



Metodologia SCRUM: anàlisi i aplicació pràctica per a equips distribuïts

Judith Tierno Barroso

Enginyeria tècnica d'informàtica de gestió

Xavier Martínez Munné

Gener 2016



Aquesta obra està subjecta a una llicència de [Reconeixement-NoComercial-SenseObraDerivada 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FITXA DEL TREBALL FINAL

Títol del treball:	<i>Metodologia SCRUM: anàlisi i aplicació pràctica per a equips distribuïts</i>
Nom de l'autor:	<i>Judith Tierno Barroso</i>
Nom del consultor:	<i>Xavier Martínez Munné</i>
Data de lliurament (mm/aaaa):	<i>01/2016</i>
Àrea del Treball Final:	<i>Gestió de projectes</i>
Titulació:	<i>Enginyeria tècnica d'informàtica de gestió</i>
Resum del Treball (màxim 250 paraules):	
<p>Aquest treball pretén recollir un anàlisi del funcionament de les metodologies àgils en la gestió de projectes de desenvolupament de software.</p> <p>Aprofundirem en la metodologia Scrum, una de les més esteses a nivell mundial per al desenvolupament de projectes àgils. Veurem com es gestiona un projecte fent servir aquesta metodologia i analitzarem què ens aporta respecte de les metodologies tradicionals de gestió de projectes.</p> <p>Per tal de veure-ho a la pràctica, analitzarem un projecte real plantejat amb la metodologia de Scrum. Definirem uns requisits per a implementar un portal corporatiu amb una casuística determinada i tenint en compte que l'equip de treball està distribuït a diferents localitzacions geogràfiques.</p>	
Abstract (in English, 250 words or less):	
<p>This work aims to collect an analysis of how agile methodologies works applied to management of software development projects.</p> <p>Delving into the Scrum methodology, one of the most widespread method for Agile development projects. We will see how to manage a project using this methodology and analyze what gives us respect for traditional methods of project management.</p> <p>To see it in practice, we'll analyze a real project planned using the Scrum methodology. We'll define requirements for implementing a corporate portal with a specific casuistry and considering that the team is distributed in different geographical locations.</p>	

Paraules clau (entre 4 i 8):

Metodologies àgils, Scrum, gestió de projectes de desenvolupament de software

Índex

1. Introducció	1
1.1 Context i justificació del Treball	1
1.2 Objectius del Treball	2
1.3 Enfocament i mètode seguit	2
1.4 Planificació del Treball	2
1.5 Breu sumari de productes obtinguts	4
1.6 Breu descripció dels altres capítols de la memòria	4
2. Bases de gestió d'un projecte	6
Cicle de vida d'un projecte de gestió de software	6
Metodologia en cascada	6
Factors crítics	8
Metodologies àgils	9
¿Què són son les metodologies de desenvolupament àgil?	9
Manifest àgil	9
Principis àgils	15
Principals models àgils	16
Comparativa entre enfoc tradicional i metodologies àgils	16
3. SCRUM	19
Components de la metodologia	20
Elements	21
Reunions	22
Rols	23
Requisits i abast del producte	24
Backlog	25
Histories d'usuari	27
Sprint	27
Planificació	29
Planificació de poker	29
Seguiment, control i mesuraments	31
Resum del procés d'Scrum	32
4. Cas pràctic: desenvolupament d'un portal corporatiu amb Scrum	33
Requeriments del projecte	33
Requeriments funcionals	33
Planificació del projecte	36
Seguiment	40
Anàlisi	42
5. Conclusions	43
6. Glossari	46
7. Bibliografia	48
8. Annexos	49

Llista de figures

Il·lustració 1 - Diagrama del model en cascada	6
Il·lustració 2 - Factors crítics per al desenvolupament d'un projecte seguint el model en cascada	8
Il·lustració 3 - Manifest àgil	10
Il·lustració 4 - Model gràfic del desenvolupament en cascada. Font: Northware software.	16
Il·lustració 5 - Model gràfic del desenvolupament amb Scrum. Font: Northware software.	17
Il·lustració 6 - Principals diferències entre metodologies àgils i tradicionals	18
Il·lustració 7 - Pilars de la metodologia Scrum	19
Il·lustració 8 - Framework Scrum. Font: xsfera.com	20
Il·lustració 9 - Taula resum dels elements d'Scrum	21
Il·lustració 10 - Exemple de Burn down Chart – font: wikipedia	22
Il·lustració 11 - Representació del product backlog	26
Il·lustració 12 - Exemple de product backlog. Font: Mountain goat software	26
Il·lustració 13 - Format d'una història d'usuari	27
Il·lustració 14 - Exemple d'Sprint backlog	28
Il·lustració 15 - Exemple de Task Board. Font: eclipse.org	29
Il·lustració 16 - Planificació de poker	30
Il·lustració 17 - Sprint burn down chart. Font: Wikipedia	31
Il·lustració 18 - Product burn down chart. Font: Scrum institute	32
Il·lustració 19 - Procés Scrum resumit	32
Il·lustració 20 - Mostra de wireframe del projecte	35
Il·lustració 21 - Burndown chart inicial del projecte	39
Il·lustració 22 - Organització del seguiment diari de l'equip	41

1. Introducció

1.1 Context i justificació del Treball

Des d'un temps cap aquí ha proliferat l'ús de les metodologies àgils en la gestió de projectes per a desenvolupament de software a mida. L'objectiu d'aquestes és agilitzar els processos empresarials i involucrar als equips de treball.

El que moltes vegades no es té en compte quan es comencen a fer servir, és que les metodologies com SCRUM ens donen un marc de treball, però aquest, no sempre és aplicable al 100% a la realitat del projecte a on ens trobem.

Això no vol dir que no puguem ser "àgils", sinó que ens haurem d'adaptar per a que el procés tingui sentit dintre de la nostra empresa i del projecte que tenim entre mans.

Aquest error, el no adaptar la metodologia a les circumstàncies particulars de cada cas, juntament amb una aplicació "segons conveniència", és comés diàriament a moltes empreses.

Com a conseqüència d'una aplicació errònia, les empreses abandonen l'ús de metodologies àgils amb el pretext de que no funcionen.

Aquest treball pretén recollir les pautes del marc de treball que ens donen les metodologies àgils per a projectes de desenvolupament de software i analitzar la millor aplicació per a una de les casuístiques que cada vegada es dona amb més freqüència a les empreses: els membres de l'equip estan a diferent ubicació geogràfica.

Per a il·lustrar la problemàtica i veure les millores que poden introduir-se al procés, es farà servir un cas pràctic: el desenvolupament d'un portal corporatiu amb equip de treball distribuït.

1.2 Objectius del Treball

Els objectius del treball són dos ben concrets:

- Analitzar Scrum com a metodologia àgil aplicada a la gestió de projectes
- Anàlisi d'un cas pràctic amb Scrum: Desenvolupament d'un portal corporatiu amb un equip distribuït
 - Les bases aplicades al projecte
 - Pros
 - Contres
 - Milliores aplicables al procés

1.3 Enfocament i mètode seguit

Per a enfocar el projecte es farà un anàlisi de la metodologia Scrum en detall. Entre les accions previstes, es plantegen les següents:

- Recerca d'informació en diversos medis:
 - Bibliografia especialitzada
 - Webs especialitzades en el tema
 - Assistència a events de la comunitat local
 - Consulta amb experts del sector

De cara a l'anàlisi del cas pràctic, es plantejarà un enunciat d'un supòsit que podria ser un projecte real. Es tractaran les necessitats del client per una banda i per l'altra la gestió interna de l'equip encarregat de desenvolupar el projecte, fent servir Scrum.

La metodologia Scrum serà la que aplicarem per a la gestió del propi projecte, dividint el projecte en 5 sprints, després d'haver detallat el product backlog, es a dir, l'abast del projecte. Aquests termes són explicats en el punt que recull el detall de la metodologia en aquets document.

S'adjunta també una planificació bàsica dels diferents sprints.

1.4 Planificació del Treball

El treball es divideix en 4 entregues, que s'han dividit en 5 blocs o sprints d'acord amb la metodologia Scrum. Per a cada sprint es detallen les tasques i el pes estimat per a cada tasca.

Sprint 1 - Data inici: 21/09/2015 - Duració: 15 dies

Id	Tasca	Pes estimat
1.1	Definir l'abast del projecte	8
1.2	Recerca d'informació	20
1.3	Introducció	2
1.4	Context i justificació del treball	2
1.5	Objectius del treball	2
1.6	Breu sumari dels productes obtinguts	1
1.7	Generar índex del document	5
1.8	Descripció dels capítols de la memòria	13

Sprint 2 - Data inici: 15/10/2015 - Duració: 25 dies

Id	Tasca	Pes estimat
2.1	Revisió d'entrega anterior i correccions	8
2.2	Cicle de vida d'un projecte	2
2.3	Factors crítics	2
2.4	Metodologia en cascada	2
2.5	Metodologies àgils	13
2.6	Comparativa entre enfoc tradicional i metodologies àgils	5

Sprint 3 - Data inici: 19/11/2015 - Duració: 15 dies

Id	Tasca	Pes estimat
3.1	Revisió d'entrega anterior i correccions	8
3.2	Scrum - introducció	1
3.3	Components de la metodologia	5
3.4	Requisits i abast del projecte	5
3.5	Backlog	5
3.6	Sprint	5
3.7	Planificació	2
3.8	Seguiment	2
3.9	Tancament i retrospectiva	2
3.10	Control i mesuraments	2

