)** Fundado en 1969, es la organización líder mundial en gestión de proyectos. Su objetivo es **profesionalizar la disciplina**, ofreciendo estándares, certificaciones y recursos para mejorar la práctica en todo tipo de industrias.

- **Publica la guía PMBOK®** (Project Management Body of Knowledge).

- Ofrece certificaciones como **PMP®** (Project Management Professional).

- Promueve buenas prácticas, ética profesional y mejora continua.

*"El PMI surge para **unificar criterios y elevar la gestión de proyectos a nivel profesional**, ofreciendo estándares, certificaciones y recursos aplicables en múltiples industrias."*

HISTORIA- ¿QUÉ ES EL PMBOK?

- **PMBOK® Guide – Guía de los Fundamentos para la Dirección de Proyectos** Es un marco de referencia que **organiza la gestión de proyectos en procesos, áreas de conocimiento y fases.**

- Define **49 procesos** distribuidos en:

- **5 grupos de procesos:** Inicio, Planificación, Ejecución, Monitoreo y Cierre.

- **10 áreas de conocimiento:** como alcance, tiempo, costos, calidad, etc.

- Cada **proceso** tiene:

- Entradas

- Herramientas y técnicas

- Salidas

- *El **PMBOK** no es una metodología rígida, sino una guía flexible que puede adaptarse a distintos contextos.*

PMBOK: GRUPOS DE PROCESOS

"Estos cinco grupos de procesos reflejan el ciclo de vida de cualquier proyecto, desde su concepción hasta su cierre, y permiten organizar el trabajo de forma lógica y progresiva."

Grupo de procesos	Descripción breve
1. Inicio	Define el proyecto, sus objetivos y obtiene la aprobación para comenzar.
2. Planificación	Establece el alcance, cronograma, costos, calidad, riesgos y recursos necesarios.
3. Ejecución	Se lleva a cabo el trabajo planificado, se gestionan equipos y se entregan resultados.
4. Monitoreo y Control	Se supervisa el avance, se comparan resultados con el plan y se aplican correcciones.
5. Cierre	Se finaliza el proyecto, se documenta lo aprendido y se liberan recursos.

PMBOK: AREAS DE CONOCIMIENTO

"Las áreas de conocimiento del PMBOK agrupan procesos relacionados por temática, como la gestión del alcance, del tiempo o de los costos. Estas áreas permiten abordar cada dimensión del proyecto de forma especializada, facilitando su planificación, ejecución y control."

Nº	Área de conocimiento
1	Gestión de la Integración
2	Gestión del Alcance
3	Gestión del Cronograma
4	Gestión de los Costos
5	Gestión de la Calidad
6	Gestión de los Recursos
7	Gestión de las Comunicaciones
8	Gestión de los Riesgos
9	Gestión de las Adquisiciones
10	Gestión de los Interesados

LA MATRIZ DE PROCESOS: ¿CÓMO SE ORGANIZA?

La matriz del **PMBOK** Cruza los **5 grupos de procesos** con las **10 áreas de conocimiento** para organizar los **49** procesos de gestión de proyectos.

- Cada celda representa un proceso específico (ej. “Controlar el cronograma” en el área de cronograma y el grupo de monitoreo).
- Esta estructura permite **planificar y gestionar proyectos de forma integral**, asegurando que cada aspecto esté cubierto en cada proceso/fase.

