

Planificación y Gestión de Proyectos Informáticos

Seguimiento y control de proyectos

Jose M. Moyano

jmoyano@ugr.es

Departamento de Ciencias de la Computación e Inteligencia Artificial

Universidad de Granada

Curso 2022-23



DaSCI

Instituto Andaluz de Investigación en
Data Science and Computational Intelligence



**UNIVERSIDAD
DE GRANADA**

Índice

Seguimiento y control del proyecto

Retrospectivas del proyecto

Seguimiento y control del proyecto

El punto de partida

Disponemos de:

- ▶ Programación del proyecto
- ▶ Aplicación de recursos en cada instante
- ▶ Flujo de caja aceptado y coste global

Seguimiento y control del proyecto

Tendríamos que llevar a cabo un seguimiento continuo de lo planificado.

Tomando medidas oportunas cuando:

- ▶ Se produzcan retrasos
- ▶ Existan costes por encima de lo planificado
- ▶ Se contravenga alguna de las condiciones acordadas que fueron base en la decisión de realizar este proyecto

Seguimiento y control del proyecto

- ▶ El progreso de un proyecto se mide conforme se van completando tareas y produciendo resultados (work products): modelos, código, casos de prueba...
- ▶ Los resultados del trabajo no se consideran completados hasta que se aprueban como parte del proceso de QA, por ejemplo, usando revisiones técnicas.

Seguimiento y control del proyecto

Periódicamente (p.ej. semanalmente), el gestor debería:

- ▶ Recopilar datos resumidos del proyecto: planificación, bugs, tiempo invertido...
- ▶ Comparar esos datos con el plan: hitos completados, errores identificados, esfuerzo realizado...
- ▶ Revisar y actualizar la lista de riesgos del proyectos.
- ▶ Revisar los cambios propuestos y aprobados y sus efectos acumulativos sobre el plan del proyecto.
- ▶ Informar del estado del proyecto a los “stakeholders”, preferiblemente mediante mecanismos que mantengan la visibilidad del estado del proyecto.
- ▶ Tomar medidas correctivas si los resultados reales se desvían significativamente de los planes o aparecen nuevos riesgos que deben tenerse en cuenta.

Seguimiento y control del proyecto

Al final de cada fase o iteración, el gestor debería:

- ▶ Considerar solicitudes de cambios.
- ▶ Reestimar esfuerzo, plan temporal y coste.
- ▶ Mantener un log del proyecto.

Seguimiento y control del proyecto

Objetivo del seguimiento: Conocer si el proyecto está bajo control.

Podemos decir que el proyecto está bajo control si los hitos se están alcanzando:

- ▶ A tiempo
- ▶ Con los recursos estimados
- ▶ Con un nivel de calidad esperado

Seguimiento y control del proyecto

¿Por qué es necesario el seguimiento y control?

Porque al realizar la planificación hacemos estimaciones de: tareas necesarias, recursos para cada tarea, productividad estimada, ...
Es probable que no coincida exactamente lo planificado con la realidad.

Seguimiento y control del proyecto

Tan pronto como se observen desviaciones hay que:

- ▶ Replanificar
- ▶ Renegociar el plan del proyecto con los clientes (si es necesario)

Seguimiento y control del proyecto

Actividades de control:

- ▶ Desarrollar **estándares de productividad**. Establecen las condiciones o medidas que deben darse cuando las tareas se realizan de forma correcta.
- ▶ Establecer **sistemas de monitorización** e informes. Determinar qué datos son necesarios, así como quién y cuándo los debe recibir.
- ▶ **Medir los resultados**. Determinar los niveles de cumplimiento sobre las metas y estándares.

Seguimiento y control del proyecto

Actividades de control:

- ▶ Iniciar **acciones correctivas**. Reforzar estándares, ajustar metas, o replanificar.
- ▶ **Recompensar** y disciplinar. Elogiar, remunerar, y disciplinar al personal.
- ▶ **Documentar** los métodos de control. Cualquier actividad dentro del plan de control, como los estándares, los métodos de medición o monitorización, ..., deben quedar reflejados en la documentación correspondiente.