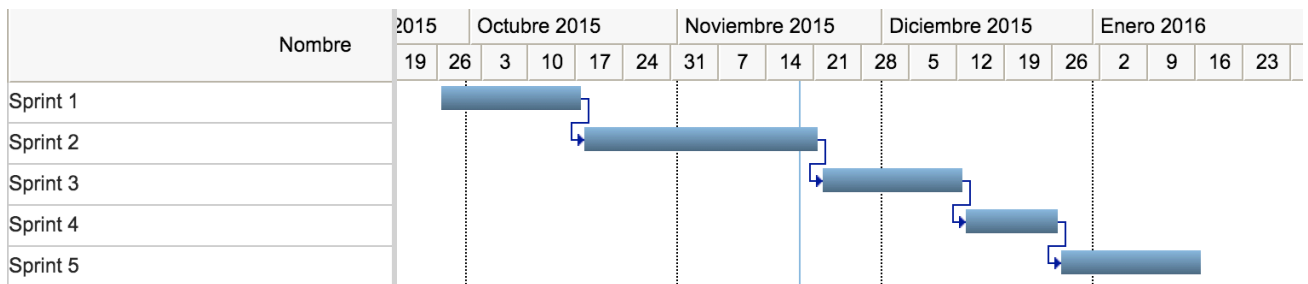
Sprint 4 - Data inici: 10/12/2015 - Duració: 10 dies

Id	Tasca	Pes estimat
4.2	Plantejament del cas pràctic	3
4.3	Requeriments	5
4.4	Particularitats	2
4.5	Abast	2
4.6	Planificació	3
4.7	Seguiment	3
4.8	Anàlisi	3

Sprint 5 - Data inici: 24/12/2015 - Duració: 15 dies

Id	Tasca	Pes estimat
5.1	Revisió d'entrega anterior i correccions	8
5.2	Conclusions del treball	8
5.3	Glossari	3
5.4	Bibliografia	1
5.5	Annexos	1
5.6	Resum executiu	5

Planificació global



1.5 Breu sumari de productes obtinguts

Del projecte es generaran dos documents i el vídeo de presentació. Els documents són:

- Memòria del projecte
- Resum executiu

1.6 Breu descripció dels altres capítols de la memòria

Bases de gestió d'un projecte

En aquest primer bloc del projecte analitzarem el cicle de vida de la construcció de software amb els seus factors crítics.

Passarem per les metodologies tradicionals, com és la metodologia en cascada per després introduir les metodologies àgils i veure les principals diferències entre unes i altres.

SCRUM

En aquesta part analitzarem en detall els components de la metodologia Scrum i la seva aplicació.

Veurem com es gestiona l'abast, la planificació i com es realitza el seguiment del projecte.

Cas pràctic: desenvolupament d'un portal corporatiu amb Scrum

Detallarem un cas d'un projecte a desenvolupar tenint equips de treball distribuïts.

Analitzarem com es gestiona de manera pràctica la planificació i seguiment del projecte aplicant la metodologia Scrum.

2. Bases de gestió d'un projecte

Cicle de vida d'un projecte de gestió de software

Quan ens plantegem un nou projecte de software, sabem que existeixen un passos bàsics que ens marquen el procés a seguir.

Aquest procés ens ajuda a donar un ordre lògic al projecte i a prioritzar en cada moment el que és més important. Els passos bàsics que considerem són:

- Presa de requeriments
- Anàlisi
- Implementació
- Proves i Validació
- Posta en marxa

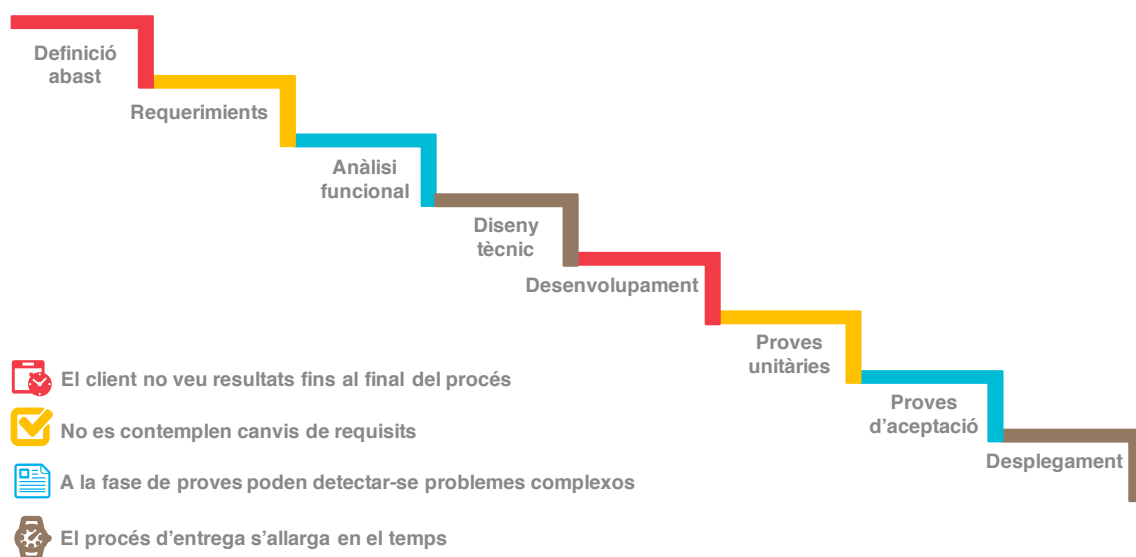
L'objectiu de seguir un procés es aconseguir resultats predictibles, que millorin la qualitat i la productivitat del projecte.

A continuació analitzem el model en cascada, metodologia que s'ha fet servir tradicionalment en el desenvolupament de projectes de software.

Metodologia en cascada

El model en cascada és el marc de treball que està més estès actualment com a metodologia de gestió de projectes IT.

Contempla les següents fases:



II-lustració 1 - Diagrama del model en cascada

- **Definició de l'abast del projecte:** Tancar els paràmetres principals del projecte, definir què queda inclòs i què no. Aquesta fase es dona quan el projecte és només una idea que es vol posar en marxa, però no s'ha aprofundit en detall. Definir l'abast ens permetrà fer una estimació inicial d'esforços i costos.
- **Preses de requeriments:** Quan es comença a aprofundir en la idea i es planteja el còm portar-la a la realitat, sorgeixen moltes qüestions que es plantejaran en aquesta fase. Es detallaran els requeriments necessaris per a portar a terme el projecte.
- **Anàlisi funcional:** Definició de la funcionalitat de manera detallada que tindrà el producte a desenvolupar.
- **Disseny tècnic:** Definició tècnica detallada de les funcionalitats a desenvolupar necessàries per a portar a terme el projecte.
- **Desenvolupament:** Fase de desenvolupament dels requisits definits a les fases anteriors.
- **Proves unitàries:** Proves fetes pel mateix equip de treball abans d'entregar el producte desenvolupat, per tal d'assegurar la qualitat de l'entregable. D'aquesta fase sortiran correccions que faran que sigui necessari iterar entre la fase anterior de desenvolupament i l'actual de proves.
- **Proves d'acceptació:** Proves fetes per part del client amb l'objectiu de validar el producte entregat. De la mateixa manera que a les proves unitàries, poden sorgir correccions que facin que sigui necessari modificar certs punts del codi i tornar a la fase d'implementació per a revisar-los.
- **Desplegament i posta en marxa:** Una vegada el projecte es dona per validat, es desplega als entorns productius i es posa en marxa.

Aquesta metodologia però, té els seus pros i contres. Ens dona un marc de treball lògic per a desenvolupar un projecte, però podem trobar-nos inconvenients a la fase d'entrega, cosa que pot fer que el projecte fracassi.

Les raons principals solen ser aquestes:

- El client no veu cap resultat fins al final del procés, cosa que pot provocar l'efecte sorpresa en veure un resultat diferent al que tenia al cap o no coincidir amb les seves expectatives.

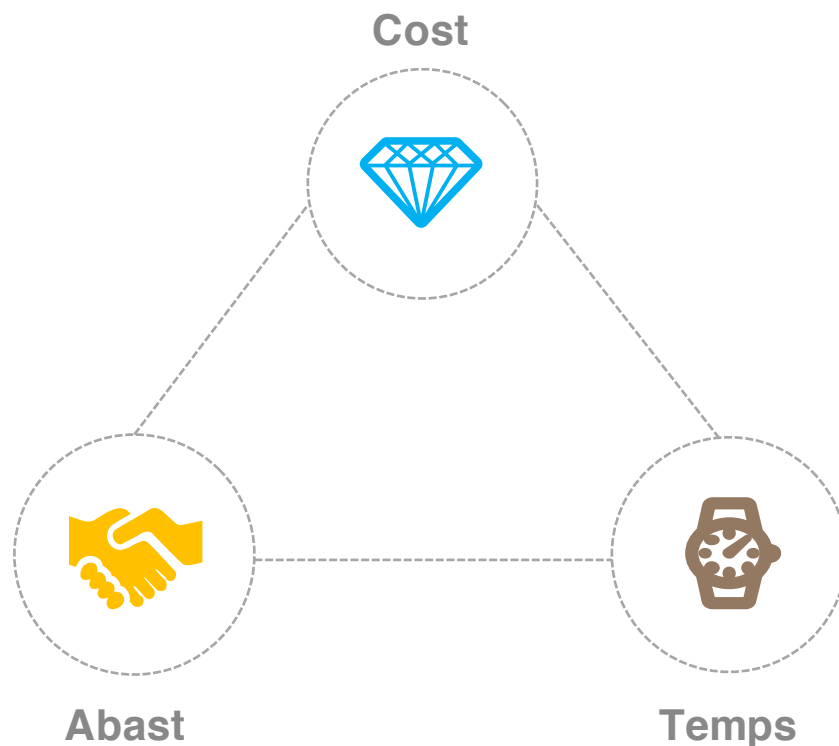
- No es contemplen canvis de requisits durant el procés, cosa que fa que el projecte no sigui flexible i no s'adapti a les necessitats canviants de la majoria del clients.
- A la fase de proves poden detectar-se problemes complexos que facin que sigui necessari replantejar punts crítics del projecte.
- El procés d'entrega s'allarga en el temps ja que fins que el producte no està finalitzat, no s'entrega.

Factors crítics

Els projectes IT seguint el model en cascada parteixen de la base de que s'ha tancat prèviament l'abast del projecte, el temps de duració i el cost total.

