

---

# Tècniques de direcció de projectes

---

PID\_00264669

Ramon G. Sedó  
Laura Benítez García

---

Temps mínim de dedicació recomanat: 4 hores

---



**Ramon G. Sedó**

**Laura Benítez García**

La revisió d'aquest recurs d'aprenentatge UOC ha estat coordinada per la professora: Laura Porta Simó (2019)

Tercera edició: setembre 2019  
© Ramon G. Sedó, Laura Benítez García  
Tots els drets reservats  
© d'aquesta edició, FUOC, 2019  
Av. Tibidabo, 39-43, 08035 Barcelona  
Realització editorial: FUOC

*Cap part d'aquesta publicació, incloent-hi el disseny general i la coberta, no pot ser copiada, reproduïda, emmagatzemada o transmesa de cap manera ni per cap mitjà, tant si és elèctric com químic, mecànic, òptic, de gravació, de fotocòpia o per altres mètodes, sense l'autorització prèvia per escrit dels titulars dels drets.*

# Índex

<b>Introducció.....</b>	<b>5</b>
<b>1. Estructura de descomposició del treball.....</b>	<b>7</b>
1.1. Descripció .....	7
1.2. Notació .....	8
<b>2. Anàlisi de punts funció.....</b>	<b>9</b>
2.1. IFPUG 4.4 .....	10
<b>3. Program Evaluation and Review Technique (PERT).....</b>	<b>11</b>
3.1. Descripció .....	11
3.2. Notació .....	11
3.3. Exemple .....	14
<b>4. Mètode del camí crític.....</b>	<b>18</b>
4.1. Descripció .....	18
4.2. Exemple .....	18
<b>5. El diagrama de Gantt.....</b>	<b>21</b>
5.1. Descripció .....	21
5.2. Notació .....	21
5.3. Exemple .....	21
<b>6. Tècniques d'assignació de recursos.....</b>	<b>23</b>
6.1. Patró de límits .....	24
6.2. Histograma de recursos .....	25
6.3. Planificació d'activitats i recursos .....	26
<b>7. Diagrama de Pareto.....</b>	<b>27</b>
<b>8. Diagrama causa-efecte.....</b>	<b>30</b>
<b>9. Gestió clàssica de projectes i noves maneres de plantejar-se la gestió de projectes.....</b>	<b>33</b>
9.1. Scrum. La conceptualització del producte. El dia a dia en un projecte àgil .....	38
9.2. Kanban com a sistema de gestió del treball en curs .....	41
<b>Bibliografia.....</b>	<b>47</b>



## Introducció

La **gestió de projectes** és un conjunt d'activitats específiques que s'empren per a administrar el projecte. Aquestes activitats comprenen diversos aspectes:

- Estimar l'esforç necessari per a desenvolupar un projecte.
- Planificar tasques i recursos.
- Controlar tasques.
- Seguir el projecte.
- Controlar les incidències.
- Controlar canvis.

Per a desenvolupar aquestes activitats és necessari utilitzar tècniques específiques. Les tècniques presentades en aquest mòdul són les més usades en la gestió de projectes. La majoria de les tècniques presentades estan basades en les descrites a MÉTRICA Versión 3, metodologia desenvolupada pel Ministeri d'Administracions Públiques espanyol, per al procés de planificació de sistemes d'informació. La resta de les pràctiques estan basades en altres metodologies de la gestió de projectes, com el Project Management Book Of Knowledge (PMBOK) del Project Management Institute (PMI).



# 1. Estructura de descomposició del treball

La tècnica en què es basa la planificació de la majoria dels projectes és la tècnica denominada **estructura de descomposició del treball**<sup>1</sup> (EDT).

<sup>(1)</sup>En anglès, WBS (*work breakdown structure*).

A partir de l'EDT, es dedueixen els diagrames de PERT i de Gantt, que permeten planificar el projecte.

En l'organització d'un projecte, aquesta tècnica permet estructurar les activitats, de manera que serveix de **llista de comprovació** de les tasques que s'han de fer en un projecte i també d'eina de comptabilitat analítica del projecte. Si a cada tasca s'hi assigna el cost corresponent que té, permet portar un control comptable del projecte.

## 1.1. Descripció

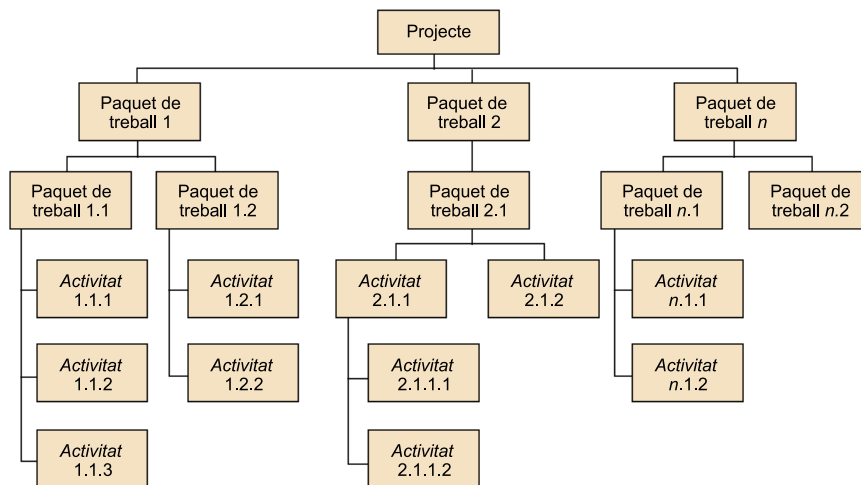
L'EDT presenta una descomposició de les activitats d'un projecte segons la naturalesa que tenen. Es representa com un gràfic jeràrquic, un arbre, un aplegador d'activitats: desenvolupament, qualitat, gestió, etc.

Els grups d'activitats serveixen de suport per a seguir el projecte, tant en temps com en costos, i constitueix una eina històrica per a projectes futurs.

El desglossament del treball en grups d'activitats es fa fins que s'obtenen activitats que permetin entendre el treball que s'ha de fer i l'estimació dels temps i costos de l'elaboració d'aquest treball.

El gràfic està ordenat de menys detall a més detall. L'ha construït l'equip de projecte i assegura que tothom sàpiga la feina requerida.

## 1.2. Notació



En aquest exemple, les activitats o tasques del projecte s'agrupen en  $n$  grups d'activitats principals formades per agrupacions d'activitats específiques.

El paquet de treball 1 està format per dos paquets de treball, 1.1 i 1.2, que engloben tres activitats en el cas del paquet 1.1 i dues activitats en el cas del paquet 1.2.

El paquet de treball 1 podria ser el de les activitats d'anàlisi; el paquet 2, el de les activitats de disseny; el 3, el de les de desenvolupament, i així successivament.



## 2. Anàlisi de punts funció

L'anàlisi de punts funció és una de les tècniques més utilitzades en l'estimació de la durada de projectes per al Web.

Les tècniques d'estimació tenen com a objectiu calcular el cost total del desenvolupament d'un projecte.

L'estimació del cost dels productes de programari és una de les activitats més difícils i propenses a l'error. És difícil fer una estimació exacta de cost al començament d'un desenvolupament a causa del gran nombre de factors coneguts o esperats que en determinen la complexitat i desconeguts o no esperats que es produiran en qualsevol moment que en determinen la incertesa.

Les tècniques d'estimació ajuden en aquesta tasca i donen com a resultat un nombre d'hores d'esforç, a partir de les quals es calcula el cost corresponent. L'estimació ens aporta un nombre d'hores aproximat que cal combinar amb els recursos per a obtenir la planificació d'activitats pel que fa al temps i establir les fites del projecte.

Les tècniques d'estimació més fiables es basen en l'**anàlisi de punts funció**. La tècnica de punts funció permet avaluar un sistema d'informació a partir d'un coneixement mínim de les funcionalitats i entitats que intervenen.

Les característiques més destacables d'aquesta tècnica són:

- És una unitat de mesura empírica.
- Preveu el sistema com un tot que es divideix en determinades funcions.
- És independent de l'entorn tecnològic en què s'ha de desenvolupar el sistema.
- És independent de la metodologia que s'utilitzarà.
- És independent de l'experiència i de l'estil de programació.
- És fàcil d'entendre per a l'usuari.

El resultat de l'aplicació d'aquesta tècnica es dona en **punts funció**, que posteriorment s'han de passar a dies d'esforç, per a la qual cosa sí que s'han de tenir en compte l'experiència de l'equip de desenvolupament i l'estil de programació, l'aplicació d'una metodologia o l'altra i la tecnologia.

Aquest càlcul de dies per punt funció s'ha de basar en l'experiència adquirida en la valoració i elaboració de sistemes anteriors, i s'ha d'actualitzar el valor de conversió després d'haver acabat cada projecte.

Entre les tècniques d'estimació basades en l'anàlisi de punts funció, l'IFPUG 4.4 (*unadjusted functional size measurement method*), també definit com la norma ISO/IEC 20926:2003 *Software engineering* definit per l'International Function Point Users Group i evolucionat a partir de la proposta original d'Allan Albrecht a IBM, s'està convertint en l'estàndard *de facto* en el càlcul de punts funció.

### **2.1. IFPUG 4.4**

Definit pels seus creadors com el "mètode estàndard per a mesurar el desenvolupament de programari des del punt de vista de l'usuari".

El mètode es basa principalment en la identificació dels components del sistema informàtic en termes de transaccions i grups de dades lògiques que són rellevants per a l'usuari en el seu negoci. A cadascun d'aquests components hi assigna un nombre de punts per funció basant-se en el tipus de component i la complexitat que té, i la sumatòria d'això ens dona els punts de funció sense ajustar. L'ajust és un pas final que es basa en les característiques generals de tot el sistema informàtic que es comptabilitza.

### 3. Program Evaluation and Review Technique (PERT)

L'objectiu del PERT<sup>2</sup> és establir les dependències entre les diferents tasques del projecte per a saber de quina manera s'han d'encadenar aquestes tasques en la planificació. Aquestes dependències o prelacions s'estableixen a partir de les precedents.

<sup>(2)</sup>Program Evaluation and Review Technique.

#### 3.1. Descripció

El mètode PERT parteix de la descomposició del projecte en una sèrie d'obres parcials o activitats, entenent per **activitat** l'execució d'una tasca, que exigeix per a elaborar-la la utilització de recursos tals com recursos humans o programari i maquinari.

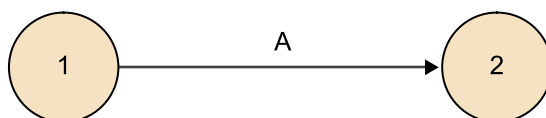
Així, per exemple, l'elaboració del brífing, la recopilació d'informació, el desenvolupament de continguts, el disseny de l'arquitectura del lloc web, el desenvolupament gràfic, la programació, la contractació del servei d'hostatge i la configuració d'aquest servei, l'etapa de proves i el llançament són activitats que formen part del projecte de construcció d'un lloc web.

Després del concepte d'activitat, el mètode PERT estableix el concepte d'incidència. Una **incidència** és un esdeveniment, un punt en el temps, una data en el calendari. La incidència no consumeix recursos, només indica el principi o el final d'una activitat o d'un conjunt d'activitats.