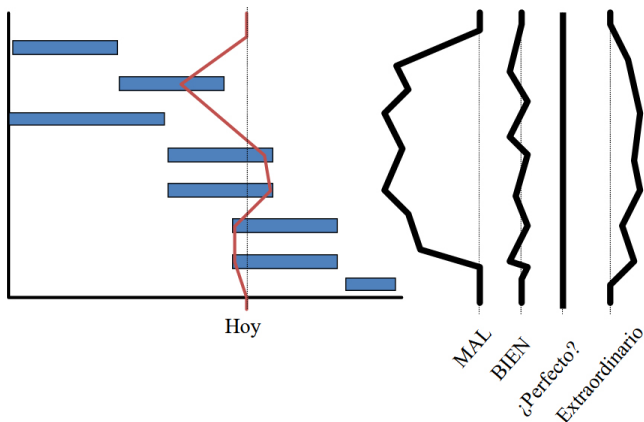
Comparación entre planificación y realidad.

Una posible opción es, sobre un diagrama de Gantt, realizar una línea quebrada vertical que:

- ▶ Comienza y termina en el día actual.
- ▶ Los distintos vértices de la línea se sitúan en el punto de progreso de las distintas tareas no terminadas, o las que deberían haber comenzado y no lo han hecho.

Seguimiento y control del proyecto

Comparación entre planificación y realidad.



Seguimiento y control del proyecto

Cuando aparecen problemas con respecto al proyecto, la decisión sobre qué hacer debe tomarse al nivel jerárquico apropiado.

- ▶ A nivel **operativo**. Los pequeños ajustes, el director del proyecto lo deja a los técnicos.
- ▶ A nivel **táctico**. Ajustes del plan de mayor nivel (retrasos de una semana, etc.), los trata el director del proyecto.
- ▶ A nivel **estratégico**. Grandes retrasos u otras incidencias de mayor nivel (p.ej. se han fusionado dos empresas y podemos utilizar un sistema similar a la otra, hay cambios en la estrategia de la empresa, etc.). Las decisiones son más drásticas.

Seguimiento y control del proyecto

Replanificar o corregir

La replanificación debe realizarse de forma continua para que el calendario se ajuste a la realidad.

- ▶ Los desarrolladores se sentirán menos frustrados si ven que los objetivos son alcanzables.
- ▶ Los clientes tendrán más claro lo que pueden esperar.

En algunos casos, en lugar de modificar el plan, se podrá *forzar* al equipo de desarrollo para que la situación real se aproxime a la planificada.

Crisis en proyectos informáticos

Llamamos crisis de un proyecto al periodo que va desde que se produce una situación seria de desajuste, hasta que es reencauzada.

Crisis en proyectos informáticos

Posibles métodos de gestión de la crisis:

- ▶ Anuncio y publicidad general del problema al equipo.
- ▶ Reasignación de responsabilidades y autoridad.
- ▶ Actualización de la información de situación.
- ▶ Relajación de las restricciones sobre los recursos.
- ▶ Poner al personal del proyecto a trabajar a tope.
- ▶ Establecer una fecha de finalización o reencauzamiento de la crisis.

Crisis en proyectos informáticos

Recuperación tras la crisis

- ▶ Eliminar al personal (o recursos) innecesario.
- ▶ Realizar un estudio tras la crisis.
- ▶ Replanificar el proyecto (coste pendiente y total).

Seguimiento y control del proyecto

¿Cómo se consigue que las personas adopten buenas prácticas?

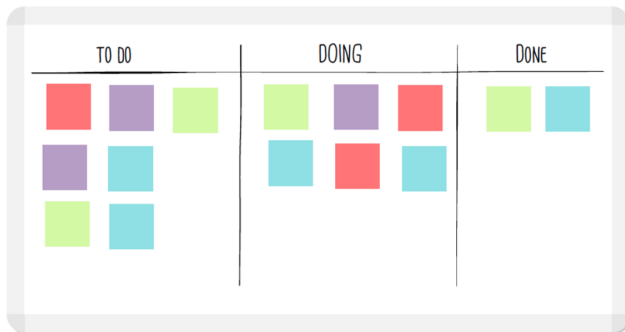
Muy fácil: haciéndolas visibles.

La visibilidad de un proceso que funciona hace que la gente se dé cuenta de ello y quiera utilizarlo. Además, facilita el trabajo del gestor del proyecto. Por ejemplo: el uso de scrum o kanban.

Cualquier proceso visible es un “radiador de información”.