En el moment en que sorgeix alguna variació en qualsevol d'aquests tres factors, l'èxit del projecte pot estar en joc.

És habitual treballar una fase prèvia de definició de la oferta a on detallen aquest tres punts per tal de quan comenci el projecte estigui clar què s'ha de fer, quin cost té i quant de temps es trigarà.



Il·lustració 2 - Factors crítics per al desenvolupament d'un projecte seguint el model en cascada

Amb aquests factors definits, podem acotar altres factors necessaris per a portar a terme el projecte, com són l'equip necessari o els riscos.

Això ens dona una perspectiva de treball definida, però ¿què passa quan sorgeixen canvis en algun d'aquests factors?

- Tenint un abast tancat és difícil gestionar els canvis del client, ja que comporta negociar les condicions contractuals de temps i cost.
- Durant el procés de desenvolupament el client no té visibilitat sobre el resultat final. Això pot provocar canvis d'última hora no contemplats al projecte i posar en risc la data d'entrega i el cost.

Com a conclusió podem dir que qualsevol variació sobre aquests factors pot posar en risc l'èxit del projecte.

Metodologies àgils

¿Què són son les metodologies de desenvolupament àgil?

Les metodologies de desenvolupament àgil, aporten un nou enfoc al desenvolupament, plantejant un procés iteratiu de desenvolupament, en comptes de lineal, com acostumen a ser les metodologies tradicionals.

Agafarem la definició de la wikipedia per a ampliar la resposta a la pregunta:

El desenvolupament àgil de software envolta un nou enfoc radical per a la presa de decisions als projectes de software, refereix a mètodes d'enginyeria del software basats en el desenvolupament iteratiu e incremental, a on els requisits i solucions evolucionen amb el temps segons la necessitat del projecte, així el treball es realitzat mitjançant la col·laboració d'equips auto-organitzats i multidisciplinaris, immersos en un procés de presa de decisions a curt termini compartit.

Manifest àgil

A la web <http://www.agilemanifesto.org/> podem trobar el manifest del desenvolupament de programari àgil, juntament amb els 12 principis àgils exposats en el següent punt.

Aquest manifest, reflecteix la filosofia de les metodologies de desenvolupament àgil. És la seva essència.

Estem posant al descobert millors maneres de desenvolupar programari fent-ho i ajudant a altres a fer-ho.

Mitjançant aquesta feina hem acabat valorant:

Individus 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 de cenyir-se a una planificació

És a dir, encara que els elements de la dreta tenen valor, nosaltres valorem més els de l'esquerra.

El manifest àgil

Prioritzem...

Per sobre de...

- | | | |
|---|---|---------------------------------------|
| ✓ <i>Individus i interaccions</i> | → | • <i>Processos i eines</i> |
| ✓ <i>Programari que funciona</i> | → | • <i>Documentació exhaustiva</i> |
| ✓ <i>Col·laboració amb el client</i> | → | • <i>Negociació de contractes</i> |
| ✓ <i>Resposta al canvi</i> | → | • <i>Cenyir-se a una planificació</i> |

Il·lustració 3 - Manifest àgil

Jeff Sutherland, qui va inventar Scrum al 1993, va treballar amb Ken Schwaber per a formalitzar Scrum. Junts van estendre i millorar Scrum a moles empreses de software i varen ajudar a escriure el manifest Agile.

Aquest article, pel seu interès, s'ha extret de la web Microsoft Developer Network. L'article es titula "*Principios y valores de Agile, por Jeff Sutherland*".

Principis i valors d' Agile, per Jeff Sutherland

Aquests valors no són quelcom que els creadors del manifest àgil pensessin encomiar i a continuació oblidar-se'n. Son valors que funcionen. Cada metodologia àgil individual enfoca aquests valors d'una manera lleugerament diferent, però totes aquestes metodologies tenen processos i exercicis concrets que fomenten un o més d'aquests valors.

Individus e interaccions

Els individus e interaccions són essencials pels equips d'alt rendiment. Els estudis de "saturació de la comunicació" durant un projecte varen

mostrar que, quan no existeix cap problema de comunicació, els equips poden redir 50 vegades més que la mitjana del sector.

Per tal de facilitar la comunicació, les metodologies àgils confien en cicles freqüents d'inspecció-i-adaptació. Aquests cicles poden variar de cada pocs minuts amb programació entre iguals, cada poques hores amb integració continua o tots els dies amb una reunió de posta en marxa, fins cada iteració amb una revisió i retrospectiva.

No obstant, simplement augmentar la freqüència dels comentaris i la comunicació no és suficient per a eliminar els problemes de comunicació. Aquests cicles d'inspecció i adaptació només funcionen bé quan els membres de l'equip presenten diversos comportaments clau:

- Respecte pel valor de cada persona
- veracitat en cada comunicació
- transparència de totes les dades, accions i decisions
- confiança en que cada persona recolzarà a l'equip
- compromís amb l'equip i els objectius de l'equip

Per a fomentar aquests tipus de comportament, l'administració àgil ha de proporcionar un entorn de suport, els entrenaments de l'equip han de facilitar la seva inclusió i els membres de l'equip han de mostrar-los. Només llavors podran aconseguir els equips tot el seu potencial.

Per a que els equips arribin a aquests tipus de comportament és més fàcil del que pot semblar. La majoria dels equips eviten la veritat, la transparència i la confiança degut a les normes culturals o les experiències negatives passades de conflictes generats per comunicadors sincers.

Per a combatre aquestes tendències, la direcció i els membres de l'equip han de facilitar el conflicte positiu. Quan els equips s'enganxen en conflicte positiu, no només fomenten un comportament més productiu, sinó que també funcionen per a tenir accés a altres avantatges:

- La millora del procés depèn de que l'equip generi una llista d'impediments o problemes a la organització, s'enfronti directament a ells i a continuació, els elimini sistemàticament en ordre de prioritat.
- La innovació només es produeix amb l'intercanvi lliure d'idees conflictives, un fenomen estudiat i documentat per Hirotaka Takeuchi i Ikujiro Nonaka, els padrins de Scrum.
- La resolució d'agendes conflictives apareix quan els equips s'alineen al voltant d'objectius comuns i aflora els aspectes i conflictes potencials.
- El compromís de treballar junts només s'aconsegueix quan les persones acorden els objectius comuns i a continuació, s'esforcen per millorar personalment i com a equip.

Aquest últim punt sobre el compromís és especialment important. Només quan els individus i els equips es comprometen, se senten responsables d'entregar un alt valor, que és la línia base per als equips de desenvolupament de software. Les metodologies àgils faciliten el compromís i la mateixa organització, que és la força que controla per a aconseguir resultats en un equip àgil.

Per a crear equips d'alt rendiment, les metodologies àgils valoren als individus i les interaccions per sobre dels processos i les eines. Parlant pràcticament, totes les metodologies àgils busquen augmentar la comunicació i col·laboració a través dels cicles d'inspecció i adaptació freqüents.

Però, aquests cicles només funcionen quan els coordinadors àgils fomenten el conflicte positiu que es necessita per a compilar una base sòlida de veritat, transparència, confiança, respecte i compromís en els seus equips àgils.

Software que funciona per sobre de la documentació completa

El software que funciona és una de las grans diferències que aporta el desenvolupament àgil. Totes las metodologies àgils que es representen en el Manifest àgil (Agile Manifesto) subratllen l'entrega al client de parts petites de software que funciona a intervals fixos.

Tots els equips àgils han d'establir el que significa que diguin "software que funciona", conegut amb freqüència com la definició de terminat. En un nivell alt, una part de la funcionalitat s'ha completat només quan les seves característiques passen totes les proves i poden ser utilitzades per un usuari final.

Com a mínim, els equips han d'anar més enllà del nivell de prova unitària i provar en el nivell de sistema. Els millors equips també inclouen proves d'integració, rendiment i acceptació del client, en la seva definició del que significa haver terminat una part de la funcionalitat.

Una companyia de CMMI nivell 5, que ja té una de las tasses de defectes més baixes del món, ha mostrat amb dades extenses en molts projectes, els avantatges de procediments àgils. Específicament, es sistematitza la velocitat, doble eficàcia de producció i es redueixen defectes en un 40 per cent amb les següents accions:

- Defineix les proves d'acceptació al definir la característica
- Implementa les característiques consecutivament i en ordre de prioritat
- Executa les proves d'acceptació en cada característica tan aviat com s'implementen
- Corregeix els error que s'identifiquen com a prioritat màxima el més ràpidament possible

El manifest àgil recomana que els equips entreguin el software que funciona a intervals establerts. Capaces d'arribar a un acord respecte al que significa correcte, és una de les maneres pràctiques que els equips àgils donen lloc a l'alt rendiment i l'alta qualitat necessària per aconseguir aquest objectiu.

Col·laboració del client per sobre de la negociació del contracte

Durant les dos darreres dècades, la taxa d'èxit dels projectes ha augmentat més del doble a tot el món. Aquestes millores es van produir com a resultat dels projectes menors i de les entregues freqüents, que permetia als clients proporcionar comentaris sobre el software que funciona parcialment a intervals regulars. Els redactors del manifest sabien de què estaven parlant quan varen emfatitzar que aconseguir la implicació del client en el procés de desenvolupament de software és essencial per a l'èxit.

Les metodologies àgils fomenten aquest valor, ja que un representant del client treballa ma a ma amb l'equip de desenvolupament. Per exemple, el primer equip de Scrum tenia milers de clients. Donat que no era possible implicar-los a tots en el desenvolupament del producte, varen seleccionar un Proxy de client, anomenat propietari del producte.

El propietari del producte representava no només a tots els clients en el camp, sinó també l'administració, vendes, suport, serveis de client i a altres interessats. El propietari del producte era responsable d'actualitzar la llista de requisits cada quatre setmanes a mida que l'equip lliurava el software que funciona, tenint en compte la realitat actual i els comentaris reals dels clients i parts interessades. Els clients van ajudar activament a donar forma al software que s'estava creant.