TABLA AMPLIADA DE LOS PROCESOS DEL PMBOK (6ª

Áreas ↓ \ Grupos →	INICIAR	PLANIFICAR	EJECUTAR	M. & CONTROLAR	CERRAR
4.- Gestión de la INTEGRACIÓN	4.1 Desarrollar el Acta de Constitución del Proyecto.	4.2 Desarrollar el Plan para la Dirección del Proyecto	4.3 Dirigir y Gestionar el Trabajo del Proyecto 4.4 Gestionar el Conocimiento del Proyecto	4.5 Monitorizar y Controlar el Trabajo del Proyecto. 4.6 Realizar el Control Integrado de Cambios.	4.7 Cerrar el Proyecto o Fase.
5.- Gestión del ALCANCE		5.1 Planificar la Gestión del Alcance. 5.2 Recopilar los requerimientos 5.3 Definir el Alcance 5.4 Crear la EDT ("WBS")		5.5 Validar el Alcance. 5.6 Controlar el Alcance.	
6.- Gestión del CRONOGRAMA		6.1 Planificar la Gestión del Cronograma 6.2 Definir las actividades 6.3 Secuenciar las actividades 6.4 Estimar la Duración de las Actividades 6.5 Desarrollar el Cronograma		6.6 Controlar el Cronograma.	
7.- Gestión de los COSTES		7.1 Planificar la Gestión de los Costes 7.2 Estimar el Coste 7.3 Determinar el Presupuesto		7.4 Controlar los Costes.	
8.- Gestión de la CALIDAD		8.1 Planificar la Gestión de la Calidad	8.2 Gestionar la Calidad	8.3 Controlar la Calidad	
9.- Gestión de los RECURSOS		9.1 Planificar la Gestión de Recursos 9.2 Estimar los Recursos de la Actividades	9.3 Adquirir recursos 9.4 Desarrollar el Equipo. 9.5 Dirigir al Equipo	9.6 Controlar los recursos	
10.- Gestión de las COMUNICACIONES		10.1 Planificar la Gestión de las Comunicaciones.	10.2 Gestionar las Comunicaciones	10.3 Monitorizar las Comunicaciones	
11.- Gestión de los RIESGOS		11.1 Planificar la Gestión de Riesgos 11.2 Identificar los Riesgos. 11.3 Realizar el Análisis Cualitativo. 11.4 Realizar el Análisis Cuantitativo. 11.5 Planificar la Respuesta	11.6 Implementar la Respuesta a los Riesgos	11.7 Monitorizar los Riesgos.	
12.- Gestión de las ADQUISICIONES		12.1 Planificar la Gestión de las Adquisiciones	12.2 Efectuar las Adquisiciones.	12.3 Controlar las Adquisiciones.	
13.- Gestión de los INTERESADOS	13.1 Identificar a los Interesados	13.2 Planificar el involucramiento de los Interesados	13.3 Gestionar la participación de los Interesados.	13.4 Monitorizar el involucramiento de los Interesados.	
49 procesos - TOTAL:	2	24	10	12	1

TABLA SIMPLIFICADA DE LOS PROCESOS DEL PMBOK (6ª EDICIÓN)

Área de conocimiento	¿Qué gestiona?	Procesos típicos (ejemplos)
Integración	Coordinación general del proyecto	Acta de constitución, plan del proyecto, cierre
Alcance	Qué se va a entregar	Recopilar requisitos, definir alcance, crear EDT (estructura de desglose de trabajo)
Cronograma	Cuándo se hace cada tarea	Secuenciar actividades, estimar duración, desarrollar cronograma
Costos	Cuánto cuesta	Estimar costos, determinar presupuesto, controlar costos
Calidad	Nivel de calidad esperado	Planificar calidad, gestionar calidad, controlar calidad
Recursos	Quién hace qué	Estimar recursos, adquirir recursos, dirigir equipo
Comunicaciones	Cómo se comunica el equipo	Planificar comunicaciones, gestionar comunicaciones
Riesgos	Qué puede salir mal	Identificar riesgos, analizar riesgos, planificar respuestas
Adquisiciones	Qué se compra o contrata	Planificar adquisiciones, efectuar compras, controlar contrataciones
Interesados	Quién está involucrado o afectado	Identificar interesados, gestionar participación

Aplicación en proyectos de software

"En proyectos de desarrollo de software, los procesos del PMBOK se adaptan para gestionar desde la definición de requerimientos hasta la entrega del producto. Por ejemplo, en el grupo de Inicio se formaliza el acta de constitución del proyecto; en Planificación se definen sprints, recursos y cronogramas; en Ejecución se desarrollan funcionalidades; y en Monitoreo y Control se revisa el avance mediante métricas como el burn-down chart. Finalmente, el Cierre incluye la validación del producto y la documentación técnica."

Ejemplo de Caso práctico: “Proyecto de software”.