Seguimiento y control del proyecto

Kanban boards



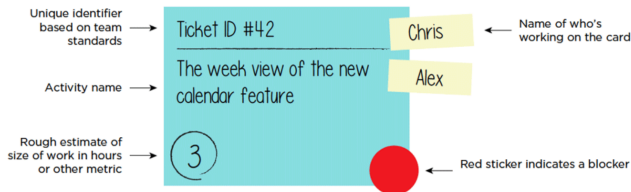
Seguimiento y control del proyecto

Kanban boards



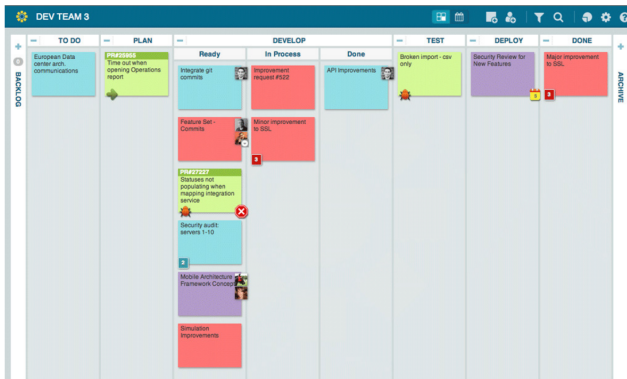
Seguimiento y control del proyecto

Kanban boards



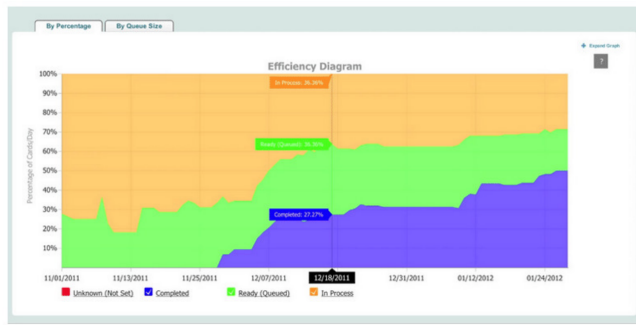
Seguimiento y control del proyecto

Kanban boards



Seguimiento y control del proyecto

Kanban boards



Seguimiento y control del proyecto

NASA Software Engineering Laboratory Recommended Approach to Software Development

Para el éxito de un proyecto:

- ▶ Cree y siga un plan de desarrollo de software (SDP).
- ▶ Aproveche el potencial humano de su equipo.
- ▶ Minimice la burocracia.
- ▶ Establezca los requisitos y gestiona sus cambios.
- ▶ Evalúe periódicamente la salud y el progreso del proyecto (y replanifique cuando sea necesario).

Seguimiento y control del proyecto

NASA Software Engineering Laboratory Recommended Approach to Software Development

Para el éxito de un proyecto:

- ▶ Revise las estimaciones de tamaño, esfuerzo y plan temporal periódicamente, sin insistir en mantener las estimaciones iniciales.
- ▶ Defina y gestione las transiciones de una fase a otra del proyecto de desarrollo de software.
- ▶ Fomente el espíritu de equipo.
- ▶ Comience el proyecto con un equipo pequeño formado por personal experimentado.

Seguimiento y control del proyecto

NASA Software Engineering Laboratory Recommended Approach to Software Development

Para evitar el fracaso de un proyecto:

- ▶ No permita trabajar de forma no sistemática. Aunque se trate de un trabajo creativo, existen principios y práctica útiles cuyo uso es beneficioso.
- ▶ No establezca objetivos poco razonables.
- ▶ No implemente cambios sin evaluar su impacto y requerir su aprobación.
- ▶ No implemente lo que no sea necesario: pequeños cambios que parecen mejorar el sistema tienden a aumentar su complejidad.

Seguimiento y control del proyecto

NASA Software Engineering Laboratory Recommended Approach to Software Development

Para evitar el fracaso de un proyecto:

- ▶ No añada más personal del necesario, especialmente al comienzo del proyecto. Añádalo sólo cuando tenga trabajo útil que hacer para ellos.
- ▶ No asuma que incumplir un plazo intermedio se puede corregir más adelante.
- ▶ No relaje sus estándares para recortar costes o acortar plazos, tiende a introducir errores y desmotiva.
- ▶ No asuma que mucha documentación garantiza algo.

Retrospectivas del proyecto

Son necesarias para la introducción de medidas de mejora continua: los procesos y prácticas utilizados se evalúan y ajustan reflexionando sobre lo acontecido durante la ejecución del proyecto.

Retrospectivas del proyecto

Se debe establecer un mecanismo consistente mediante el que realizar un análisis postmortem del proyecto (o de una iteración):

- ▶ Extraer lecciones de lo sucedido.
- ▶ Establecer mecanismos de mejorar de cara al futuro.