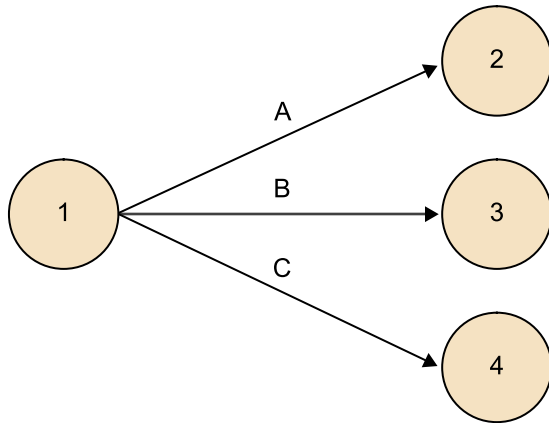
#### 3.2. Notació

Per a representar les diferents activitats en què es descompon un projecte, i també les incidències corresponents, s'utilitza una **estructura de graf**. Els arcs del graf representen les activitats i els vèrtexs, les incidències.

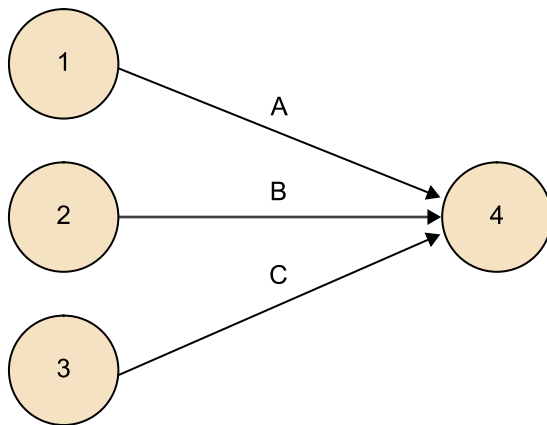
Així, el vèrtex 1 del diagrama que apareix a continuació indica la incidència inici de l'activitat A i el vèrtex 2, la incidència final d'aquesta activitat. L'arc que uneix els vèrtexs 1 i 2 representa l'elaboració mateixa de l'activitat.



D'altra banda, com s'ha dit, una incidència pot representar el principi o final d'un conjunt d'activitats. En efecte, en la figura següent, el vèrtex 1 representa la incidència inici de les activitats A, B i C.

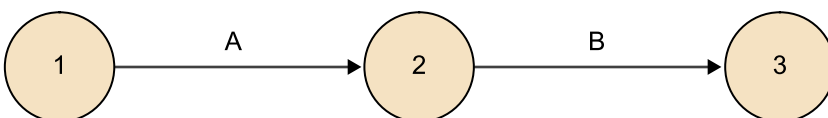


En canvi, en la figura que apareix a continuació, el vèrtex 4 representa la incidència final de les activitats A, B i C.

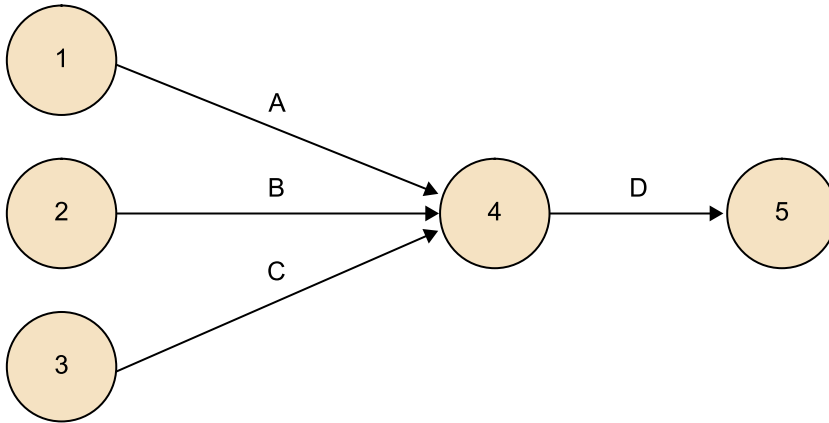


Una vegada descompost el projecte en activitats, la fase següent del mètode PERT consisteix a establir les relacions existents entre les diferents activitats. Aquestes relacions indiquen l'ordre en què s'han d'executar aquestes activitats. El cas més senzill són les **relacions lineals**, que es presenten quan, per a començar una activitat, és necessari que s'hagi acabat una única activitat (la precedent).

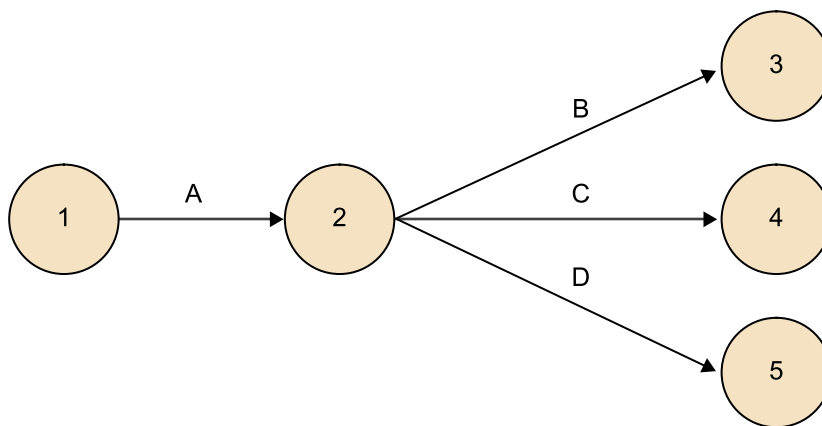
Si ens fixem en la figura següent s'aprecia que per a començar l'activitat B és necessari que s'hagi acabat l'activitat A. És a dir, el vèrtex 2 representa la incidència final de l'activitat A i, alhora, la incidència inicial de l'activitat B.



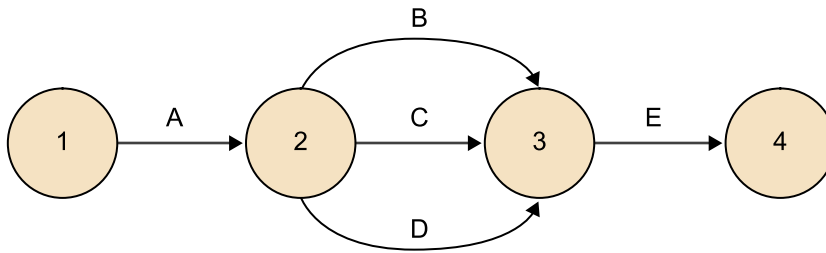
A continuació, s'estudia el cas de les prelacions que originen una **convergència**. En el diagrama següent es representa aquest cas. Per a començar l'activitat D és necessari que s'hagin acabat les activitats A, B i C. És a dir, el vèrtex 4 representa la incidència final de les activitats A, B i C i, alhora, la incidència inicial de l'activitat D.



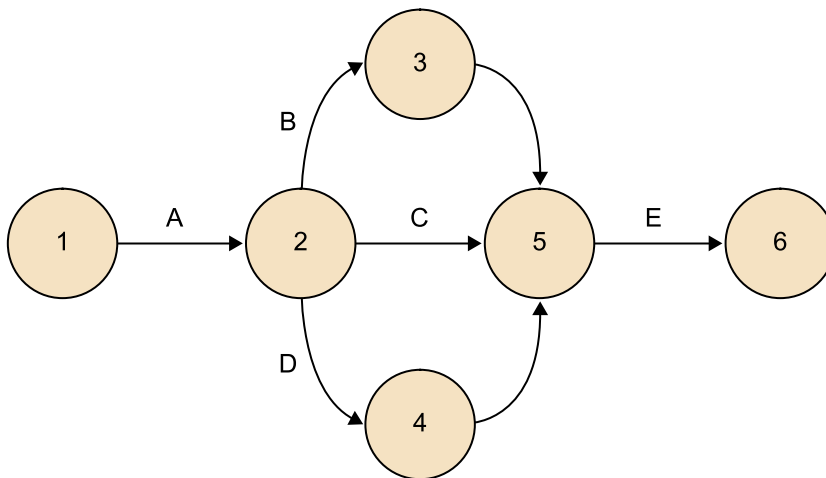
El cas oposat a l'anterior és el de les **prelacions que originen una divergència**, representat en el diagrama que apareix a continuació. En aquest cas, per a començar les activitats B, C i D és necessari que s'hagi acabat l'activitat A.



Vegem ara les poblacions següents: l'activitat A és anterior a les activitats B, C i D i aquestes activitats (B, C i D) són anteriors a l'activitat E. Una manera de reflectir-les és la que es representa en el diagrama següent. S'hi mostren les prelacions anteriors correctament. Però aquesta representació pot resultar confusa, ja que les activitats B, C i D tenen el mateix origen i la mateixa destinació.



Per a resoldre aquest problema es pot recórrer a les activitats fictícies, com es mostra en el diagrama següent, on es distingeixen perfectament les activitats B, C i D, ja que les tres activitats, fins i tot naixent en el mateix vèrtex, moren en vèrtexs diferents.



### 3.3. Exemple

Amb l'exemple següent, es pot construir el diagrama de PERT d'un projecte complet. Per a això, s'ha de començar per introduir els conceptes d'incidència inicial del projecte i d'incidència final del projecte.

S'entén per **incidència inicial del projecte** aquella que, representant el començament d'una activitat o de diverses, no en representa, tanmateix, el final de cap. En canvi, la **incidència final del projecte** és aquella que, representant el final d'una activitat o de més d'una, no en representa, tanmateix, el començament de cap. Aquesta incidència es reconeix en el graf perquè està representada per l'únic vèrtex al qual arriben, però no en surten, arcs.

A continuació es construirà el graf de PERT d'un projecte les activitats i prelacions del qual són:

- A precedeix C, D, E.
- B precedeix C.
- C precedeix K.
- D precedeix F, G.
- E precedeix J.

- F precedeix I.
- G precedeix H.
- H, I, J precedeixen L.
- K precedeix M.
- L precedeix P.
- M precedeix N.
- N, P precedeixen Q.
- Q precedeix R.

Per a construir el graf de PERT, el primer pas ha de ser recollir d'una manera sistematitzada la informació continguda en l'anterior conjunt de prelacions. Per a això, hi ha bàsicament dos procediments: la matriu d'encadenaments i el quadre de prelacions.

La **matriu d'encadenaments** consisteix en una matriu quadrada la dimensió de la qual és igual al nombre d'activitats en què s'ha descompost el projecte. Quan un element d'aquesta matriu apareix marcat amb una X, indica que per a començar l'activitat que correspon a la fila que creua aquest element és necessari que s'hagi acabat l'activitat que correspon a la columna que creua l'element.