*"Una empresa necesita desarrollar una aplicación móvil para gestionar turnos médicos. El proyecto comienza con la elaboración del acta de constitución y la identificación de los interesados (**Inicio**). Luego, el equipo define los requerimientos, el cronograma y los recursos necesarios (**Planificación**). Durante la **Ejecución**, se desarrollan las funcionalidades principales y se realizan pruebas internas. A medida que avanza el proyecto, se monitorean los tiempos y se ajustan tareas según los resultados de las pruebas (**Monitoreo y Control**). Finalmente, se entrega la aplicación, se capacita al cliente y se documenta el cierre del proyecto (**Cierre**)."*

Ejemplo de Caso práctico: “Proyecto de software”.

(en caso de construir la EDT, ¿cómo hacerlo?)

Durante la planificación, una vez definidos los **requerimientos del proyecto**, se procede a:

- **Identificar los entregables principales** (por ejemplo: diseño UI, desarrollo backend, pruebas, capacitación).
- **Descomponer cada entregable** en subentregables o componentes más manejables.
- **Organizar jerárquicamente** en un diagrama o lista estructurada.
- **Asignar códigos** a cada elemento (opcional pero útil para seguimiento).
- **Validar con los interesados** que el EDT cubre todo el alcance.

Ejemplo de Caso práctico: “Proyecto de software”.

(en caso de construir la EDT, ¿cómo hacerlo?)

Durante la construcción de la EDT, también se necesitara definir los **niveles**, los cuales indican la **profundidad de descomposición del trabajo**. Cada nivel inferior representa una **parte más específica** del entregable superior.

Nivel	Qué representa	En este ejemplo
Nivel 1	Proyecto completo	App móvil para gestión de turnos
Nivel 2	Entregables principales	Diseño de interfaz, Desarrollo, Pruebas
Nivel 3	Subentregables	Backend, Frontend, Bocetos, Prototipo
Nivel 4+	Componentes más detallados (opcional)	Módulo de turnos, Módulo de usuarios

EJEMPLO DE CASO PRÁCTICO: “PROYECTO DE SOFTWARE”. En caso de **construir la EDT**, podría ser así:

Nivel	Elemento EDT	Justificación técnica
1	App móvil para gestión de turnos	Representa el entregable final del proyecto
2	Recolección de requerimientos	Primer entregable clave para definir el alcance; surge de la planificación
2	Diseño de interfaz	Entregable visual que precede al desarrollo técnico
3	Bocetos iniciales	Subentregable que permite validar ideas con el cliente
3	Prototipo funcional	Resultado más avanzado, útil para pruebas y ajustes
2	Desarrollo	Entregable técnico principal del proyecto
3	Backend (gestión de turnos, usuarios)	Parte lógica del sistema; maneja datos y reglas
3	Frontend (pantallas, navegación)	Parte visible para el usuario; interfaz gráfica
2	Pruebas internas	Entregable que valida funcionalidad antes de la entrega
2	Capacitación al cliente	Entregable de transferencia de conocimiento; parte del cierre
2	Documentación final	Entregable formal que respalda el cierre del proyecto

EJEMPLO DE CASO PRÁCTICO: “PROYECTO DE SOFTWARE”. Ejemplo de asignacion de niveles:

- 1.0 App móvil para gestión de turnos
 - 1.1 Recolección de requerimientos
 - 1.2 Diseño de interfaz
 - 1.2.1 Bocetos iniciales
 - 1.2.2 Prototipo funcional
 - 1.3 Desarrollo
 - 1.3.1 Backend
 - 1.3.2 Frontend
 - 1.4 Pruebas internas
 - 1.5 Capacitación al cliente
 - 1.6 Documentación final

EJEMPLO DE CASO PRÁCTICO: “PROYECTO DE SOFTWARE”. ¿Por qué no se incluyeron actividades como “programar” o “testear” en la EDT?

Porque el **EDT no describe tareas, sino qué se debe entregar**. Las actividades se derivan **después**, cuando se construye el **diccionario del EDT** o se desarrolla el cronograma. Por ejemplo:

- “Programar el backend” sería una actividad dentro del entregable “Backend”.
- “Realizar pruebas unitarias” sería una actividad dentro de “Pruebas internas”.

EJEMPLO DE CASO PRÁCTICO: “PROYECTO DE SOFTWARE. EDT, ¿cuándo la construimos?”