El primer equip d'XP va començar amb un projecte de TI intern. L'equip d'XP podria tenir un usuari final de la companyia al seu equip amb ells. Aproximadament un 10 per cent del temps de consultors i equips interns poden trobar un usuari final que treballi en equip tots els dies. Per a l'altre 90 per cent del temps, han d'assignar un representat.

Aquest representat del client, a qui els equips d'XP anomenen "el client", treballa amb els usuaris finals per a proporcionar una llista clara, classificada per ordre de prioritat de les característiques que els desenvolupadors de software han d'implementar.

Es la col·laboració diària del client (o Proxy del client) que, segons mostren les dades sector, fa que els projectes àgils tinguin més de doble de taxa d'èxit que els projectes tradicionals de mitjana a tot el món.

Donat que les metodologies àgils reconeixen el valor de comprometre el client, han creat als seus equips de desenvolupament un lloc que és específicament per al representant del client.

Respondre al canvi per sobre de seguir un pla

Per tal del que els equips creïn productes que satisfan als clients i proporcionin valor, els equips han de respondre al canvi. Les dades del sector mostren que més del 60 per cent dels requisits de producte o de projecte canvien durant el desenvolupament de software.

Fins i tot quan els projectes tradicionals finalitzen a temps, segons el pressupost i amb totes les característiques del pla, els clients se senten sovint insatisfets perquè el que troben no és exactament el que desitjaven.

La llei Himpfrey diu que els clients mai saben el que desitgen fins que no veuen el software que funciona. Si els clients no veuen el software que funciona fins al final del projecte, és massa tard per a incorporar els seus comentaris.

Les metodologies àgils busquen els comentaris del client al projecte per tal de que els equips puguin incorporar el comentaris i nova informació a mida que es desenvolupa el producte.

Totes les metodologies àgils tenen processos integrats per a canviar els seus plans a intervals regulars en funció dels comentaris del client o el seu representant.

Els seus plans estan dissenyats per a entregar sempre primer el que tingui major valor comercial. Donat que el 80 per cent del valor representa el 20 per cent de les característiques, els projectes àgils ben realitzats tendeixen a finalitzar d'hora, mentre que la majoria de projectes tradicionals finalitzen tard.

Com a resultat, els clients estan més satisfets i els desenvolupadors de software gaudeixen més de la seva feina. Les metodologies àgils estan basades en el coneixement de que per a tenir èxit, s'ha de planificar per a canviar. Per això han establert processos, com revisions i retrospectives, dissenyats específicament per a desplaçar les prioritats amb regularitat en funció dels comentaris del client i el valor comercial.

Principis àgils

1. La nostra principal prioritat és satisfer al client mitjançant el lliurament primerenc i continu de programari que aportí valor.
2. Acceptem de bon grat canvis als requisits, inclús si arriben cap el final del desenvolupament. Els processos àgils aprofiten el canvi per a donar un avantatge competitiu al client.
3. Lliurem amb freqüència programari que funcioni, des d'un parell de setmanes fins a un parell de mesos, amb preferència per l'escala de temps més curta.
4. La gent de negoci i els desenvolupadors han de treballar junts de manera quotidiana durant tot el projecte.
5. Construïm projectes amb l'ajuda d'individus motivats. Els donem l'entorn i el recolzament que necessiten i confiem en ells per fer la feina.
6. El mètode més eficient i efectiu de comunicar informació cap a i dins d'un equip de desenvolupament és la conversa cara a cara.
7. El programari que funciona és la principal mesura de progrés.
8. Els processos àgils promouen el desenvolupament sostingut. Els promotors, desenvolupadors i usuaris han de ser capaços de mantenir un ritme constant de manera indefinida.
9. L'atenció contínua a l'excel·lència tècnica i al bon disseny millora l'agilitat.
10. La simplicitat, l'art de maximitzar la quantitat de feina que no es fa, és essencial.
11. Les millors arquitectures, requisits i dissenys emergeixen d'equips auto-organitzats.
12. En intervals regulars, l'equip reflexiona sobre com ésser més efectiu, s'afina i ajusta el seu comportament d'acord amb això.

Principals models àgils

Algunes de les metodologies àgils que més es fan servir avui dia són:

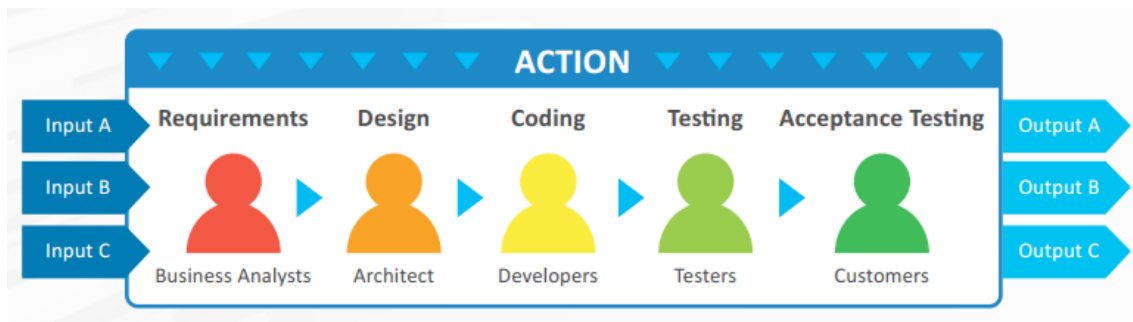
- Extreme programming
- Test drive development
- Agile project management
- Scrum
- Kanban

Per a aquest projecte d'investigació ens centrarem a partir d'ara en aquesta darrera metodologia: Scrum.

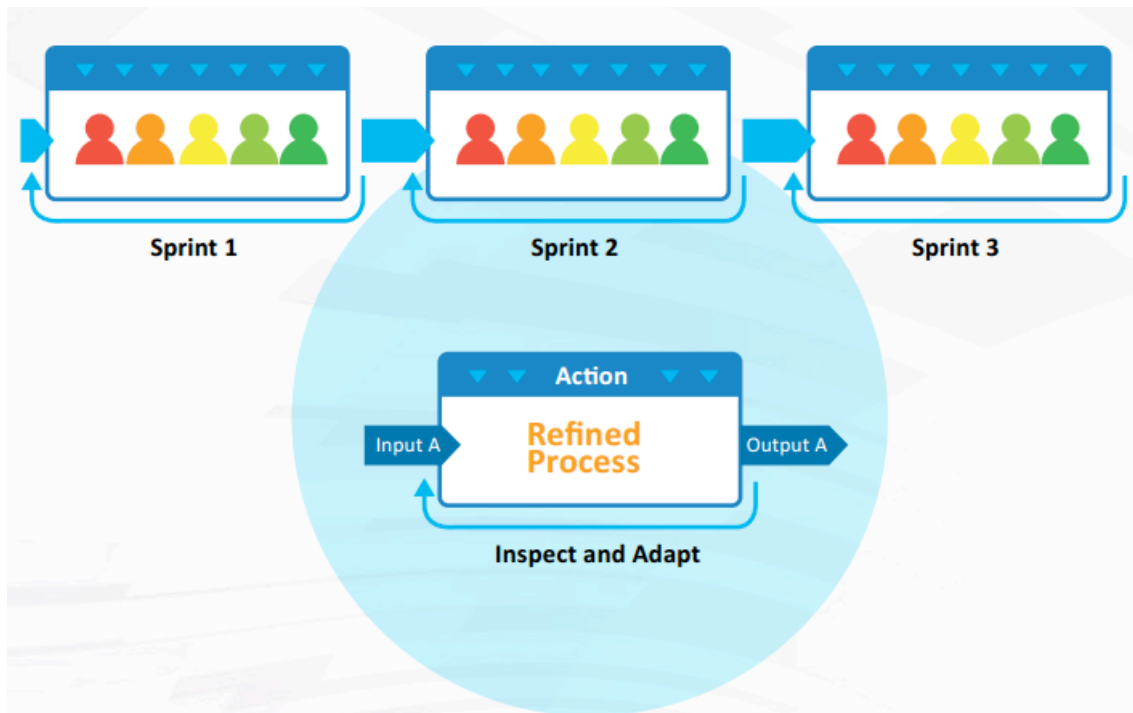
Comparativa entre enfoc tradicional i metodologies àgils

Havent analitzar els dos models de gestió, àgil i tradicional, podem concloure que les principals diferències serien:

- Iteració en el procés de desenvolupament
- Implicació de l'equip al procés
- Implicació del client al projecte
- Gestió dels canvis del projecte



Il·lustració 4 - Model gràfic del desenvolupament en cascada. Font: Northware software.



Il·lustració 5 - Model gràfic del desenvolupament amb Scrum. Font: Northware software.

Com veiem, la estructuració del procés és totalment diferent en un model del l'altre. Mentre que els enfocaments tradicionals com el model en cascada segueixen un avanç lineal, les metodologies àgils com Scrum funcionen per blocs i van refinant el procés i adaptant-se durant el projecte.

L'equip d'Scrum està involucrat en decisions de gestió del projecte, són co-responsables del resultat. L'equip de desenvolupament passa a tenir un paper clau (encara que ja era important a processos com el model en cascada). La diferència clau és la comunicació.

Hi ha tres dels principis àgils que fan especialment èmfasi en el paper que juga l'equip a un procés gestionat amb Scrum:

- *La gent de negoci i els desenvolupadors han de treballar junts de manera quotidiana durant tot el projecte.*
- *Construïm projectes amb l'ajuda d'individus motivats. Els donem l'entorn i el recolzament que necessiten i confiem en ells per fer la feina.*
- *El mètode més eficient i efectiu de comunicar informació cap a i dins d'un equip de desenvolupament és la conversa cara a cara.*

Es fomenta la conversa i l'intercanvi lliure d'idees conflictives. Tot això amb l'objectiu de millorar el procés.

L'equip de desenvolupament ha de treballar també amb la gent de negoci, implicant directament a aquests últims al procés. A les metodologies

tradicionals, el client no forma part del dia a dia, es qui encarrega i amb qui es fa seguiment, però no estan tan involucrats com ho estan amb Scrum.