	Activitats precedents																
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	P	Q	R
A																	
B																	
C	X	X															
D	X																
E	X																
F				X													
G				X													
H							X										
I						X											
J					X												
K			X														
L								X	X	X							
M											X						
N													X				
P												X					
Q														X	X		

	Activitats precedents																
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	P	Q	R
R																X	

És interessant observar que aquelles files de la matriu en què no apareix cap X indiquen les activitats que no tenen cap precedent. És a dir, aquelles activitats la incidència inicial de les quals coincideix amb la incidència inicial del projecte (activitats A i B). D'altra banda, aquelles columnes en què no apareix cap X indiquen les activitats que no tenen cap activitat següent, és a dir, aquelles activitats la incidència final de les quals coincideix amb la incidència final del projecte (activitat R).

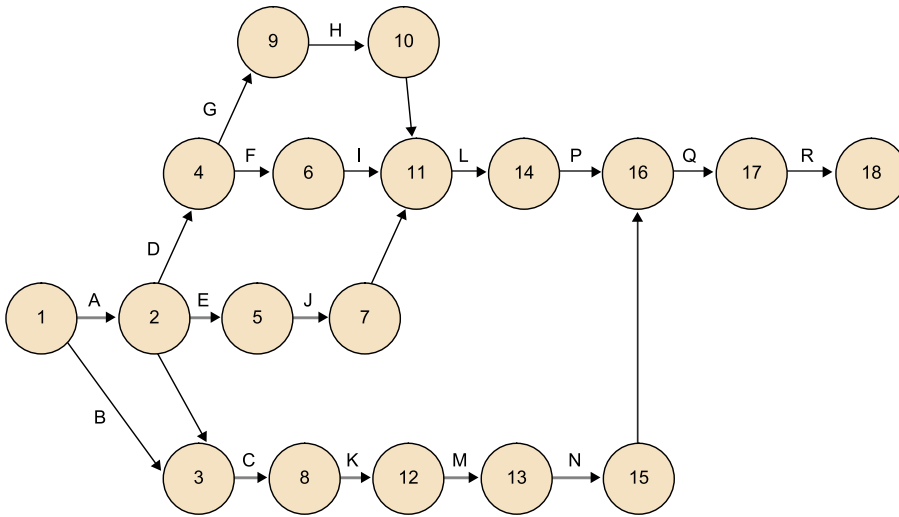
El quadre de prelacions està format per dues columnes. A la primera columna hi ha representades totes les activitats en què s'ha descompost el projecte. A la segona columna hi figuren les activitats precedents de la seva homòloga a la primera columna. Amb les dades de l'exemple s'elabora el quadre de prelacions, que és el següent:

Activitat	Precedent
A	
B	
C	A, B
D	A
E	A
F	D
G	D
H	G
I	F
J	E
K	C
L	H, I, J
M	K
N	M
P	L
Q	N, P
R	Q



L'activitat o activitats inicials del projecte es reconeixen en el quadre de prelacions perquè no tenen cap activitat precedent (activitats A i B). L'activitat o activitats finals del projecte es reconeixen perquè no apareixen a la columna (2) del quadre (activitat R).

A partir de la matriu d'encadenaments o del quadre de prelacions es construeix fàcilment el graf de PERT corresponent. Per a les dades de l'exemple, el graf de PERT està representat en el diagrama següent.



## 4. Mètode del camí crític

El mètode del camí crític està format per una sèrie de tasques que s'han de completar a temps perquè un projecte s'acabi conforme a la programació.

### 4.1. Descripció

La majoria de les tasques d'un projecte presenten un cert temps de demora (folgança) i, per tant, es poden retardar lleugerament sense afectar la data de fi del projecte. Les tasques que no es poden retardar sense que es modifiqui la data de fi del projecte es denominen *tasques crítiques*.

Per a calcular la folgança de les tasques, el mètode del camí crític calcula totes les dates d'inici i fi teòriques primerenques i tardanes per a totes les activitats de la xarxa sense considerar limitacions de recursos.

Les **dates primerenques** indiquen la primera data en què es pot començar o acabar una tasca segons les relacions de precedència que té o segons restriccions temporals. Les **dates tardanes** indiquen l'última data en què es pot començar o acabar una tasca segons aquestes mateixes relacions i restriccions.

La folgança es determina fent una anàlisi de recorregut cap endavant i una anàlisi de recorregut cap endarrere.

Les dates d'inici i fi primerenques i tardanes resultants indiquen els períodes dins dels quals s'ha de programar l'activitat.

Les tasques que formen part del denominat *camí crític* són les tasques sense folgança, les tasques crítiques, la modificació de les quals per a resoldre sobreassignacions o altres problemes de la programació afecten la data de fi del projecte.

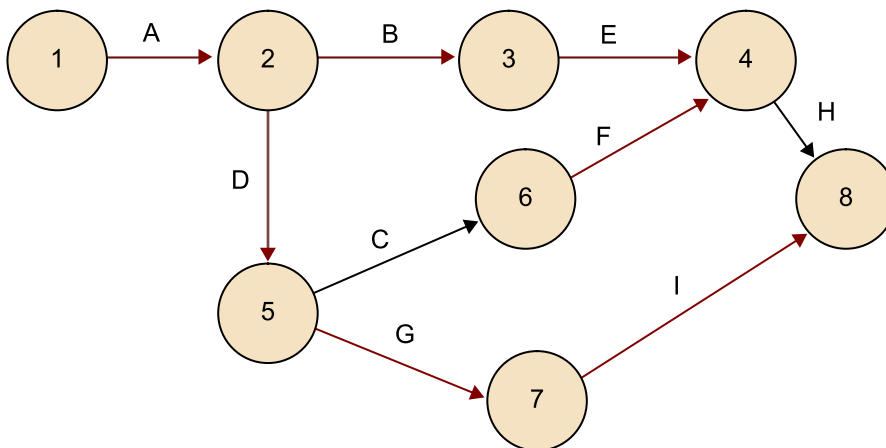
En un projecte, el camí crític és el recorregut de més durada i pot no ser únic, és a dir, en un mateix projecte podem tenir més d'un camí crític.

### 4.2. Exemple

Una empresa desitja desenvolupar un lloc web nou per al seu portal d'empleats. Les tasques descrites per a fer-ne el desenvolupament i les durades que han de tenir són les següents:

Tasca	Després de	Durada de la tasca
A	-	7 dies
B	A	3 dies
C	D	2 dies
D	A	4 dies
E	B	2 dies
F	C	2 dies
G	D	6 dies
H	E-F	1 dia
I	G	5 dies

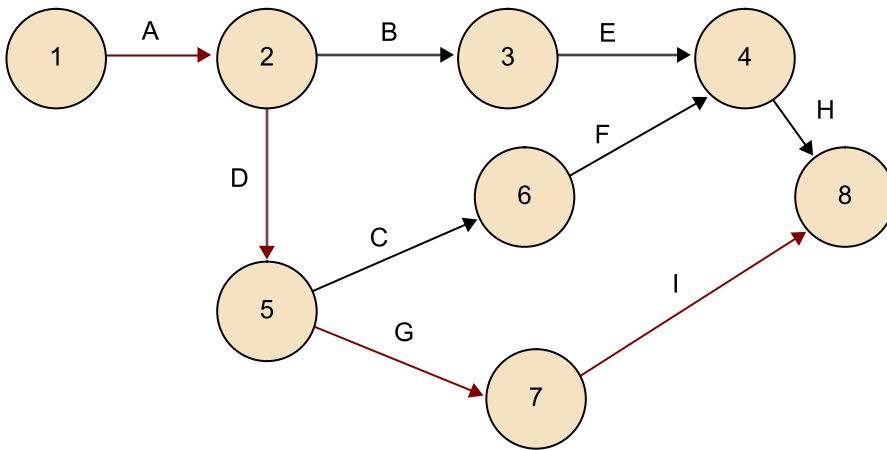
El graf de PERT del projecte és el següent:



Els camins possibles per a arribar del node 1 al node 8 són:

- A-D-G-I = 7 + 4 + 6 + 5 = 22 dies
- A-D-C-F-H = 7 + 4 + 2 + 2 + 1 = 16 dies
- A-B-E-H = 7 + 3 + 2 + 1 = 13 dies

De manera que el camí crític és el format per les tasques A, D, G, I.



Si pintem les tasques dins d'un cronograma obtenim el següent:

<b>A</b>	█																							
<b>B</b>								█																
<b>C</b>																								
<b>D</b>								█																
<b>E</b>																								
<b>F</b>																								
<b>G</b>																								
<b>H</b>																								
<b>I</b>																								
	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>	<b>14</b>	<b>15</b>	<b>16</b>	<b>17</b>	<b>18</b>	<b>19</b>	<b>20</b>	<b>21</b>	<b>22</b>		

Les tasques B, C, E, F i H d'aquest projecte tenen una certa folgança ja que poden presentar certes modificacions temporals sense afectar la data d'acabament del projecte; en canvi, qualsevol modificació en la durada de les tasques A, D, G i I influiria directament en l'alteració de la data de fi de projecte.

## 5. El diagrama de Gantt

El **diagrama de Gantt** o **cronograma** té com a objectiu representar el pla de treball, i mostrar les tasques que s'han de fer, el moment de començar-les i acabar-les i la manera en què estan encadenades entre si.

### 5.1. Descripció

El gràfic de Gantt és la manera habitual de presentar el pla d'execució d'un projecte, recollint a les files la relació d'activitats que s'han de fer i a les columnes l'escala de temps que es maneja, mentre que la durada i situació en el temps de cada activitat es representa amb una línia dibuixada al lloc corresponent.

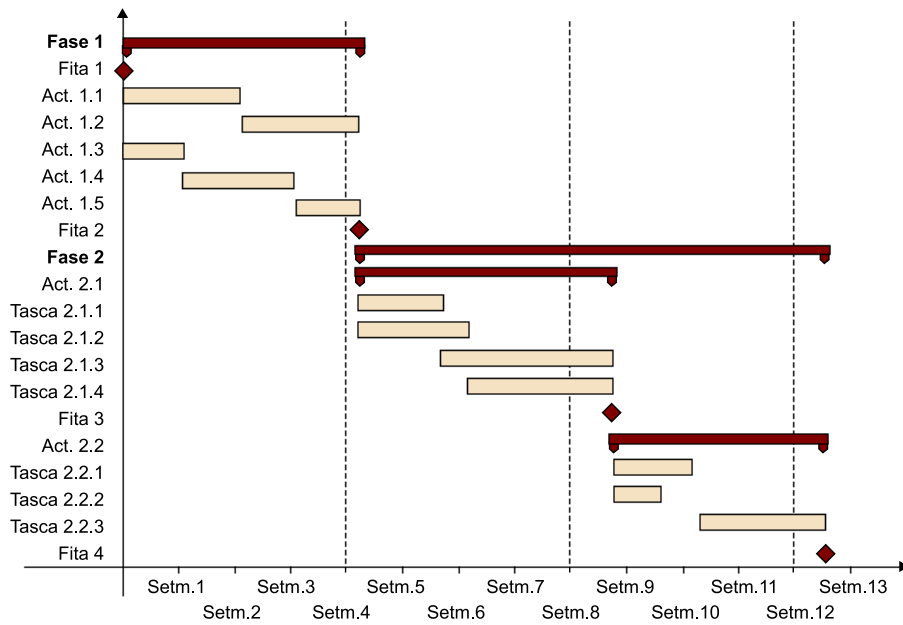
### 5.2. Notació

- És un model fet sobre eixos de coordenades, on les tasques se situen sobre l'eix d'ordenades ( $y$ ) i els temps, sobre el d'abscisses ( $x$ ).
- Cada activitat es representa amb una barra limitada per les dates previstes de començament i final.
- Les activitats s'agrupen en fases i es poden descompondre en tasques.
- Cada activitat ha de tenir recursos associats.
- Les fites són un tipus d'activitat que no representa treball ni té recursos associats.
- Les activitats es poden encadenar per dos motius:
  - **Encadenament funcional o per prelacions.** Per exemple, un programa no es pot provar fins que s'ha escrit.
  - **Encadenament orgànic o per ocupació de recursos.** Per exemple, un programador no pot començar un programa fins que s'ha acabat l'anterior.