No siempre es obligatorio construir un EDT, pero en el contexto del PMBOK y de una gestión profesional de proyectos, **es altamente recomendable** y en muchos casos **es considerado una buena práctica esencial**, especialmente cuando el proyecto:

- Tiene múltiples entregables o componentes técnicos
- Requiere coordinación entre equipos
- Implica planificación detallada de tiempos, costos y recursos
- Necesita trazabilidad clara del alcance



DESCANSO.

GESTIÓN DE PROYECTO (recordando los procesos).



Integrar los diferentes componentes de un proyecto para lograr un objetivo específico durante las etapas de:



[Inicio] → [Planificación] → [Ejecución] → [Monitoreo y Control] → [Cierre]

DE LA ESTRUCTURA AL LIDERAZGO Y LA PLANIFICACIÓN OPERATIVA



*"Una vez que comprendemos cómo se organiza un proyecto a través de procesos y áreas de conocimiento, el siguiente paso es entender quién lidera esa estructura y cómo se toman decisiones clave. **El Director de Proyecto** es quien **articula** los **procesos**, **gestiona** las **restricciones** y **guía** al **equipo** hacia los **objetivos**. Para hacerlo, necesita herramientas que le permitan estimar tiempos, esfuerzos y costos con precisión. En este bloque, exploraremos el rol del director, la triple restricción y los métodos de estimación aplicables a proyectos de software."*

OBJETIVO DEL DIRECTOR DE PROYECTO



Cumplir con todos los objetivos del proyecto dentro de las restricciones establecidas

PRINCIPALES RESTRICCIONES DE UN PROYECTO

Triple Restricción



Tiempo



Costos



Alcance

ESTIMACIÓN DE TIEMPO, ESFUERZO Y COSTO EN PROYECTOS DE SOFTWARE

• *"Recordemos que el alcance define qué se va a hacer. El esfuerzo es la cantidad de trabajo necesario para cumplir con ese alcance. Por eso, al estimar tiempo y costo, partimos del alcance y lo traducimos en esfuerzo medible."*

- **Alcance** es el "qué".
- **Esfuerzo** es el "cuánto trabajo requiere ese qué".

DEL ALCANCE A ESTIMACIONES: ESFUERZO, DURACIÓN Y COSTO

Una vez definido el alcance del proyecto, el **director realiza estimaciones** para planificar y controlar:

Esfuerzo: ¿Cuánto trabajo implica? → Se mide en HH (horas-hombre), MD (días-hombre), MM (meses-hombre).

Duración: ¿Cuánto tiempo llevará? → Depende del esfuerzo y la disponibilidad de recursos.

Costo: ¿Cuánto costará? → Se calcula en función del esfuerzo y las tarifas aplicables.

Estas estimaciones son la base para construir el cronograma, asignar recursos y definir el presupuesto.

¿QUÉ SIGUE DESPUÉS DE ESTIMAR?

Una vez realizadas las estimaciones, el director puede avanzar en la **planificación detallada** del proyecto:

- **Construcción del cronograma** → Secuencia de actividades, dependencias y fechas clave.
- **Asignación de recursos** → Personas, equipos, materiales necesarios.
- **Definición del presupuesto** → Costos por actividad, por fase, por recurso.
- **Identificación de riesgos** → Análisis de incertidumbres y planes de contingencia.

Todo esto se integra en el **plan del proyecto**, que guía la ejecución y el control.

Plan de proyecto: documento integral que reúne todos los elementos necesarios para **dirigir, ejecutar y controlar** el proyecto. No es solo el cronograma ni el presupuesto: es el conjunto completo de planes que definen cómo se va a trabajar.

INTRODUCCIÓN AL ANÁLISIS DE PUNTOS DE FUNCIÓN

¿Cómo estimamos el **tamaño funcional** de un proyecto de software?

Una vez definido el alcance y estimado el esfuerzo, necesitamos medir el tamaño funcional del software para afinar la planificación.

El método de **Análisis de Puntos de Función (FPA)** permite cuantificar los requerimientos funcionales desde la perspectiva del usuario, sin depender de la tecnología utilizada. En otras palabras, es un **método que sirve para medir lo que hace un software, pero desde el punto de vista del usuario**, no desde el código ni la tecnología usada.