Sens dubte això enriqueix tot el procés ja que quan hi ha desconexió entre l'equip de desenvolupament i el client, tendeixen a sorgir malentesos derivats de la falta de comunicació o especificitat, donant lloc a preses de decisions deliberades i poc encertades.

Però encara i tots els punts que hem comentat, la diferència més abismal per excel·lència és la que fa referència al segon principi de Agile:

- Acceptem de bon grat canvis als requisits, inclús si arriben cap el final del desenvolupament. Els processos àgils aprofiten el canvi per a donar un avantatge competitiu al client.

Els canvis de requisits en un projecte tradicional poden fer fracassar el projecte o portar a renegociar les condicions contractuals. Amb Scrum, aquest és l'enfoc des de el principi: acceptar canvis als requisits.

Com a resum, aquest és el llistat de les principals diferències entre els processos de gestió àgils i els tradicionals:

Amb les metodologies àgils

1. Desenvolupament iteratiu
2. Desenvolupament centrat en resultats i entregues
3. Entregues parcials per rebre feedback
4. Acceptació de canvis al requisits com a norma
5. Auto-organització de l'equip
6. El client forma part de l'equip

Amb la metodologia en cascada

1. Desenvolupament lineal
2. Desenvolupament centrat en fases de validació (cascada)
3. Entrega a la fase de proves
4. Els canvis de requisits s'han de negociar
5. L'equip és gestionat per un responsable
6. El client rep feedback als seguiments, però no està integrat a l'equip

II·lustració 6 - Principals diferències entre metodologies àgils i tradicionals

3. SCRUM

En els punts anteriors, s'ha fet una introducció al que són les metodologies àgils i la seva diferència amb altres metodologies tradicionals. A continuació, s'explicarà en detall, què és Scrum, en què consisteix la seva metodologia i com s'aplica a la pràctica.

La metodologia Scrum té aquests tres pilars:

- Desenvolupament incremental en lloc d'una planificació i execució completa del desenvolupament
- Aconseguir qualitat posant més èmfasi en el coneixement de l'equip i la seva auto-organització, que no pas en els processos
- Abordar diferents fases del projecte a la vegada, en comptes de fer-ho de manera seqüencial o en cascada.

Pilars de la metodologia Scrum



01 Desenvolupament incremental

En lloc d'una planificació i execució completa del desenvolupament



02 Aconseguir qualitat

Posant més èmfasi en el coneixement de l'equip i a seva auto-organització i no en els processos



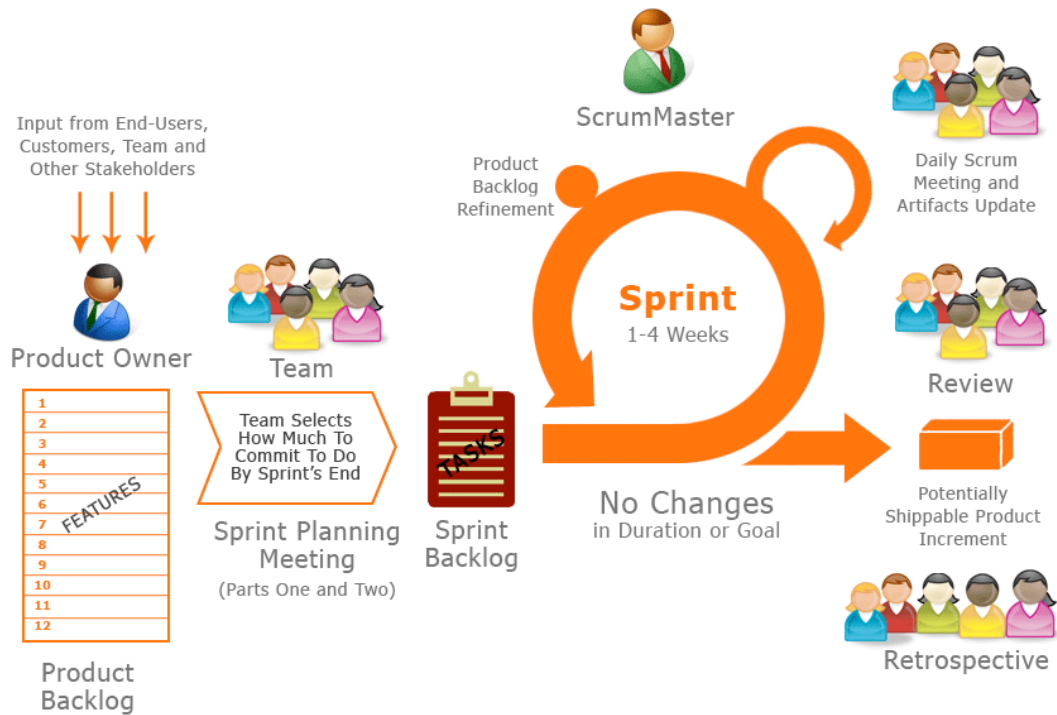
03 Abordar diferents fases a la vegada

En comptes de fer-ho de manera seqüencial o en cascada

Il·lustració 7 - Pilars de la metodologia Scrum

Amb aquesta filosofia s'ha creat un marc de treball.

El següent diagrama resumeix el procés:



II·lustració 8 - Framework Scrum. Font: xsfera.com

Amb la metodologia Scrum, la base del desenvolupament són cicles curts, d'una duració entre 1 i 4 setmanes, anomenats Sprints.

Components de la metodologia

Podem dividir la metodologia Scrum en tres components principals:

- Elements: Reports de els graus d'avanç per al seguiment i control
- Reunions: Punts de seguiment, control i definició
- Rols: Figures involucrades per a la pràctica de la metodologia

Aquesta taula mostra el resum dels components de la metodologia:

Rols

Product owner

Propietari del producte. Normalment és una figura representada pel client o pel contacte de l'equip amb el client.

Scrum master

Facilitador del procés d'scrum. Responsable de vetllar pel bé del procés i de fer complir la metodologia.

Equip de desenvolupament

Equip encarregat del desenvolupament de les tasques del projecte planificades a un sprint.

Elements

Product backlog

Llistat del requeriments del producte a desenvolupar. Conté tant requeriments funcionals definits pel client, com requeriments de l'equip necessaris per a portar a terme el projecte.

Sprint backlog

Subconjunt de tasques del backlog que s'executaran al sprint en curs.

Burn down chart

Gràfica de seguiment a on es mostra el grau d'avanç i l'esforç pendent. S'aplica al backlog i per cada sprint.

Reunions

Sprint planning

Reunió de planificació de la iteració a on se seleccionen les tasques i es valora l'esforç necessari.

Daily meeting

Reunió de seguiment a on l'equip revisa l'sprint. Cada membre respon a: què faràs avui, què vas fer ahir i 3. quins impediments t'has trobat.

Sprint review

Revisió de l'sprint amb client a on es fa una demo del que s'ha desenvolupat.

Sprint retrospective

Anàlisi per part de l'equip de l'sprint finalitzat i àrees de millora.

II-Il·lustració 9 - Taula resum dels elements d'Scrum

A continuació explicarem cada un d'aquests components.

Elements

Product Backlog

El product backlog és el document a on es recull tots els requisits del projecte. Conté les descripcions genèriques de les funcionalitats a desenvolupar i la prioritització per part de negoci.

El responsable de gestionar el product backlog és la figura de product owner, representant del client. S'inclouen estimacions a alt nivell, tant de l'esforç de desenvolupament com de l'impacte pel client.

Sprint Backlog

L'Sprint Backlog és un subconjunt de requisits del projecte (inclosos prèviament al product backlog) i seleccionats per a ser desenvolupats al proper cicle de desenvolupament o sprint.

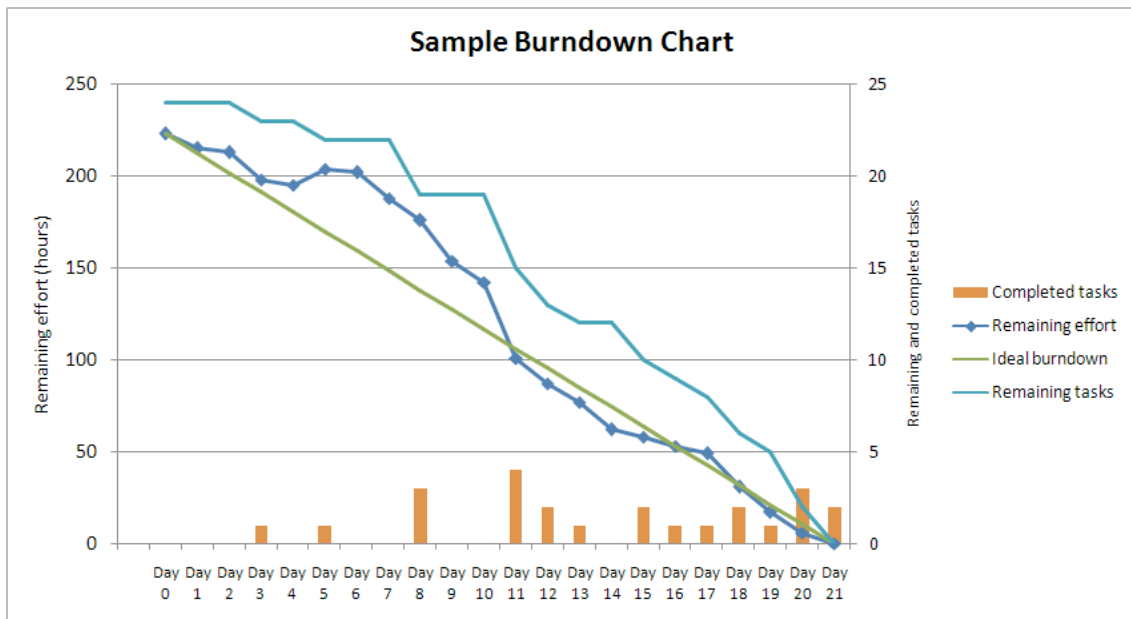
Els requisits es divideixen en tasques i s'assigna una estimació en punts d'història de desenvolupament per part de l'equip. Les tasques no poden tenir un nombre de punts superior a 16, sinó hauran de ser dividides en tasques més petites.