Es poden fer activitats en paral·lel sempre que no tinguin dependència funcional o orgànica.

### 5.3. Exemple

El diagrama següent és un exemple típic d'un diagrama de Gantt, en què apareixen fases, activitats, tasques i fites.



Perquè un Gantt sigui realista i fiable ha d'anar acompanyat d'un gràfic que reflecteixi l'activitat dels tècnics (histograma de recursos) que componen l'equip del projecte.

A la xarxa hi ha disponible una gran quantitat d'eines i software que permeten desenvolupar aquests diagrames de forma ràpida i àgil (com per exemple: Gantter).

## 6. Tècniques d'assignació de recursos

L'assignació de recursos és una tasca fonamental en la planificació, ja que cal considerar aspectes tècnics de cada recurs com la disponibilitat que tenen, la capacitat de treball o els impediments horaris.

És fonamental que els treballs es descomponguin fins a la unitat mínima de tractament, és a dir, s'ha de descompondre el projecte en fases, les fases en activitats i les activitats en tasques, i assignar una tasca a un recurs. No s'ha de caure en l'error d'assignar una activitat a diversos recursos. Si no es fa així, és molt difícil preveure l'ocupació plena de tots els recursos, i es donen situacions anòmales, com és ara que un recurs tingui una ocupació molt baixa i un altre, una ocupació excessiva. Convé que la planificació estigui perfectament depurada, ja que altrament es produeixen els problemes següents:

- Es dilata innecessàriament el termini de lliurament.
- Es reté sense feina un nombre considerable d'usuaris que podrien treballar en un altre projecte.
- S'encareix considerablement el cost del projecte, ja que es tenen assignats uns recursos "que no treballen" tot el temps.

Per això és necessari tenir en consideració els aspectes següents a l'hora d'assignar recursos:

- Quantificar necessitats i dates d'incorporació de recursos.
- Obtenir un patró que marqui els límits del projecte.
- Considerar la capacitat, els coneixements i l'experiència de cada recurs.
- Considerar la complexitat, la mida i els requeriments tècnics de cada tasca.
- Assignar tasques senzilles a recursos amb poca experiència. Si s'assignen tasques senzilles a recursos amb molta experiència, s'infrutilitzen.
- Assignar tasques complexes a recursos amb molta experiència. Si s'assignen les tasques complexes a recursos amb poca experiència, perdran molt temps preguntant als seus companys i, el que és més greu, faran perdre molt temps a la resta de l'equip.

- Construir l'histograma de recursos, per a veure la coherència de les assignacions.
- Tractar d'assignar una tasca a un recurs únic, descomponent tot el que sigui necessari.
- Vigilar que no hi hagi buits a l'histograma.

### 6.1. Patró de límits

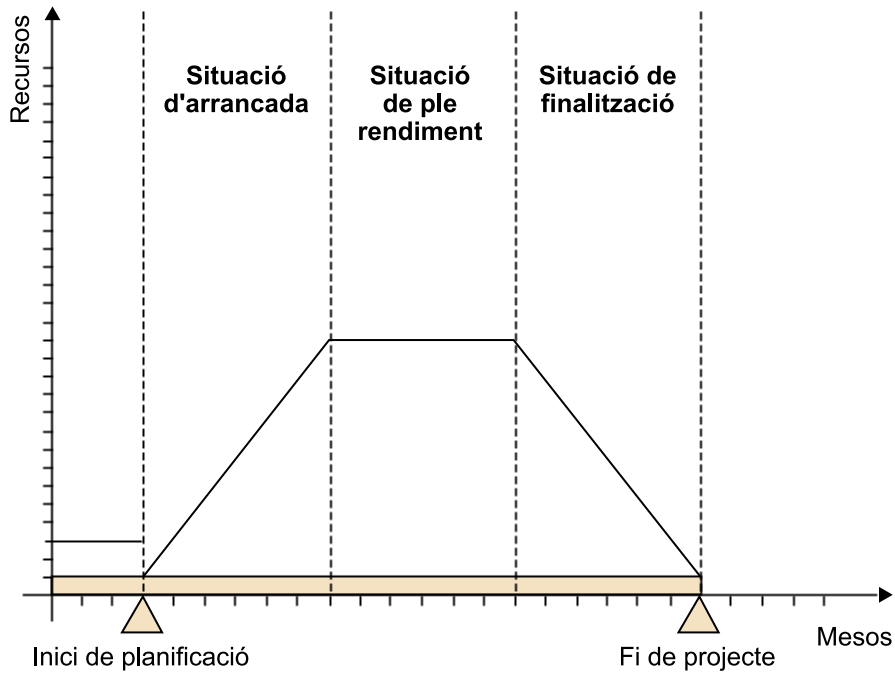
La tècnica de patró de límits té com a objectiu establir els límits de recursos aproximats. Per a això, una vegada sabuts per l'estimació l'esforç total i el termini de lliurament, cal fer les operacions següents:

- Establir l'esforç en mesos (amb decimals).
- Deducir la part corresponent a disseny, en la qual es decideixen els continguts, els serveis i les funcionalitats, ja que és una fase amb un tipus d'activitats diferents de la resta.
- Establir la durada en mesos (amb decimals). Normalment aquesta dada se sap pel compromís adquirit.
- Deducir la durada corresponent a l'anàlisi d'activitats del projecte.
- Considerar que tot projecte té tres situacions clarament diferenciades:
  - **Posada en marxa.** Durant aquesta situació s'incorporen gradualment recursos al projecte fins a assolir-ne el nombre màxim.
  - **Ple rendiment.** Aquesta és una situació d'estabilitat quant al nombre de recursos.
  - **Acabament.** Quan es van acabant les tasques, comencen a abandonar el projecte els tècnics gradualment.
- Seguidament, cal distribuir els recursos:
  - Una vegada obtingut el nombre de recursos mitjans, s'han de considerar les tres situacions que presenta el projecte: posada en marxa, ple rendiment i acabament.
  - S'ha de considerar que aquestes tres fases tenen una durada aproximadament igual.
  - S'ha de considerar que el cap de projecte és present durant tot el cicle de vida.
  - S'ha d'extreure de la part de disseny l'esforç corresponent al cap de projecte.
  - S'ha de dividir la durada estimada (sense la part de disseny) en tres parts iguals.



- S'ha de distribuir la meitat de l'esforç estimat (sense la part d'anàlisi) en la situació de **ple rendiment**.
- S'ha de distribuir l'altra meitat, a parts iguals, entre la situació de **posada en marxa** i la d'**acabament**.

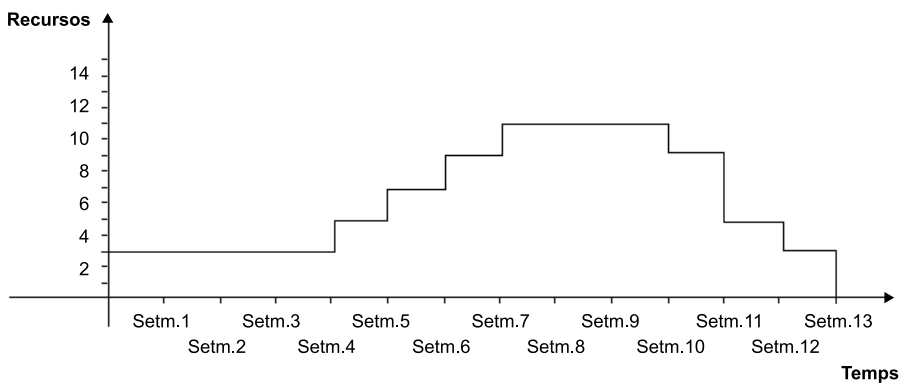
Com a conseqüència d'això s'obté el gràfic que apareix a continuació i que serveix per a establir els límits de l'assignació de recursos.



## 6.2. Histograma de recursos

L'**histograma de recursos** és un gràfic sobre uns eixos de coordenades: els recursos són sobre l'eix de les ordenades i el temps, sobre l'eix de les abscisses.

A mesura que s'incorporen recursos al projecte, augmenta el gràfic, i al contrari quan són desassignats.



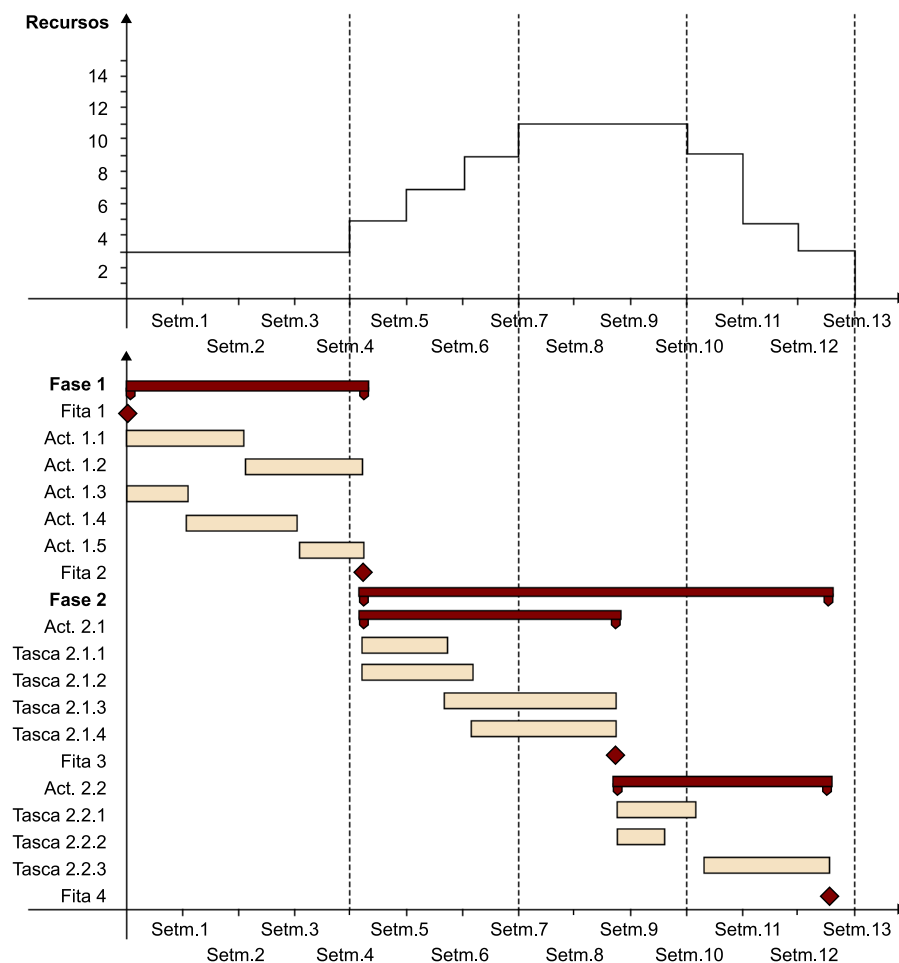
Exemple gràfic d'un histograma de recursos

### 6.3. Planificació d'activitats i recursos

Per a fer adequadament la planificació és necessari observar el següent:

- S'han de construir paral·lelament el Gantt i l'histograma de recursos.
- S'han de fer els encadenaments funcionals a partir del PERT.
- S'han d'assignar els recursos:
  - amb criteri de capacitat professional;
  - encadenant-ne les activitats orgànicament.
- S'ha de considerar el màxim de recursos que proporciona el patró de límits per a elaborar l'histograma de recursos.
- L'histograma de recursos ha de reflectir amb exactitud els recursos utilitzats en el Gantt.