Este enfoque facilita:

- Estimaciones más objetivas y replicables (cuánto trabajo lleva hacer un sistema).
- Comparaciones entre proyectos (aunque estén hechos con distintas tecnologías).
- Mejores decisiones sobre esfuerzo, duración y costo (cuánto tiempo, esfuerzo y dinero se necesita).

Tamaño funcional: Cantidad de funcionalidades que ofrece al usuario. Mide *qué hace* el sistema desde el punto de vista del usuario (entradas, salidas, consultas, archivos), no *cómo lo hace*.

RESUMIENDO: ¿QUÉ ES EL ANÁLISIS DE PUNTOS DE FUNCIÓN (FPA)?

RTA: Una técnica para medir el **tamaño funcional** del software desde la perspectiva del usuario.

- No depende de la tecnología ni del lenguaje de programación.
- Permite estimar esfuerzo, duración y costo de forma objetiva.
- Es un estándar internacional (ISO/IEC 20926).

PUNTOS DE FUNCIÓN (FPA)

- Antes de existir FPA, la métrica de comparación entre proyectos de software, eran las líneas de código.
- La métrica resultaba demasiada técnica o superficial. El usuario no necesariamente llega a entender que se está midiendo.
- Antes de existir FPA, otra métrica de comparación entre proyectos de software, eran la cantidad de pantallas, informes o archivos que entregaba dicho software.
- FPA Toma esto y mide la **función**, no la **cantidad** de pantallas **por** esa **función**.

PROCESO DE MEDICIÓN: OBJETIVOS

- Ser una medida **consistente**:

Consistente: Dos profesionales analizando un mismo proyecto llegan al **mismo resultado**.

- Simple para minimizar el esfuerzo de la medición.

¿CÓMO REALIZAR LA MEDICIÓN (en PFA)?

Recordemos que lo que se busca medir es: ¿cuanto valor funcional le ofrece el sistema al usuario?.

Eso incluye:

- ¿Qué puede hacer el usuario con el sistema?
- ¿Qué datos almacena y gestiona el sistema?
- ¿Qué información presenta o devuelve el sistema?

¿CÓMO REALIZAR LA MEDICIÓN EN PFA?

El análisis divide las funciones del sistema en dos grandes grupos:

1. Funciones de transacción (Interacción con el usuario)

Son las acciones que el usuario realiza con el sistema. Se dividen en:

- **Entradas externas (EE):** El usuario envía datos al sistema (ej. cargar un formulario).
- **Salidas externas (SE):** El sistema devuelve información procesada (ej. un informe).
- **Consultas externas (CE):** El usuario pide información sin que se procese (ej. buscar un cliente).

2. Funciones de datos (Almacenamiento)

Son los datos que el sistema guarda y usa. Se dividen en:

- **Archivos lógicos internos (ALI):** Datos que el sistema mantiene (ej. base de clientes).
- **Archivos de interfaz externa (AIE):** Datos que vienen de otros sistemas (ej. lista de productos de un proveedor).

¿Por qué se mide así?

Porque cada tipo de función tiene un **peso distinto** según su complejidad. Esto permite calcular los **puntos de función** totales del sistema.

¿CÓMO REALIZAR LA MEDICIÓN?



**Componentes
funcionales
básicos**



• **Entrada externa
(EI -> External
input)**



• **Salida externa (EO
-> External output)**

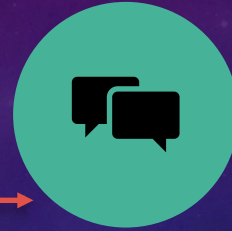


• **Consulta externa
(EQ -> External
query)**

¿CÓMO REALIZAR LA MEDICIÓN?



Funciones de Almacenamiento de datos



• **ILF: Internal Logical File**(mantenido por la propia app)



• **EIF: External Interface File** (**referenciado por la app, pero mantenido por otro sistema**)

Ambos se consideran **funciones de almacenamiento**, y se evalúan según su **complejidad** (baja, media, alta), en función de la cantidad de **elementos de datos (DET)** y **tipos de registros (RET)** que contienen.

¿CÓMO REALIZAR LA MEDICIÓN?- ¿QUÉ SON LOS **DET Y RET**?

DET – Data Element Type (Tipo de Elemento de Datos)

Son los **campos individuales** o **atributos** que forman parte de un archivo lógico.