Retrospectivas del proyecto

Uso de métricas

- ▶ **Ley de Gilb**

“Anything you need to quantify can be measured in some way that is superior to not measuring it at all”.

Retrospectivas del proyecto

Dimensiones del proyecto y métricas, ejemplo

Dimensión	Métrica
Funcionalidad	Casos de uso completados (velocidad)
Calidad	Problemas identificados en las pruebas
Herramientas	Coste Mensual
Personal	Obstáculos encontrados pr los miembros del equipo de desarrollo
Tiempo	Días que quedan hasta la próxima relea-se
Proceso	Listas de comprobación completadas
Valor	Incremento en el uso del sistema

Retrospectivas del proyecto

¿Cómo sacar provecho a la retrospectiva?

- ▶ Idealmente, deberían participar todos los interesados en el proyecto, desde que se hizo la oferta.
- ▶ Con realizar un listado de puntos positivos y negativos suele ser necesario. El objetivo no es obtener acciones concretas, sino recolectar y destacar los distintos aspectos que nos permitan un buen desarrollo en futuros proyectos.
- ▶ Se genera un ambiente de confianza y empatía, uniendo al equipo.

Bibliografía

Libros de texto:

- ▶ Roger S. Pressman: **Software Engineering: A Practitioner's Approach** McGraw-Hill, 8th edition, 2014. ISBN 0078022126
- ▶ Shari Lawrence Pfleeger & Hoanne M. Atlee: **Software Engineering: Theory and Practice** Prentice Hall, 4th edition, 2009, ISBN 0136061699
- ▶ Ian Sommerville: **Software Engineering** Pearson, 10th edition, 2015. ISBN 0133943038

Bibliografía

Lecturas recomendadas

- ▶ Dwayne Phillips: **The Software Project Manager's Handbook: Principles That Work at Work** Wiley / IEEE Computer Society, 2nd edition, 2004, ISBN 0471674206
- ▶ Donald J. Reifer (editor): **Software Management** Wiley / IEEE Computer Society, 7th edition, 2006 ISBN 0471775622
- ▶ Richard H. Thayer (editor): **Software Engineering Project Management** Wiley / IEEE Computer Society, 2nd Edition, 2000, ISBN 0818680008

Métricas

- ▶ Lawrence H. Putnam & Ware Myers: **Five Core Metrics: The intelligence behind successful software management** Dorset House, 2003. ISBN 0932633552
- ▶ Norman E. Fenton & Shari Lawrence Pfleeger: **Software Metrics: A rigorous and practical approach, revised** Course Technology, 1998, ISBN 0534954251
- ▶ Tom DeMarco: **Controlling Software Projects: Management, measurement & estimation** Prentice-Hall PTR, 1982. ISBN 013171711

Métricas

- ▶ **Software Measurement Guidebook** NASA Software Engineering Laboratory, SEL-94-102, rev.1, 1995.
- ▶ Lawrence H. Putnam & Ware Myers: **Measures for Excellence: Reliable software on time, within budget** Yourdon Press, 1992, ISBN 0135676940
- ▶ Robert B. grady & Deborah L. Caswell: **Software Metrics: Establishing a Company-Wide Program** Prentice Hall PTR, 1987. ISBN 0138218447
- ▶ Capers Jones: **Programming Productivity** McGraw-Hill, 1986, ISBN 0070328110

Bibliografía

Clásicos

- ▶ Freferic P. Brooks, Jr.: **The Mythical Man-Month: Essays on Software Engineering** Addison-Wesley, 1995 ISBN 0201835959
- ▶ Alan M. Davis: **201 Principles of Software Development** McGraw-Hill, 1995. ISBN 0070158401
- ▶ Barry W. Boehm: **Software Engineering Economics** Prentice-Hall PTR, 1991, ISBN 0138221227
- ▶ **Manager's Handbook for Software Development** NASA Software Engineering Laboratory, SEL-84-101, rev.1, 1990.
- ▶ **Software Engineering Laboratory (SEL) Relationships, Models and Managemenet Rules** NASA Software Engineering Laboratory, SEL-91-001, 1991

Planificación y Gestión de Proyectos Informáticos

Seguimiento y control de proyectos

Jose M. Moyano

jmoyano@ugr.es

Departamento de Ciencias de la Computación e Inteligencia Artificial

Universidad de Granada

Curso 2022-23



DaSCI

Instituto Andaluz de Investigación en
Data Science and Computational Intelligence



**UNIVERSIDAD
DE GRANADA**