Seguidament es pot veure gràficament com s'ha de fer la planificació d'activitats i la de recursos.



## 7. Diagrama de Pareto

El **diagrama de Pareto** és una eina de representació gràfica que identifica els problemes més importants, segons la freqüència d'ocurrència o cost (diners, temps) que tenen i permet establir les prioritats d'intervenció. En definitiva, és un tipus de distribució de freqüències que es basa en el **principi de Pareto**, sovint denominat *regla 80/20*, el qual indica que el 80% dels problemes són originats per un 20% de les causes. Aquest principi ajuda a separar els errors crítics, que normalment solen ser pocs, dels molts no crítics o trivials.

El diagrama de Pareto comunica de manera clara i evident el resultat de l'anàlisi de comparació i prioritització dels problemes més importants.

La construcció del diagrama de Pareto consta de les etapes següents:

- 1) **Decidir com s'han de classificar les dades:** per tipus de defecte (manera molt usual de fer-ho), per màquina, per fase del procés, per torn, etc.
- 2) **Determinar el temps de recollida de les dades:** en termes d'hores, dies, setmanes o mesos.
- 3) **Obtenir les dades i ordenar-les:** en aquesta fase s'ha de preparar el full de recollida de dades.
- 4) **Dibuixar els eixos de coordenades:** es col·loquen, en l'eix vertical, l'escala de mesura de les freqüències o cost i, en l'eix horitzontal, les causes en ordre decreixent de la unitat de mesura.
- 5) **Dibuixar el diagrama:** representació gràfica de les dades recollides en el full.
- 6) **Construir una línia de freqüència acumulada.**
- 7) **Fer l'anàlisi de Pareto:** el diagrama posa en relleu els problemes més importants sobre els quals serà necessari actuar.

En aquest exemple de diagrama de Pareto veiem que el 80% dels errors els causen errors de tipus E, B i C.

Taula de Pareto			
Tipus d'error	Nombre d'errors	% del total	% acumulat del total
E	124	38,0	38,0
B	83	25,5	63,5

Taula de Pareto			
Tipus d'error	Nombre d'errors	% del total	% acumulat del total
C	55	16,9	80,4
F	28	8,6	89,0
D	18	5,5	94,5
A	8	2,5	96,9
G	5	1,5	98,5
H	5	1,5	100
Total	326	100	

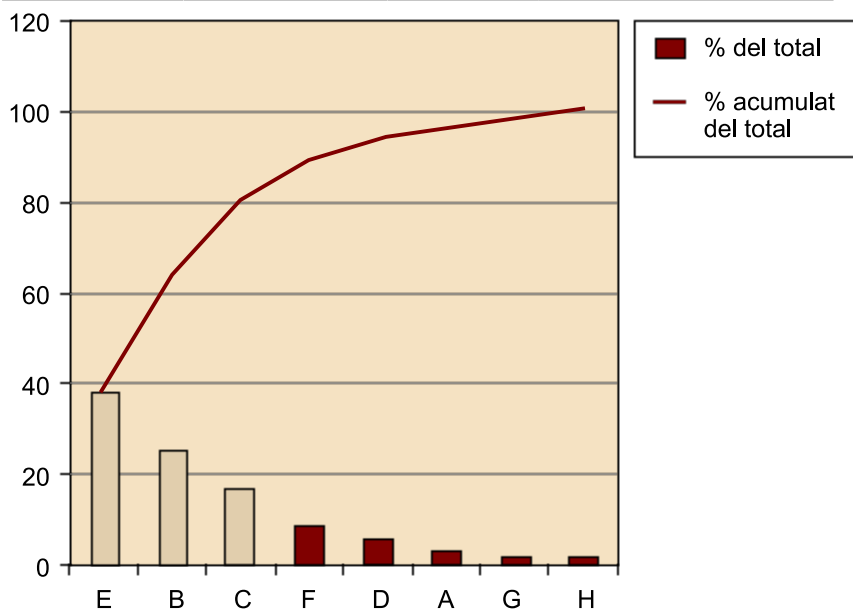


Diagrama de Pareto

Els diagrames de Pareto permeten identificar els problemes més importants i generar nous diagrames de Pareto individuals per a aquests problemes.

Si s'emprenen accions correctores hem de dibuixar els diagrames de Pareto abans i després per a comprovar els resultats assolits.

D'altra banda, sempre resulta molt útil fer l'anàlisi observant el cost dels defectes en termes monetaris, sobretot si es pretenen reduir els costos de la no-qualitat. Per a això, construïm el diagrama de Pareto en termes de cost d'eliminació de cadascun dels defectes o en termes de pèrdues econòmiques que comporten cadascun dels defectes. Aquesta manera de fer ens permet saber si la identificació i eliminació dels problemes o defectes ens permet assolir beneficis enormes o, almenys, no incórrer en grans pèrdues. A vegades, una quantitat petita de defectes provoca pèrdues grans i, en canvi, una quantitat gran de defectes pot provocar pèrdues bastant reduïdes.

La utilització d'aquesta eina presenta els avantatges següents:

- Permet observar els resultats de les accions de millora implantades ja que compara dos diagrames del mateix fenomen en moments diferents de temps.
- És una eina polivalent i fàcil d'aplicar, no solament en el control de la qualitat sinó en qualsevol àmbit.
- Utilitzada en presentacions i reunions augmenta l'eficàcia i la rapidesa de la comunicació ja que permet identificar ràpidament i a simple vista el problema més greu.

## 8. Diagrama causa-efecte

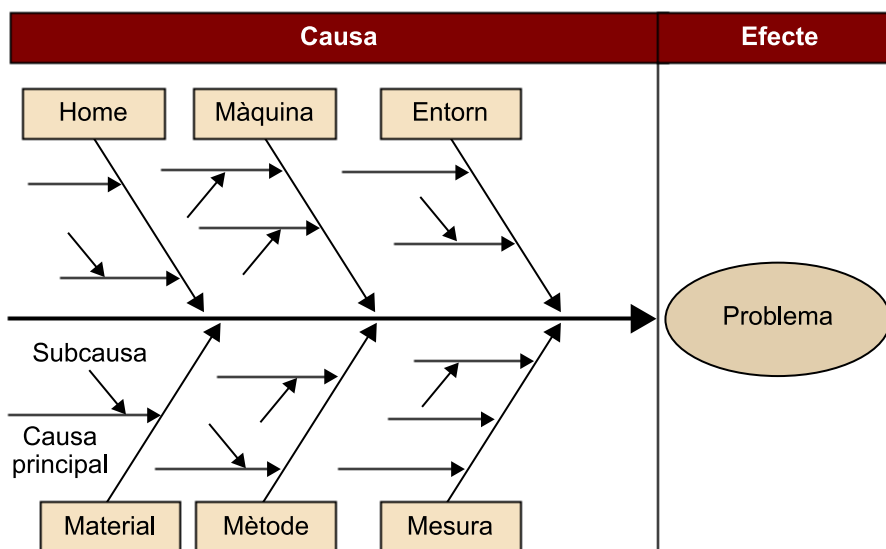
El **diagrama d'Ishikawa** o **diagrama causa-efecte** s'utilitza per a recollir de manera gràfica totes les causes possibles d'un problema o identificar els aspectes necessaris per a assolir un determinat objectiu (efecte).

### Kaoru Ishikawa

L'enginyer japonès Kaoru Ishikawa és considerat el pare de l'anàlisi científica de les causes de problemes en processos industrials, i dóna nom al diagrama d'Ishikawa, els gràfics del qual agrupen per categories totes les causes dels problemes.

L'ús d'aquest diagrama, d'un gran impacte visual, mostra les interrelacions entre un efecte i les causes possibles que té de manera ordenada, clara, concisa i d'un sol cop d'ull, i permet així comprendre millor l'efecte en estudi, fins i tot en situacions molt complexes, ja que centra l'atenció de tots els components del grup de manera estructurada i sistemàtica.

Per l'aspecte que té, també es denomina **diagrama d'espina de peix** (*fishbone*).



Font: Wikipedia Commons. Autor: VARGUX

Per a desenvolupar el diagrama causa-efecte s'han de seguir els passos següents:

1) **Definir i determinar clarament el problema o efecte que serà analitzat.** El problema és una cosa que generalment volem millorar o controlar. Aquest problema ha de ser específic i concret: incompliment de les dates d'instal·lació, quantitats inexactes en la facturació, errors tècnics, etc.

2) **Identificar els factors o les causes que originen l'efecte**, amb una pluja d'idees (*brainstorming*), i classificar-los segons la categoria o naturalesa que tenen. Per a classificar els factors o les causes trobats, sovint s'utilitza com a referència les categories de les **quatre emes** definides per Ishikawa: mà d'obra, maquinària, materials i mètodes. Aquestes categories són els rètols de les espines.

3) **Representar el diagrama**. Una vegada enumerades totes les causes les hem d'anar col·locant en el diagrama agrupant les de naturalesa similar.

4) **Analitza les relacions causa-efecte que deriven de la construcció del diagrama**. Una vegada identificades les causes hem d'identificar les "causes més probables", ja que no totes les que apareixen durant la pluja d'idees estan estretament relacionades amb el problema.

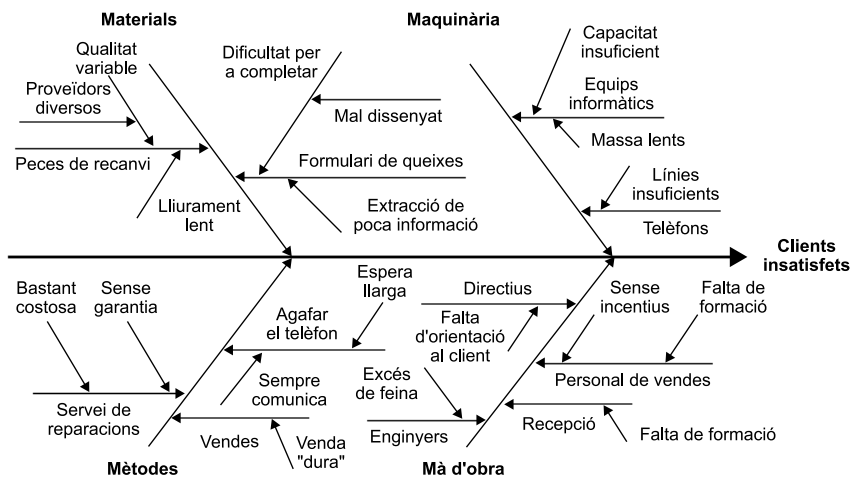
Entre altres aplicacions, es pot utilitzar per a saber i afrontar les causes dels defectes, les anomalies o les reclamacions, reduir costos, obtenir millores en els processos, millorar la qualitat dels productes, els serveis i les instal·lacions, i establir procediments normalitzats, tant operatius com de control.