Ejemplo: En una tabla de clientes, los DET podrían ser: nombre, apellido, DNI, dirección, teléfono, etc.

Cada campo **único y significativo** cuenta como un DET.

RET – Record Element Type (Tipo de Registro Lógico)

Son los **subgrupos de datos** dentro de un archivo lógico que tienen **significado propio**.

Ejemplo: Si tenemos una tabla de empleados que se agrupa por sucursal, cada sucursal sería un RET.

También puede pensarse como **categorías o conjuntos de registros** que se diferencian por estructura o propósito.

¿CÓMO REALIZAR LA MEDICIÓN?

Pasos para calcular los Puntos de Función

- 1 Identificar funciones del usuario
- 2 Clasificar según tipo (entrada, salida, consulta, etc.)
- 3 Determinar complejidad (simple, media, compleja)
- 4 Asignar valor estándar según la tabla
- 5 Sumar los puntos de función sin ajustar (**PFSA**)
- 6 Aplicar factores de ajuste si corresponde

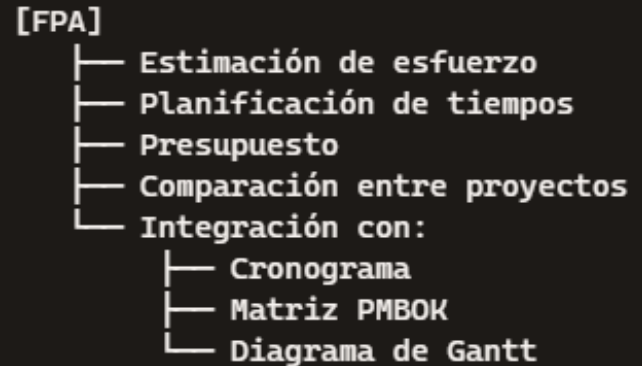
Los **puntos de función** no son elementos físicos, sino valores numéricos que se asignan a cada **función del sistema**, según su tipo y su complejidad.

¿QUÉ APORTA EL ANÁLISIS DE PUNTOS DE FUNCIÓN A LA GESTIÓN DE PROYECTOS?

El FPA no solo mide el tamaño funcional del software, sino que también:

- Permite estimar el esfuerzo de forma objetiva
- Mejora la precisión en la planificación de tiempos y costos
- Facilita la comparación entre proyectos
- Ayuda al director a tomar decisiones informadas sobre recursos, cronograma y presupuesto
- Se integra con otras herramientas como el cronograma, la matriz de procesos y el diagrama de Gantt

“El análisis funcional es una herramienta clave para transformar requerimientos en estimaciones concretas. Su uso fortalece la planificación y la toma de decisiones del director del proyecto”.



EJEMPLO: SISTEMA DE GESTIÓN DE FÚTBOL

Para el siguiente ejemplo, se considerará que todas las funciones identificadas serán de **complejidad media**.

Objetivo del sistema

- Desarrollar una aplicación que permita gestionar la información de jugadores, equipos, partidos y estadísticas en una liga de fútbol amateur.

Actores principales

- **Administrador:** carga y modifica datos de jugadores, equipos y partidos.
- **Usuario general:** consulta estadísticas, resultados y posiciones.

Funcionalidades clave

- Registro y edición de jugadores, equipos y partidos.
- Consulta de estadísticas individuales y grupales.
- Generación de reportes por fecha, equipo o jugador.
- Visualización de tabla de posiciones y fixture.

Alcance del análisis

- Se aplicará la técnica de **Análisis de Puntos de Función (FPA)** para estimar el tamaño funcional del sistema, como base para calcular esfuerzo, duración y costo de desarrollo.