Les tasques del Sprint Backlog no són assignades a persones concretes de l'equip, sinó que serà cada membre qui s'assignarà les tasques a desenvolupar a si mateix en el moment d'inici i segons avanci el sprint.

Burn down chart

La burn down chart ens mostra la evolució del projecte, ja que a l'inici de cada sprint mostra la quantitat de requisits pendents que hi ha al backlog de producte.

Dibuixant la informació a mode de gràfica es podrà veure el progrés del projecte.



Il·lustració 10 - Exemple de Burn down Chart – font: wikipedia

Reunions

Sprint Planning

Els sprints són de duració fixa durant tot el projecte i la seva durada s'estableix al inici. El backlog de producte conté el llistat de totes les funcionalitats a desenvolupar al projecte i dintre de l'sprint Planning se seleccionen les que es desenvoluparan en el següent cicle.

Aquesta reunió es manté al inici de cada sprint i assisteix tot l'equip. L'objectiu és definir quin volum de tasques es pot incloure al proper sprint. La prioritat està marcada per product owner, figura representant del client.

Al final de la reunió estaran definides les tasques a desenvolupar en el següent sprint, amb el criteri de tot l'equip.

Daily Meeting

Cada dia es realitza una reunió amb tots el membres de l'equip per a revisar l'estat de l'sprint en curs. La duració de la reunió no pot ser de més de 15 minuts, ja que l'objectiu és posar-se al dia. Si sorgeixen temes que requereixen més temps, es tractaran en detall fora de la reunió.

Cada membre de l'equip ha de respondre a aquestes tres preguntes:

- ¿Què vas fer ahir?
- ¿Què faràs per avui?
- ¿Has tingut algun problema que t'hagi impedit aconseguir el teu objectiu?

Només els involucrats al projecte poden participar activament a la reunió. També s'anomena a aquesta reunió "stand-up Meeting" ja que sol fer-se de peu per a agilitzar-la. Sempre es fan al mateix lloc i a la mateixa hora.

Sprint Review Meeting

Aquest reunió té lloc després de finalitzar cada sprint i el seu objectiu és analitzar quines tasques del sprint s'han finalitzat i quines no.

S'acompanya d'una demo a on es presenten les tasques finalitzades al product owner, mostrant el producte real desenvolupat.

En aquest moment es recullen els seus comentari i en cas de que sigui necessari s'incorporaran al backlog les tasques o canvis que s'hagin comentat durante la reunió.

Sprint Retrospective Meeting

La reunió de retrospectiva es fa només amb l'equip de treball i té com a objectiu intercanviar impressions sobre l'sprint que ha finalitzat per tal de millorar el procés de treball.

Rols

Product owner

El product owner és la veu del client al projecte. És l'analista de negoci i les seves tasques principals són:

- Detallar les histories d'usuari (funcionalitats a desenvolupar)
- Prioritzar les funcionalitats del backlog
- Col·laborar amb l'equip per aclarir dubtes o temes que puguin sorgir durant un sprint
- Validar les entregues fetes per l'equip a la reunió de revisió

Scrum master

L'Scrum master vetlla pel bé del procés d'Scrum i és el responsable de fer complir la metodologia. El seu objectiu és eliminar els obstacles amb els que es trobi l'equip durant un sprint, per tal de portar-lo a terme.

No s'ha de confondre l'Scrum master amb el líder de l'equip, donat que a Scrum l'equip s'auto-gestiona.

Les seves tasques principals són:

- Guiar el procés d'Scrum
- Eliminar impediments
- Protegir l'equip d'interferències

Equip de desenvolupament

L'equip de desenvolupament és responsable d'entregar el producte. És recomanable que l'equip estigui format per entre 3 i 9 persones, amb habilitats complementaries per a dur a terme el projecte.

Les seves tasques principals són:

- Desenvolupar el producte
- Definir les tasques
- Estimar els costos
- Garantir la qualitat

Requisits i abast del producte

En un projecte amb Scrum sempre tindrem el primer sprint inicial, anomenat sprint 0 a on es defineix el producte inicial a desenvolupar.

En aquest moment no es podem fer estimacions precises, ja que no es disposa de mètriques sobre les que basar-se, per tant, l'enfoc d'aquest primer sprint inicial és de desenvolupar el nucli del producte. Les tasques incorporades al backlog s'estimaràn en dies.

En cas de que es disposi d'informació prèvia del rendiment de l'equip, es podrà fer una estimació més encertada i basant-se en punts d'història.

Les dues fites principals que ens permetran acotar l'abast del projecte són:

1. Definir els objectius del projecte

És imprescindible definir l'enfoc del projecte, però no només això, sinó que l'equip ha d'entendre quins són els objectius i les prioritats. Entenent les necessitats del client i el per què es desenvolupa el projecte es podrà donar un millor enfoc en el dia a dia per a abordar els desenvolupaments.

2. Definició del backlog inicial

La llista inicial d'elements del backlog en aquest moment del projecte vindrà definida pel product owner, que donarà prioritats a detallar els elements que permetin disposar d'un producte mínim viable per a obtenir feedback quan abans per part del client.

En aquesta fase, es constituirà també l'equip de treball, pilar imprescindible del projecte. D'acord a la definició del projecte es buscaran persones per a l'equip que tinguin el perfil més adient per a portar a terme el projecte.

És imprescindible a l'inici del projecte explicar detalladament la metodologia a l'equip, així com instaurar les reunions diàries i especificar els períodes entre entregues. De la mateixa manera s'haurà de revisar el backlog per a que tothom tingui clares les funcionalitats que contindrà el producte a desenvolupar.

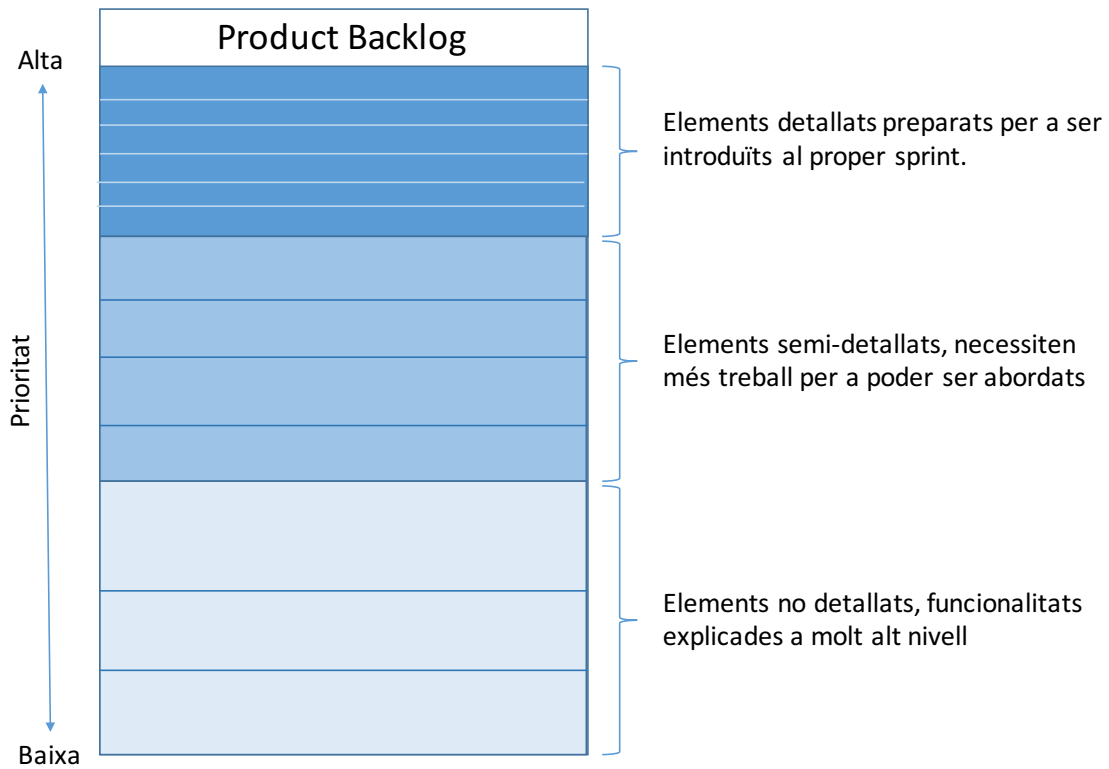
Backlog

Com s'ha introduït abans, el backlog conté un recull de totes les funcionalitats a desenvolupar i la seva prioritització en base a les necessitats del client. Aquest llistat és gestionat pel product owner amb l'ajuda del Scrum master, que donarà suport a les estimacions de les funcionalitats.

El backlog començarà amb uns requisits generals de manera que l'equip pugui començar a desenvolupar. Segons avanci el projecte, evolucionarà també el backlog, donat que sortiran tasques o requisits secundaris.

Els elements que conté el backlog s'anomenen històries d'usuari i s'aniran detallant segons avanci el projecte. El que si tindrà cada història des de l'inici és una estimació i la prioritat pel negoci (a més de la descripció).