Malgrat la senzillesa aparent d'aquesta eina, l'aplicació presenta una sèrie d'avantatges:

- Proporciona una metodologia racional per a resoldre problemes.
- Permet sistematitzar les causes possibles d'un problema.
- Afavoreix el treball en equip i permet que els treballadors plantegin de manera creativa les seves opinions i que la comunicació sigui clara i eficaç.

Aquesta tècnica d'identificació de les causes possibles d'un problema no aporta solucions. El diagrama ajuda a passar d'opinions a teories comprovables, però només la recopilació de dades sobre les causes més probables aporta informació sobre quina és la solució que s'ha de prendre per a resoldre l'efecte o problema.

### Exemple de diagrama d'espina de peix per a "clients insatisfets"





## 9. Gestió clàssica de projectes i noves maneres de plantejar-se la gestió de projectes

La gestió de projectes és un conjunt d'activitats específiques que es fan servir per administrar el projecte.

Aquest mòdul tracta de la gestió de projectes des d'un punt de vista clàssic, però hi ha enfocaments i plantejaments nous que intenten evitar els errors més habituals que es cometen en la gestió de projectes: les **metodologies àgils**.

### La complexitat del projecte multimèdia

Cal recordar que els **productes multimèdia**, en general, són projectes complexos i es basen en la incertesa per definició. Aplicat a l'entorn del desenvolupament de productes multimèdia, podem dir que un **projecte és complex** quan, tant la tecnologia que cal utilitzar per a resoldre un problema com els coneixements sobre els requisits del projecte, no són coneguts d'una manera clara al principi del procés.

Els requisits tendeixen a ser complexos amb molta facilitat. De fet, podem parlar de requisits simples si:

- El client els pot capturar tots i transmetre'ls al desenvolupador, i aquest és capaç d'entendre'ls completament.
- No hi ha diverses parts interessades en el projecte amb interessos divergents.
- El client sap exactament què necessita.

Com es pot veure, això difícilment és assolible. A més, hem de tenir en compte que els productes multimèdia són desenvolupats per persones que treballen en grup. Si les persones ja som complexes des del punt de vista individual, aquesta complexitat s'incrementa a bastament quan treballem en grup.

En les metodologies tradicionals, altrament dites *predictives*, s'ha intentat lluitar contra aquesta complexitat de manera que tant els requisits com la tecnologia –tots dos, aspectes inherents al desenvolupament de productes multimè-

dia– desapareguessin de l'equació, amb l'obtenció de fases molt grans de presa de requisits, llargues anàlisis d'aquests requisits, i contractes massa tancats i restrictius com a resultat.

Malgrat els esforços efectuats, és difícil lluitar contra la complexitat dels productes multimèdia i aquestes aproximacions no han comportat la realització de projectes més bons, sinó tenir projectes amb problemes.

### **Problemes en el desenvolupament de productes multimèdia**

Els factors habituals pels quals un producte multimèdia falla són els següents:

- Objectius del projecte poc realistes o poc articulats.
- Estimacions poc acurades dels recursos necessaris per a portar a terme el projecte.
- Requisits del sistema mal definits.
- Informació pobra sobre l'estat del projecte.
- Riscs no gestionats.
- Comunicació insuficient entre els usuaris, els desenvolupadors i els clients.
- Utilització de tecnologia poc madura.
- Incapacitat de gestionar la complexitat del producte.
- Pràctiques de desenvolupament descuidades.
- Gestió ineficaç del projecte.
- Pressions comercials.

En un estudi realitzat per BCS l'any 2017, podem veure que dels 214 projectes analitzats de tots els àmbits duts a terme amb metodologies predictives, el 23,8% van ser cancel·lats i el 32,2% es van acabar per sobre del pressupost o del temps previst. Això vol dir que prop d'un 55% dels projectes no van tenir el final desitjat. Altres estudis corroboren aquestes dades. Si tinguéssim en compte la satisfacció final del client amb el producte que es lliura, segurament aquests percentatges encara s'incrementarien més.

Alguns d'aquests problemes són especialment típics i preocupants:

- El client triga molt temps a poder utilitzar el resultat del projecte; mentre tant, el context canvia i els competidors treuen nous productes. Si es cancel·la el projecte, el client haurà gastat els diners per a no-res.
- El projecte s'ha complicat més del que es podia esperar: hi ha retard i s'han d'accelerar els lliuraments. Comencen els pedaços i no hi ha temps per a proves ni per al control de qualitat.
- L'equip fa hores extra i està poc motivat: simplement es dedica a complir ordres. Cadascú va a la seva.
- El client demana canvis! Aquests canvis són difícils de realitzar, ja que el projecte s'està acabant i no hi ha temps ni pressupost suficients.
- Tothom acaba insatisfet, tant el client com l'equip.

### Per què apareixen aquests problemes?

D'una banda, les diverses fases s'allarguen més del previst, i sempre en surten perjudicades les fases de proves i de possibles canvis.

De l'altra, el client només ha estat veient paper (documents de disseny) fins a una fase molt avançada del projecte. El client talpeja, gasta diners en una cosa que no té cap mena d'aplicació. Són documents, no els poden fer servir.

Tanmateix, sempre apareixen les discussions entre el client i el proveïdor. L'un considera que l'altre no ha entès què volia i que no s'estan complint els terminis previstos. L'altre pensa que el que se li demana no estava a l'abast del projecte i que hi perd diners. Tot això fa que es dediqui molt d'esforç a objectius que aporten poc valor.

És a dir, en el desenvolupament de productes multimèdia es compleix perfectament el *principi de Pareto*, segons el qual el 80% del valor d'un producte prové del 20% de les funcionalitats implementades. Per tant, invertim molt de temps, esforç i diners en coses que no es faran servir. Hem de fer-ho pagar al client, això?

Sembla clar que, per a molts, aquesta no és la millor manera de desenvolupar un producte multimèdia, per això sorgeix el desenvolupament àgil de projectes.

### El desenvolupament àgil

L'any 2001, disset enginyers d'alt nivell en el camp del desenvolupament de programari es van reunir a Utah per a explorar i compartir quin consideraven que havia de ser el futur del desenvolupament de programari. Dins del grup, hi

havia molts dels proponents de metodologies emergents en aquell moment, com Scrum, Extreme Programming, Crystal, Feature Driven Development, i d'altres. Junts, van coincidir en el nom per al seu moviment: *agile* (àgil).

En aquesta reunió, els integrants van constituir l'Agile Alliance (Aliança Àgil), i van escriure *Manifesto for agile software development* (popularment, *The agile manifesto*; en català, *El Manifest àgil*): un conjunt d'estaments que han servit com a declaració de valors i principis que han de seguir les metodologies àgils.

- Persones i interaccions per sobre de processos i eines.
- Programari que funciona per sobre de documentació exhaustiva.
- Col·laboració amb el client per sobre de negociació de contractes.
- Resposta al canvi per sobre del seguiment d'una planificació.

Els valors àgils semblen simples i obvis la primera vegada que hom els sent, i continuen sense modificar-se des del dia que els fundadors de l'Aliança Àgil els van publicar com a part d'*El Manifest àgil*.

Aquests dotze punts són els principis sobre els quals es basa el desenvolupament de programari àgil, una bona pràctica en els equips que desenvolupen per mitjà de metodologies àgils, llevat la de revisar periòdicament la llista de principis per assegurar-se que es continuen seguint.

### **Principis d'El Manifest àgil**

- 1) La nostra prioritat principal és satisfer el client mitjançant el lliurament inicial i continu de programari avaluable.
- 2) Són benvinguts els canvis en els requisits, fins i tot en etapes tardanes del desenvolupament. Els processos àgils aprofiten el canvi com un avantatge competitiu del client.
- 3) Cal alliberar programari sovint, des d'unes quantes setmanes fins a uns quants mesos, amb preferència per l'escala més curta.
- 4) La gent de negocis i els desenvolupadors han de treballar plegats diàriament al llarg del projecte.
- 5) Construïm els projectes al voltant de persones motivades. Els oferim l'entorn i el suport que necessiten, i hi confiem perquè facin la feina.
- 6) El mètode més eficient i efectiu per a compartir informació amb l'equip de desenvolupament són les converses cara a cara.
- 7) El programari que funciona és la principal mesura de progrés.
- 8) Els processos àgils promouen el desenvolupament sostenible. Els patrocinadors, els desenvolupadors i els usuaris han de ser capaços de mantenir el ritme indefinidament.
- 9) L'atenció contínua a l'excel·lència tècnica i al bon disseny milloren *agile*.
- 10) La simplicitat –l'art de maximitzar la feina no feta– és essencial.
- 11) Les arquitectures, els requisits i els dissenys més bons emergeixen d'equips auto-organitzats.
- 12) En intervals regulars, l'equip reflexiona sobre la manera de ser més efectiu i, consegüentment, afina i ajusta el context.

Podeu trobar el Manifest àgil, en la redacció original, a l'adreça d'Internet següent:  
<http://agilemanifesto.org>

### **Col·laboració amb el client per sobre de negociació contractual**

Aquest valor àgil també busca evitar el gran disseny inicial, però ho fa posant l'èmfasi a mantenir un diàleg continu entre l'equip de desenvolupament i el client, tan obert i fluid com sigui possible. Els contractes són bons i necessaris, especialment quan protegeixen els interessos de les dues parts, tal com succeeix en els bons contractes.

Però molts dels fundadors de l'Aliança Àgil eren consultors i coneixien els riscos que s'amaguen darrere dels contractes. Un dels més habituals en la nostra indústria és que el proveïdor es compromet a realitzar una certa quantitat de feina en un temps determinat, en què el profit que tregui del projecte dependrà de la rapidesa i el baix cost amb què sigui capaç de dur a terme els requisits mínims del client. Si algun cop heu volgut reformar la cuina, segurament esteu al cas de com de perjudicial per a les dues parts acaba sent aquest tipus de contracte.

Els autors d'*El Manifest àgil* van concloure que un projecte basat exclusivament en un contracte posa l'èmfasi en el lloc incorrecte. Ells van preferir promoure un entorn col·laboratiu, en què el client i l'equip treballen conjuntament per aconseguir un objectiu que és compartit, l'èxit del projecte.