EJEMPLO: SISTEMA DE GESTIÓN DE FÚTBOL

- Para el siguiente ejemplo, se considerará que todas las funciones identificadas serán de **complejidad media**.
- El sistema requerido es:
 - Registro de Equipos de fútbol
 - Registros de partidos
 - Buscar partido por fecha
 - Actualización de datos del equipo
 - Eliminar equipos
 - Listado de equipos
 - 1 reporte de los equipos registrados por rango de fechas
 - 1 reporte de partidos
 - 4 Tablas de DB

EJEMPLO: SISTEMA DE GESTIÓN DE FÚTBOL

- Para el siguiente ejemplo, se considerará que todas las funciones identificadas serán de **complejidad media**.
- El sistema requerido es:
 - Registro de Equipos de fútbol (EI 4PF)
 - Registros de partidos (EI 4PF)
 - Buscar partido por fecha (EQ 4PF)
 - Actualización de datos del equipo (EI 4PF)
 - Eliminar equipos (EI 4PF)
 - Listado de equipos (EO 5PF)
 - 1 reporte de los equipos registrados por rango de fechas (EO 5PF)
 - 1 reporte de partidos (EO 5PF)
 - 4 Tablas de DB (ILF 40PF)

EJEMPLO: SISTEMA DE GESTIÓN DE FÚTBOL (Resumen)

Función	Tipo	Ejemplo
Buscar	EQ	Buscar partido por fecha
Insertar	EI	Registro de equipos y partidos
Actualizar	EI	Modificar datos del equipo
Eliminar	EI	Eliminar equipos
Listar	EO	Listado de equipos
Reportes	EO	Informe por rango de fechas y de partidos
Tablas	ILF	4 tablas de base de datos

VALORES ESTÁNDAR (IFPUG) INTERNATIONAL FUNCTION POINT USERS GROUP (p/aplicar al ejemplo)

Tipo / Complejidad	Baja	Media	Alta
(EI) Entrada externa	3 PF	4 PF	6 PF
(EO) Salida externa	4 PF	5 PF	7 PF
(EQ) Consulta externa	3 PF	4 PF	6 PF
(ILF) Archivo lógico interno	7 PF	10 PF	15 PF
(EIF) Archivo de interfaz externo	5 PF	7 PF	10 PF

La complejidad se determina según:

- DET (Data Element Types)**: cantidad de campos únicos.
- RET (Record Element Types)**: cantidad de subgrupos lógicos.
- FTR (File Types Referenced)**: cantidad de archivos referenciados (en funciones de transacción).

EJEMPLO: SISTEMA DE GESTIÓN DE FÚTBOL

Tipo / Complejidad	Baja	Media	Alta	TOTAL
(EI) Entrada externa	3 PF	4 x 4 PF	6 PF	16
(EO) Salida externa	4 PF	3 x 5 PF	7 PF	15
(EQ) Consulta externa	3 PF	1 x 4 PF	6 PF	4
(ILF) Archivo lógico interno	7 PF	4 x 10 PF	15 PF	40
(EIF) Archivo de interfaz externo	5 PF	0 x 7 PF	10 PF	0
			PFSA	75

CÁLCULO DE PUNTOS DE FUNCIÓN AJUSTADO

FACTOR DE AJUSTE (IFPUG)

La tabla contiene **14 características generales del sistema** (llamadas *General System Characteristics* o GSCs), cada una evaluada en una escala de **0 a 5**:

#	Factor de ajuste (GSC)	Escala (0–5)	¿Qué mide?
1	Comunicación de datos	0–5	Interacción con otros sistemas
2	Procesamiento distribuido	0–5	Lógica en múltiples nodos
3	Objetivos de rendimiento	0–5	Requisitos de velocidad y respuesta
4	Configuración del equipamiento	0–5	Dependencia del hardware
5	Tasa de transacciones	0–5	Volumen de operaciones
6	Entrada de datos en línea	0–5	Captura directa de datos
7	Interfase con el usuario	0–5	Facilidad de uso y navegación
8	Actualizaciones en línea	0–5	Modificación de datos en tiempo real
9	Procesamiento complejo	0–5	Lógica de negocio sofisticada
10	Reusabilidad del código	0–5	Uso compartido de componentes
11	Facilidad de implementación	0–5	Instalación y configuración
12	Facilidad de operación	0–5	Mantenimiento y uso cotidiano
13	Instalaciones múltiples	0–5	Uso en distintas ubicaciones
14	Facilidad de cambios	0–5	Adaptabilidad y evolución del sistema

FACTOR DE AJUSTE (IFPUG)

¿Cómo se usa esta tabla?