Així, el backlog de producte podríem representar-lo així:



II-lustració 11 - Representació del product backlog

Aquest és un exemple de product backlog real d'un projecte:

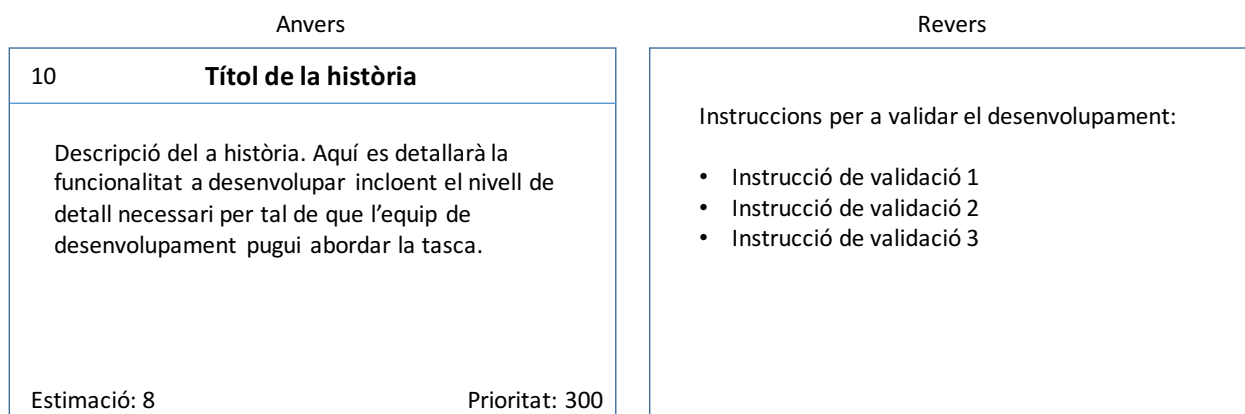
ID	Theme	As a/an	I want to...	so that...	Notes	Priority	Status
2	Game	moderator	create a new game by entering a name and an optional description	I can start inviting estimators	If games cannot be saved and returned to, the description is unnecessary	Required	done
2	Game	moderator	invite estimators by giving them a url where they can access the game	we can start the game	The url should be formatted so that it's easy to give it by phone.		done
5	Game	estimator	join a game by entering my name on the page I received the url for	I can participate			done
6	Game	moderator	start a round by entering an item in a single multi-line text field	we can estimate it			done
8	Game	estimator	see the item we're estimating	I know what I'm giving an estimate for			done
40	Game	participant	always have the cards in the same order across multiple draws	it's easy to compare estimates	-	Replaced with A08 because I didn't want the story to talk about "the same order" as that might be a UI implementation detail	todo
35	Non-functional	user	have the application respond quickly to my actions	I don't get bored			done
36	Non-functional	user	have nice error pages when something goes wrong	I can trust the system and it's developers			done
A11	Non-functional	Researcher	results to be stored in a non-identifiable way	I can study the data to see things like whether estimates converged around the first opinion given by "estimator A" for example	No names or story text should be stored but we should store each card of each hand, know who played it, and know the final accepted estimate		
A05	Game	moderator	edit an item in the list of items to be estimated	so that I can make it better reflect the team's understanding of the item			
22	Archive	moderator	export a transcript of a game as a CSV file	I can further process the stories and estimates	Exported file should be directly importable back into the system.		done

II-lustració 12 - Exemple de product backlog. Font: Mountain goat software

Histories d'usuari

Les histories d'usuari és el nom que se li dóna al requisits que conformen el backlog i contenen la descripció de la funcionalitat a desenvolupar. Acostumen a definir-se en targetes o post-its, ja que contenen informació tant en l'avers com al revers.

Com es veu a la següent imatge, una història d'usuari té format de targeta, amb informació per davant i per darrere.



Il·lustració 13 - Format d'una història d'usuari

La informació a recollir a una història d'usuari és:

- Identificador: Identificador únic de la història d'usuari
- Títol: Títol descriptiu de la història
- Descripció: descripció detallada de la història
- Estimació: Tems estimat en unitats de desenvolupament
- Prioritat: Priorització de la història respecte a les altres històries del backlog. Quan més gran sigui el nombre, més prioritat tindrà.
- Instruccions per validar: indicacions que confirmaran que el desenvolupament és correcte

Sprint

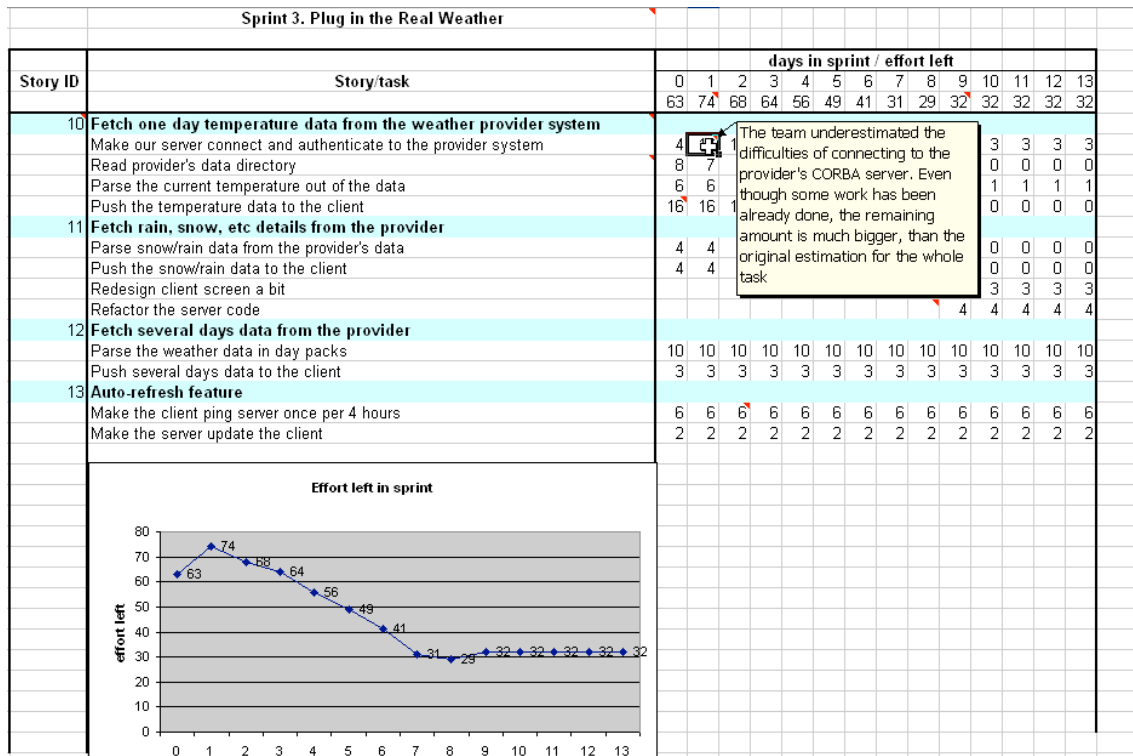
Un sprint és un cicle de desenvolupament. Per a cada sprint es prepara un sprint backlog, que és un subconjunt del product backlog, però només conté les tasques que seran abordades durant l'sprint.

Aquest backlog és elaborat per l'equip de projecte a la reunió d'sprint Planning. Les valoracions d'esforç són revisades també per l'equip. En cas que es valorin tasques amb una estimació superior a 16 hores, es crearan tasques més petites.

L'equip repartirà entre els seus membres les tasques a desenvolupar durant l'sprint. Poden haver-hi tasques que no quedin assignades en un primer moment, però les persones de l'equip que finalitzin la tasca que

tenen en curs s'assignaran una nova tasca ells mateixos entre les que estan sense assignació.

Aquest és un exemple d'sprint backlog d'un projecte real:



Il·lustració 14 - Exemple d'Sprint backlog

Per a cada sprint es mesura l'esforç que queda pendent respecte del esforç total planificat per l'sprint. D'aquesta manera, es pot fer el seguiment del grau d'avanç del sprint en curs.

El seguiment diari de les tasques del sprint, assignades o no, es gestiona mitjançant un canvas anomenat Task board.

El task board permet separar les tasques del sprint en estat. Habitualment aquests estats són:

- Pendent: tasques del sprint no assignades
- En curs: tasques sobre les que està treballant un membre de l'equip
- En proves: tasques desenvolupades i en procés de proves
- Fet: Tasques finalitzades i provades

El fet de fer servir post-its permet tenir el task board visible per a tot l'equip, moure fàcilment les tasques d'estat i assignar-les.

A continuació veiem un exemple d'un Task board d'Scrum:

Story	To Do	In Process	To Verify	Done	
As a user, I... 8 points	Code the... 9 Code the... 2 Test the... 8	Test the... 8 Code the... 8 Test the... 4	Code the... DC 4 Test the... SC 8	Test the... SC 6	Code the... D Test the... SC 8 Test the... SC Test the... SC Test the... SC 6
As a user, I... 5 points	Code the... 8 Code the... 4	Test the... 8 Code the... 6	Code the... DC 8		Test the... SC Test the... SC Test the... SC 6

Il·lustració 15 - Exemple de Task Board. Font: eclipse.org

Com es veu a la imatge, hi ha una columna inicial que té la història a la que fan referència les tasques sobre les que s'està treballant.

Planificació

Planificar un sprint de manera acurada requereix de conèixer bé l'equip de treball i comptar amb un historial previ que permeti fer una estimació acurada. Basant-nos en en sprints anteriors podem determinar quin nombre d'històries podem incloure a cada sprint.

Planificació de poker

Un dels mètodes més comuns per a la planificació de tasques amb Scrum és la planificació de poker. El seu nom ve donat perquè les estimacions es representen en cartes i cada membre de l'equip mostra la carta corresponent a la estimació d'esforç que creu que tindrà la tasca que s'està valorant.

Mostra de cartes imprimibles per a fer la estimació mitjançant la planificació de poker:



Il·lustració 16 - Planificació de poker

Actualment existeixen també aplicacions per a mòbil amb aquestes valoracions, a on cada membre de l'equip selecciona i mostra la valoració de la tasca, però el procediment és el mateix.