### **Resposta al canvi per sobre del seguiment d'una planificació**

Les organitzacions que basen el seu model a "seguir el pla establert" acostumen a tenir processos creats amb les millors intencions. Per prevenir que el projecte s'endarrerixi, no es permeten canvis de funcionalitat no planificats o qualsevol altra disfunció que pugui afectar el pla inicial o, si de cas, se sotmeten a uns requisits durs, com revisar tota la planificació o, fins i tot, redactar un nou contracte o un annex a l'actual. Però, malgrat tot, això no funciona. Un control de canvis només té èxit en un entorn en què els canvis són controlables. En canvi, el desenvolupament de programari és un procés en què a mesura que avancem anem descobrint coses que no sabíem al començament, la qual cosa ens obligarà a fer canvis no esperats.

La planificació en els projectes de desenvolupament de productes multimèdia ha de ser fluida, no fixa, per al benefici de l'equip però sobretot per al benefici del producte que estem desenvolupant i, en última instància, per al benefici del client. Aquesta és la raó per la qual planifiquem per al canvi.

Dins l'àmbit de la gestió àgil de projectes, cal destacar dos conceptes: SCRUM i Kanban.

### **9.1. Scrum. La conceptualització del producte. El dia a dia en un projecte àgil**

Scrum és un marc de treball per a definir processos que es caracteritza perquè és lleuger i fàcil d'entendre.

Scrum no defineix un procés concret, sinó que proporciona les eines perquè cada equip personalitzi el marc de treball i trobi el procés més adequat a les seves circumstàncies.

Aquesta característica fa que Scrum sigui apropiat per a situacions complexes, en què és difícil predir el que passarà en el futur i en què, per tant, caldrà una gran capacitat d'adaptació, com en el cas del desenvolupament de productes multimèdia.

Scrum és un mètode empíric i, per tant, es basa a gestionar el procés de desenvolupament a partir de l'experiència (observació) i no pas per mitjà de prediccions. La conseqüència principal d'això és que les decisions es prenen tenint en compte fets coneguts, en comptes de fets hipotètics.

Amb Scrum establirem igualment una predicció inicial però durant l'execució del projecte ens anirem qüestionant, d'una manera regular, la correcció de la predicció, i l'ajustarem (per exemple, adaptarem el calendari o modificarem el conjunt de funcionalitats) perquè s'adeqüi a la realitat.

## Com funciona Scrum

### 1) El *product backlog*

Un projecte de Scrum és impulsat per una visió de producte elaborada pel propietari del producte, i s'expressa a la **pila de producte**. El *product backlog* és una llista prioritzada del que es requereix, per ordre de valor per al client o negoci, amb els elements de més valor a la part superior de la llista. El *product backlog* evoluciona durant la vida útil del projecte, i els elements s'afegeixen, es treuen o són reprioritzats contínuament.

L'equip Scrum es compon de tres rols de Scrum:

- **El propietari del producte.** Pren les entrades del que ha de ser el producte i ho tradueix en una visió de producte.
- **L'equip.** Desenvolupa el producte previst pel propietari del producte.
- **L'Scrum màster.** Té tot el que es necessita perquè l'equip Scrum assolixi l'èxit. Això passa per eliminar els obstacles d'organització, facilitar les reunions i actuar com un guardià perquè ningú no obstaculitzi la feina de l'equip.

### 2) L'*sprint*

Scrum estructura el desenvolupament de productes en cicles de treball anomenats *sprints*, repeticions de treball que típicament tenen d'una a quatre setmanes de durada. Els *sprints* són de durada determinada; mai no s'estenen més enllà de la data fixada al principi, independentment que la feina planificada per l'*sprint* s'hagi completat o no.

### 3) Planificació de l'*sprint*

Al començament de cada *sprint*, es porta a terme la **reunió de planificació de l'*sprint***. El propietari del producte i l'equip Scrum (amb la facilitació de l'Scrum màster) revisen la **pila de producte**, en discuteixen els objectius i el context de les històries d'usuari, i l'equip Scrum selecciona els elements de la pila de producte que es comprometen a fer per al final de l'*sprint*, partint de la part superior de la pila de producte, on hi ha els elements de prioritació més alta.

Cada element seleccionat a la pila de producte es descompon després en una sèrie de tasques individuals. La llista de tasques es registra en un document anomenat *sprint backlog*.

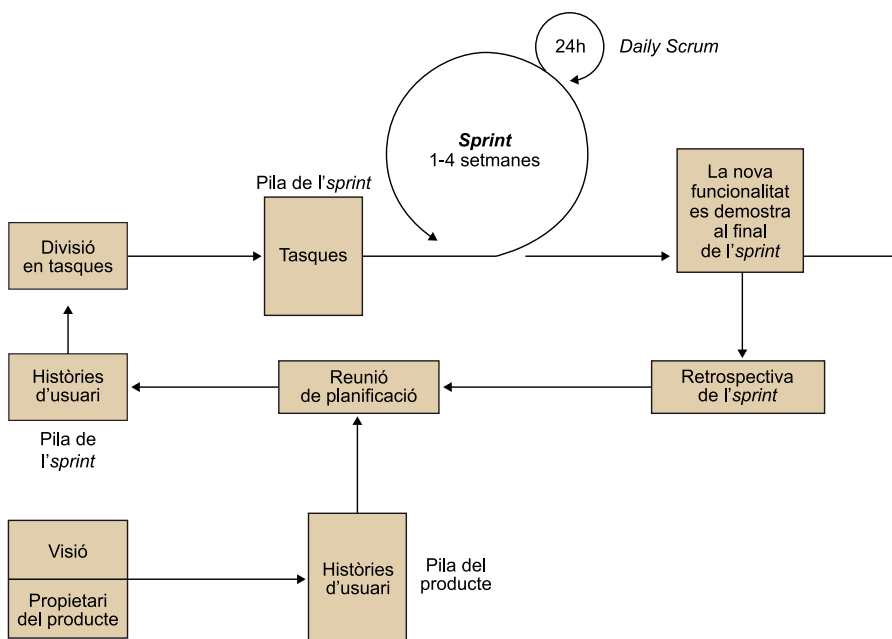
#### 4) Reunió diària de Scrum

Una vegada ha començat l'*sprint*, l'equip Scrum es dedica a una altra de les pràctiques principals de Scrum: la **reunió diària de Scrum** o *daily stand-up* en la denominació original en anglès. Es tracta d'una trobada breu (quinze minuts, com a màxim) que es duu a terme cada dia de feina a una hora determinada. Tots els membres de l'equip hi assisteixen. En aquesta reunió, es presenta la informació necessària per a inspeccionar l'avenç des del dia anterior. Aquesta informació pot donar com a resultat una replanificació i debats sobre la funcionalitat després del *daily stand-up*.

#### 5) Revisió de l'*sprint* i retrospectiva

Un cop finalitzat l'*sprint*, es duu a terme la reunió de revisió de l'*sprint*, en què l'equip Scrum i les parts interessades inspeccionen el que s'ha fet durant l'*sprint* i ho discuteixen. En aquesta reunió, hi són presents el propietari del producte, els membres de l'equip i l'*Scrum* màster, a més dels clients, gent de negocis, experts, executius i qualsevol altra persona interessada.

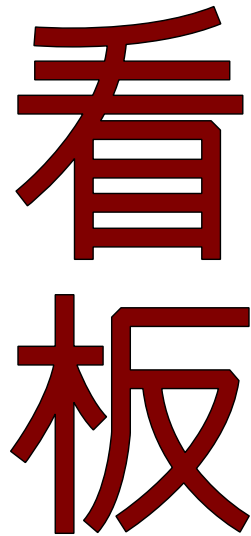
Després de la revisió de l'*sprint*, l'equip es reuneix per a la retrospectiva de l'*sprint*, una oportunitat perquè l'equip discuteixi el que ha funcionat bé durant l'*sprint* i el que no funciona, i es posin d'acord en els canvis que cal intentar fer en l'*sprint* següent per tal de ser més productius.





## 9.2. Kanban com a sistema de gestió del treball en curs

Derivat de la combinació de les dues paraules japoneses *Kan*, que significa ‘visual’, i *Ban*, que significa ‘targeta’, neix el terme *Kanban*, amb el qual s’anomena una metodologia de producció o organització del treball que es basa en senyals visuals per a gestionar l’esforç i la dedicació del nostre equip de producció. Kanban permet al gestor de l’equip de producció multimèdia identificar aglomeracions en la producció, millorar el temps de servei de tasques i optimitzar la qualitat en el procés de producció.



Kanban (en kanji 看板, on kan (看) significa “visual” i ban (板) significa “targeta”).

**Kanban** és un sistema de gestió en què es produeix exactament la quantitat de feina que el sistema és capaç d’assumir.

Kanban és un sistema de gestió del **treball en curs** (WIP<sup>3</sup>, de l’anglès, *work in progress*), que serveix principalment per assegurar una producció continuada i sense sobrecàrregues en l’equip de producció multimèdia i en el qual es produeix exactament la quantitat de feina que el sistema és capaç d’assumir. També és un sistema de treball **just a temps** (*just in time*), la qual cosa significa que evita sobrants innecessaris d’estoc, que, en la gestió de projectes multimèdia, equival a la inversió innecessària de temps i esforç en allò que no necessitem (o, simplement, és menys prioritari) i així no se sobrecarregarà l’equip.

Kanban és una aproximació a la gestió del **canvi organitzatiu**. No és un procés de desenvolupament de productes multimèdia o de gestió de projectes. També és una aproximació a la introducció de canvis en el cicle de vida de desenvolupament de productes multimèdia o metodologia de gestió de projectes ja existent. Amb Kanban es comença amb qualsevol cosa que estigui en curs en la gestió de l’equip de producció, no cal començar de zero en l’organització d’una empresa per adoptar aquest sistema.

En la gestió del treball en curs amb Kanban, es busca un concepte clau: **limitar el treball en curs**. Està demostrat que si es gestiona molt treball en curs a la vegada, els índexs de qualitat disminueixen dràsticament. En la producció de projectes multimèdia, augmentar el treball en curs implica incrementar la quantitat d'errors que tindrà aquest projecte multimèdia, a conseqüència de la poca concentració que els desenvolupadors podran dedicar a les tasques.

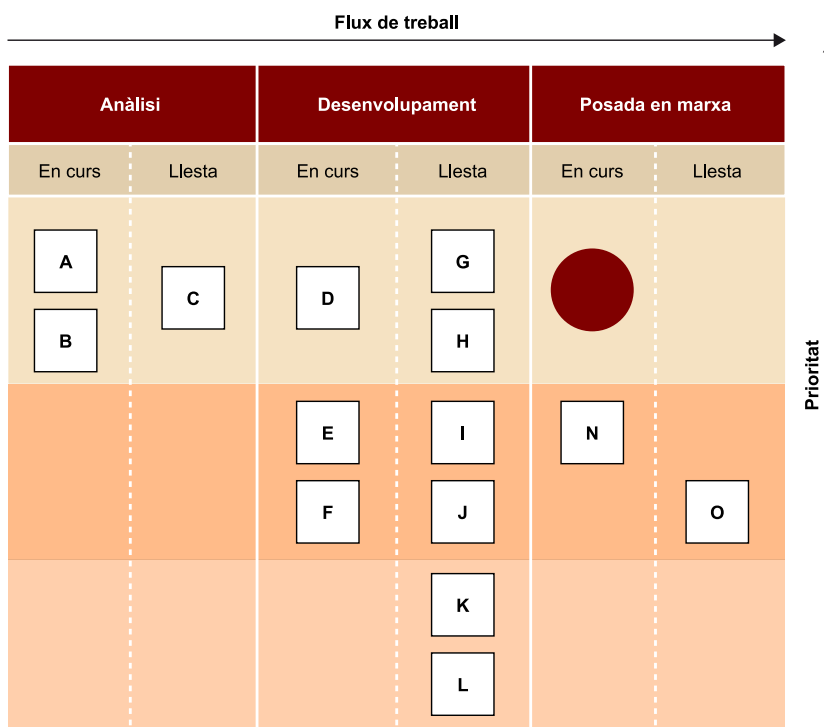
Limitar el **treball en curs** implica un augment considerable de la qualitat del programari desenvolupat en la producció d'un projecte multimèdia.