- Se asigna un puntaje de 0 a 5 a cada ítem según su **grado de influencia** en el sistema.
- La suma total da el **Factor de Ajuste (FA)**.
- Se aplica en la fórmula:
$$PFA = PFSA \times (0.65 + 0.01 \times FA)$$

Factor de Ajuste	Puntaje
Comunicación de Datos	4
Procesamiento Distribuido	4
Objetivos de Rendimiento	1
Configuración del equipamiento	1
Tasa de transacciones	3
Entrada de Datos en Línea	5
Interfase con el usuario	2
Actualizaciones en Línea	3
Procesamiento Complejo	1
Reusabilidad del Código	1
Facilidad de Implementación	
Facilidad de Operación	1
Instalaciones Múltiples	2
Facilidad de Cambios	4
Factor de Ajuste	32

PFA- PUNTOS DE FUNCIÓN AJUSTADO

- $PFA = PFSA * [0.65 + (0.01 * \text{factor de ajuste})]$

- Donde:

- **PFSA**: Puntos de función sin ajustar

- **PFA** : Puntos de función ajustado

$$PFA = 75 * [0.65 + (0.01 * 32)]$$

$$PFA = 75 * [0.65 + 0.32]$$

$$PFA = 75 * 0.97$$

$$PFA = 73.8 \rightarrow 74$$

ESTIMACIÓN DE ESFUERZO

- **PFA = 74**

- *El objetivo ahora es estimar la cantidad de esfuerzo necesario para desarrollar la aplicación. Este esfuerzo se mide en horas/hombre, meses/hombre o años/hombre. Los puntos de función en cierto modo son una medida subjetiva.*

ESTIMACIÓN DE ESFUERZO

- *La cantidad de horas/hombre por punto de función es algo difícil e impreciso de valorar, de forma global. Esto es normal, lo contrario sería suponer que la productividad de todas las empresas de desarrollo de software es igual.*

ESTIMACIÓN DE ESFUERZO (tabla estandarizada pero no universal)

Lenguaje	Horas PF promedio	Lineas de código por PF
Ensamblador	25	300
COBOL	15	100
Lenguajes 4ta Generación	8	20

$H/H = PFA * \text{Horas PF promedio}$

$H/H = 74 * 8$

$H/H = \mathbf{592 \text{ Horas hombre}}$

Ejemplo:

5 horas diarias de trabajo

1 mes = 20 días

$592 / 5 = 118,4$ días de trabajo

$118,4 / 20 = \mathbf{5,92 \text{ meses}}$ para desarrollar el software de lunes a viernes 5 horas diarias con 1 trabajador (ESTIMACIÓN de duración del proyecto)

ESTIMACIÓN DE DURACIÓN DEL PROYECTO

H/H = **592 Horas hombre**

Desarrolladores = 3

Horas = $592 / 3 = 197,3$ horas (Duración del proyecto en horas)

$197,3 / 5 = 39,46$ días de trabajo

$39,46 / 20 = 1,97$ **meses para desarrollar el software de lunes a viernes 5 horas diarias con 3 desarrolladores (ESTIMACIÓN de duración del proyecto)**

CÁLCULO DE PRESUPUESTO DEL PROYECTO

- Sueldo mensual desarrolladores: \$400.000
- Otros costos del proyecto: \$1.000.000
- Costo = (Desarrolladores * Duración meses * sueldos) + Otros costos
- Costo = $(3 * 1,97 * 400000) + 1000000 = \textbf{\$3.364.000}$

FINALMENTE: ¿CÓMO SE CONECTAN TODOS LOS CONCEPTOS? (desde los objetivos del director).

1. **Objetivo del director:** cumplir metas dentro de restricciones (tiempo, costo, alcance).
2. **Relación entre alcance, esfuerzo, duración y costo:** traducir el “qué” en “cuánto trabajo”.
3. **Estimaciones operativas:** medir esfuerzo (HH), duración (días/meses), costo (\$).
4. **Planificación:** cronograma, recursos, presupuesto, riesgos.
5. **Plan del proyecto:** integración de todas las decisiones.
6. **FPA:** medición del tamaño funcional del software.
7. **Aplicación del FPA:** funciones → puntos → esfuerzo → duración → costo.

Cada concepto se conecta con el siguiente. La gestión de proyectos no es una suma de partes aisladas, sino un proceso integrado que transforma una idea en resultados concretos.