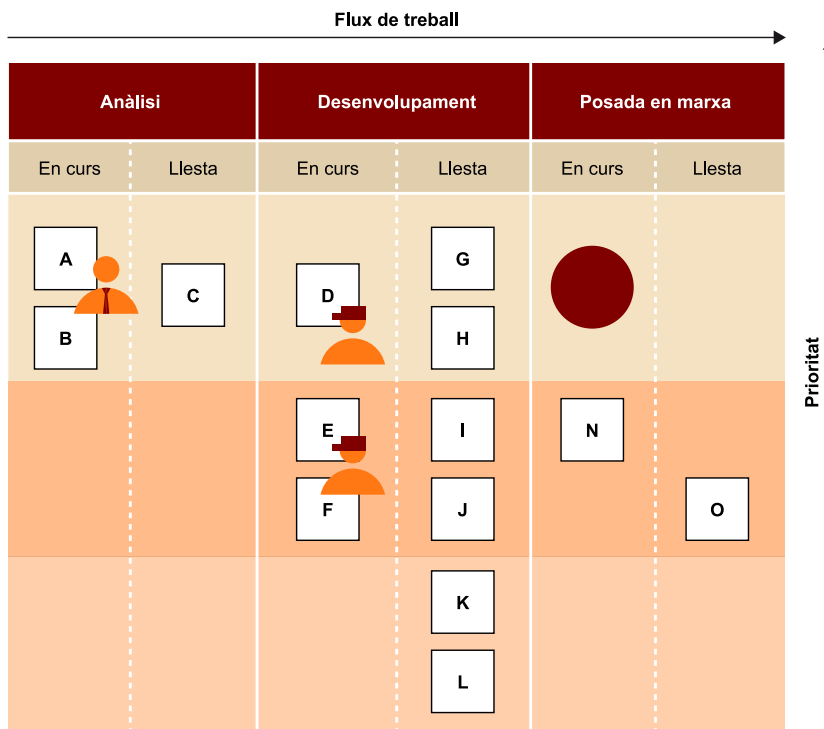
Limitar el treball en curs mitjançant la gestió del treball amb Kanban també té una altra conseqüència important, i és que disminuïm el temps de servei d'una tasca des que entra al sistema fins que en surt. Disminuint la quantitat de treball en curs, aconseguim que l'enfocament en cadascuna de les tasques sigui més alt i que el conjunt del temps dedicat a totes sigui inferior a l'utilitzat si s'assumeixen totes de cop.

Per tant, podem dir que Kanban, per mitjà d'una disciplina estricta de limitar la quantitat de feina que l'equip porta a terme alhora, ens retorna uns índexs de qualitat més alts i un temps de servei bastant més baix.

### Panell Kanban

Un panell Kanban típic s'implementaria tal com es mostra en la imatge següent:





### Representació visual

La representació visual del flux de treball mitjançant panells ha de ser fidel a la realitat i s'ha d'actualitzar constantment.

En aquesta imatge veiem un panell constituït per tres columnes que representen les diverses fases per les quals ha de fluir una tasca per a ser desenvolupada (anàlisi, desenvolupament i posada en marxa). Cada fase està subdividida en dos estats: “En curs” i “Llesta per passar a la fase següent” (aquesta divisió està representada per la línia discontinua de cada fase). L'estat “En curs” significa que “l'equip treballa actualment en aquesta tasca i en aquesta fase”, i l'estat “Llesta” significa que “l'equip ja ha acabat la feina que havia de fer en aquesta fase, i la tasca espera que el sistema pugui assumir-la per a la fase següent”. Aquesta divisió ens ajuda a localitzar embussos en el nostre procés de producció; ho aprendrem més endavant en el mòdul.

Les files podrien ser projectes diferents en els quals treballa l'empresa, però el més habitual és que les files indiquin prioritats, de manera que les tasques superiors són les prioritàries.

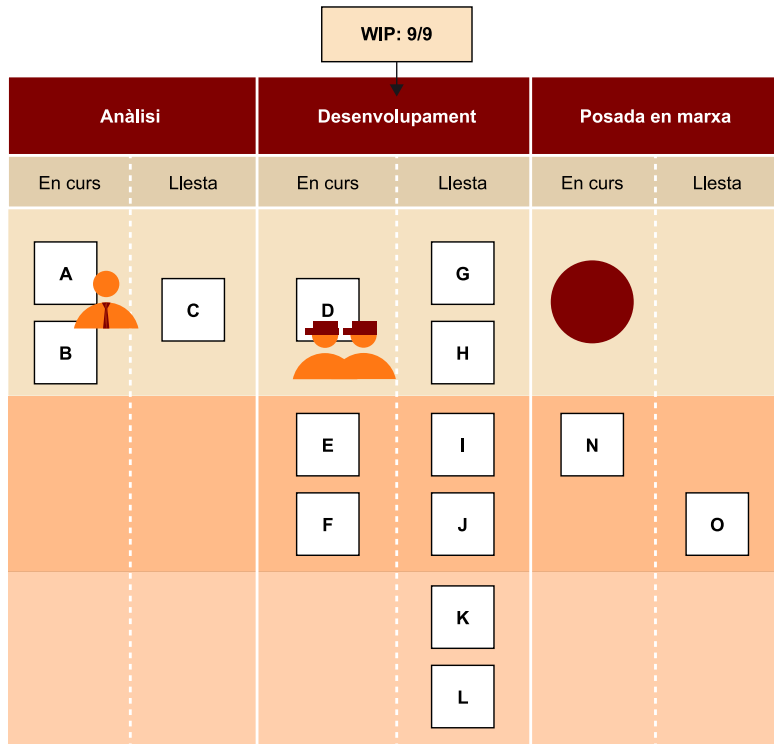
Separar en cada fase les tasques segons l'estat, “En curs” i “Llesta”, ens ajuda a localitzar embussos en el procés de producció de productes multimèdia.

### Limitar el treball en curs (WIP)

Una vegada hem representant la realitat actual del nostre equip, hem d'actuar i aplicar la primera restricció de Kanban: **limitar el WIP**. Hem de fer obligatori que els programadors deixin de programar i es dediquin a acabar d'enllestir les tasques que estan bloquejades. Cal fer que els treballadors del nostre equip només produeixin si el sistema accepta més quantitat d'aquesta feina produïda.

## Els colls d'ampolla

Una vegada que la fase “Desenvolupament” ha arribat al seu límit WIP, tenim un coll d'ampolla: la fase “Posada en marxa” provoca un embús.



En aquest moment, es poden prendre tres decisions diferents:

1) **La pitjor.** Augmentar o treure el límit WIP. Coneixem la teoria i sabem que Kanban limita el WIP, però com que **no ho podem fer**, i és **impossible**, decidim treure'l i posem un límit WIP de deu, i més endavant el posarem a quinze. Realment, no ajuda en res l'empresa ni l'equip.

2) **La dolenta.** Contractar gent per a la fase en el coll d'ampolla. La solució en un equip sobrecarregat no sempre és posar-hi més gent. S'ha de tenir en compte que incorporar gent nova requereix dedicar-hi temps, hi ha un període en què s'ha de destinar temps a formar la gent nova (mai ningú no entra sabent-ho tot, per molt experta que sigui la persona) i aquest temps es traurà de la gent que està justament en la fase de la qual depèn el temps de servei del vostre equip (en l'exemple de l'apartat “Aplicació pràctica de Kanban”, l'expert en servidors).

3) **La bona.** Posar un dels programador a ajudar en la fase “Posada en marxa”. Un bon equip d'alt rendiment és el que està format per persones expertes en alguna tasca, però que coneix i s'interessa per totes les que l'envolten. Això s'anomena **treballador T**.

En el moment en què al programador li dius “deixa de programar i posa’t a verificar”, una resposta molt habitual és “la meva feina és programar, a mi em paguen per programar i això és el que faré”. Cal tenir molt clar que l’objectiu de tot l’equip ha de ser **el servei al client** (és a dir, **facturar**). Un treballador que no està disposat a assimilar que la feina de programar no és la que el sistema de producció necessita en aquest moment (en què el panell Kanban reflecteix que tenim nou tasques programades i només una de facturada) no està fent cap favor a l’empresa.

### **L’èxit de Kanban**

Implementar Kanban correctament comporta un alt grau d’implicació en l’equip i fer entendre als treballadors que la seva feina no ha de ser “programar” o “instal·lar aplicacions”, sinó fer que l’empresa obtingui beneficis.

Aquesta cultura tan oriental, que també afecta la manera d’entendre els contractes entre client i proveïdor, és el que porta les empreses que implementen Kanban a l’èxit.



## Bibliografia

**Anderson, David J.; Reinertsen, Donald G.** (2010). *Kanban. Successful evolutionary change for your technology business* [en línia]. <<http://www.amazon.com/Kanban-Successful-Evolutionary-Technology-Business/dp/0984521402>>

**Bataller, Alfons** (2010). *La gestió de projectes*. Barcelona: Editorial UOC.

**Durán Rubio, S. E.** (2003). "Puntos por función. Una métrica estándar para establecer el tamaño del software". A: *Boletín de Política Informática* (núm. 6). Mèxic: Ed. Instituto Nacional de Estadística y Geografía de México.

**Krug, Steve** (2006). *No me hagas pensar: una aproximación a la usabilidad en la web*. Madrid: Pearson Educación.

**Ministerio de Administraciones Públicas.** *Metodología MÉTRICA Versión 3*.

**Nielsen, Jakob** (2000). *Usabilidad: diseño de sitios web*. Madrid: Prentice Hall.

**Nielsen, Jakob** (2006). *Usabilidad: [prioridad en el diseño web = (prioritizing web usability)]*. Madrid: Anaya Multimedia.

**Nielsen, Jakob** (2010). *Técnicas de eyetracking para usabilidad web*. Madrid: Anaya Multimedia.

**Pereña Brand, J.** (1996). *Dirección y gestión de proyectos* (2a. ed.). Madrid: Ediciones Díaz de Santos, SA.

*Usability Guidelines* [document en línia]. Cambridge, MA.: Information Services and Technology. <<http://ist.mit.edu/services/consulting/usability/guidelines>>