El procés per a fer les estimacions és el següent:

- Es planteja una història de les que s'inclouran al sprint
- Cada membre de l'equip selecciona una valoració per a la tasca
- Tots el membres mostren la seva valoració
- Si les valoracions són molt diferents, es discutirà sobre la tasca posant en comú les raons de la valoració de cada membre i acotant el significat de la tasca.
- El procés es repetirà fins a que les estimacions siguin semblants

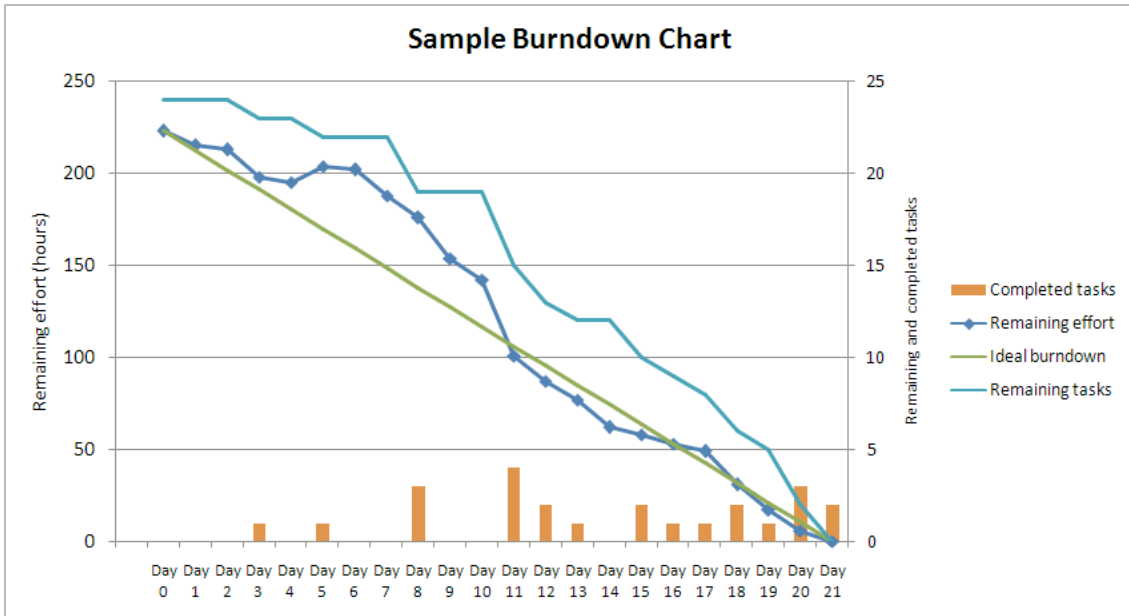
La carta amb el valor 0 indica que o bé la tasca ja està feta o que pràcticament no requereix esforç. L'interrogant ens indica que qui ha mostrat aquesta valoració no té una idea clara del esforç que pot comportar. L'infinit ens indica que la tasca excedeix la valoració màxima de 100, per tant, s'hauria de descompondre en històries més petites.

Seguiment, control i mesuraments

En un projecte d'Scrum tindrem en seguiment del sprint actual, que ens ajudarà a portar el seguiment del burn down chart global del projecte.

Com ja s'ha mencionat en els punts anteriors, a cada sprint, diàriament es fa un seguiment i cada sprint té el seu propi burn down chart.

A continuació veiem una gràfica d'un sprint de 21 dies:



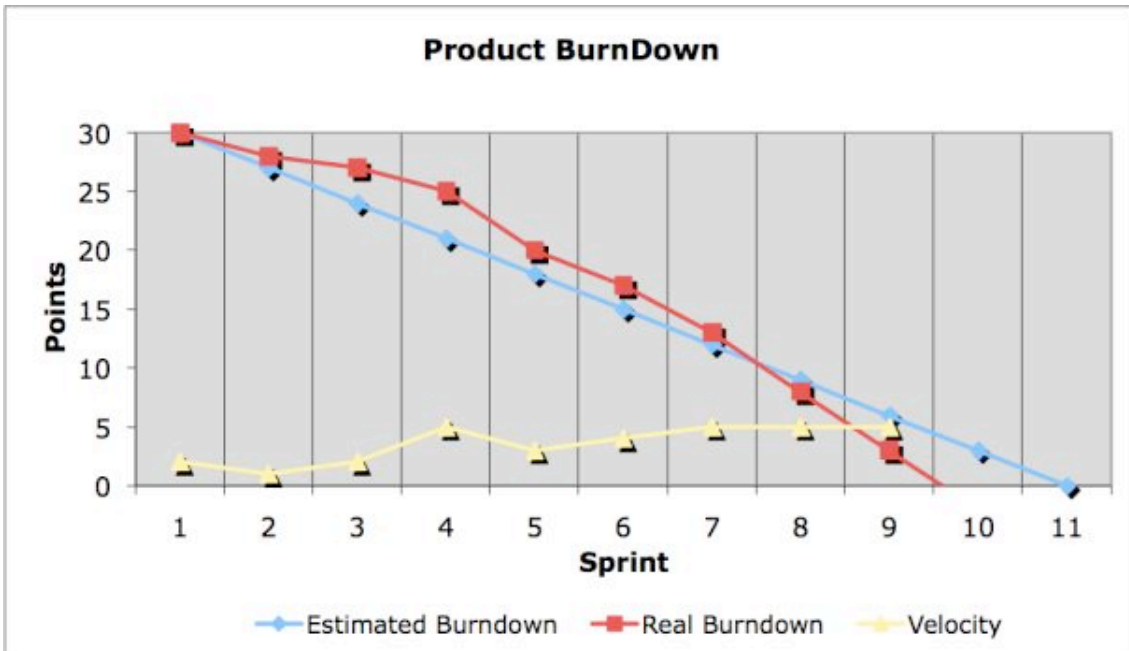
Il·lustració 17 - Sprint burn down chart. Font: Wikipedia

Dintre de la gràfica tenim diverses mètriques: tasques completades del dia, esforç pendent del sprint, tasques restants i la línia ideal sobre la que ens hauríem de moure.

És freqüent que els primers dies la gràfica mostri un nombre de tasques superior al que idealment hauríem d'haver tancat. Això es dona perquè el procés de tancar una tasca no és immediat, per tant, aquesta mètrica no avançarà fins a que no avanci l'sprint.

De manera similar, el projecte global té el seu propi burn down chart. En compte de reflectir els dies del projecte, el que reflecteix són els sprints del projecte.

A continuació veiem un exemple:



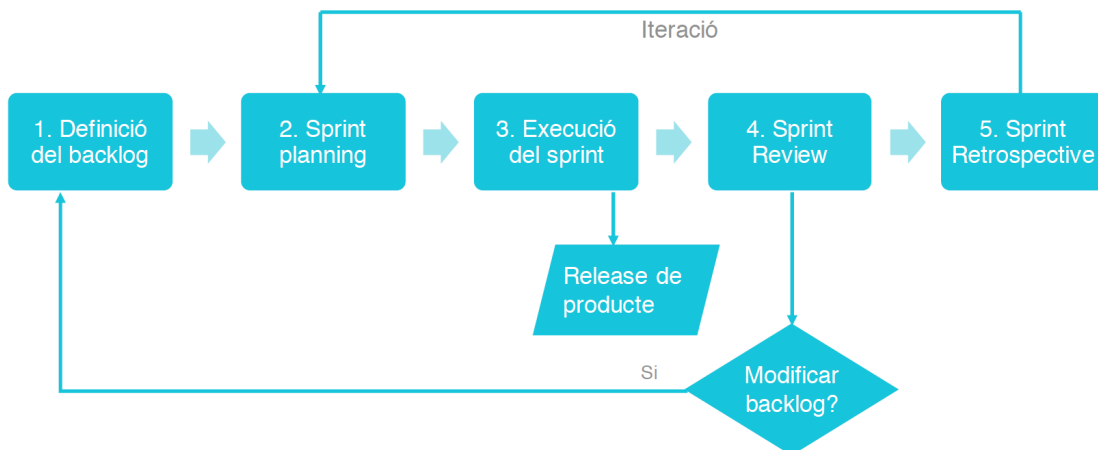
II·lustració 18 - Product burn down chart. Font: Scrum institute

Com veiem a la gràfica, tenim la estimació global del projecte en blau i l'avanç real en vermell. D'aquesta manera podem controlar si estem complint la previsió o si ens estem desviant.

Poden afegir-se a la gràfica altres mètriques. En aquest exemple veiem també la mètrica de la velocitat dels desenvolupaments.

Resum del procés d'Scrum

La següent gràfica, mostra el resum del procés Scrum que s'anirà repetint de manera iterativa fins a finalitzar el projecte.



II·lustració 19 - Procés Scrum resumit

4. Cas pràctic: desenvolupament d'un portal corporatiu amb Scrum

A continuació es planteja en un exemple de com abordar un projecte, amb requeriments aplicables a un projecte real i amb la particularitat de que l'equip de projecte és distribuït.

Requeriments del projecte

El client planteja la necessitat de desenvolupar un nou portal corporatiu per a la companyia. A més, vol que aquest portal serveixi com a base per tal de poder reutilitzar-ho per als portals de cada país de la companyia, després de llençar el portal inicial de la corporació.

Com a requisits tècnics ens ve donat el producte, un gestor de continguts que el client ja ha fet servir anteriorment i que ja coneix.

Com a requisits interns de la nostra empresa ens ve donat l'equip. Una part de l'equip estarà ubicada a Barcelona (5 persones) i una altra a Saragossa (3 persones).

En aquest supòsit assumim que l'equip té la formació i preparació necessària per a portar a terme el projecte i que està balancejat entre perfils amb més i menys experiència. L'equip de Barcelona té més experiència que el de Saragossa, que està conformat per programadors que porten poc temps a l'empresa.

Requeriments funcionals

El portal ha de constar de 5 seccions principals de contingut:

- Informació corporativa
- Accionistes e inversors
- Responsabilitat corporativa
- Clients
- Equip i activitats

Totes aquestes seccions "pengen" d'una secció d'inici a on està la home principal del portal, mostrant al menú les seccions enumerades. Cada secció té la seva pròpia subhome i sub-menu de continguts.

Dintre de l'arbre de continguts trobem pàgines editables amb una actualització més o menys freqüent i funcionalitats a mida.

La complexitat ve donada en la definició de com construïm la base del gestor de continguts per tal de que les pàgines siguin personalitzables i puguin gestionar-se per l'equip de gestió de continguts.

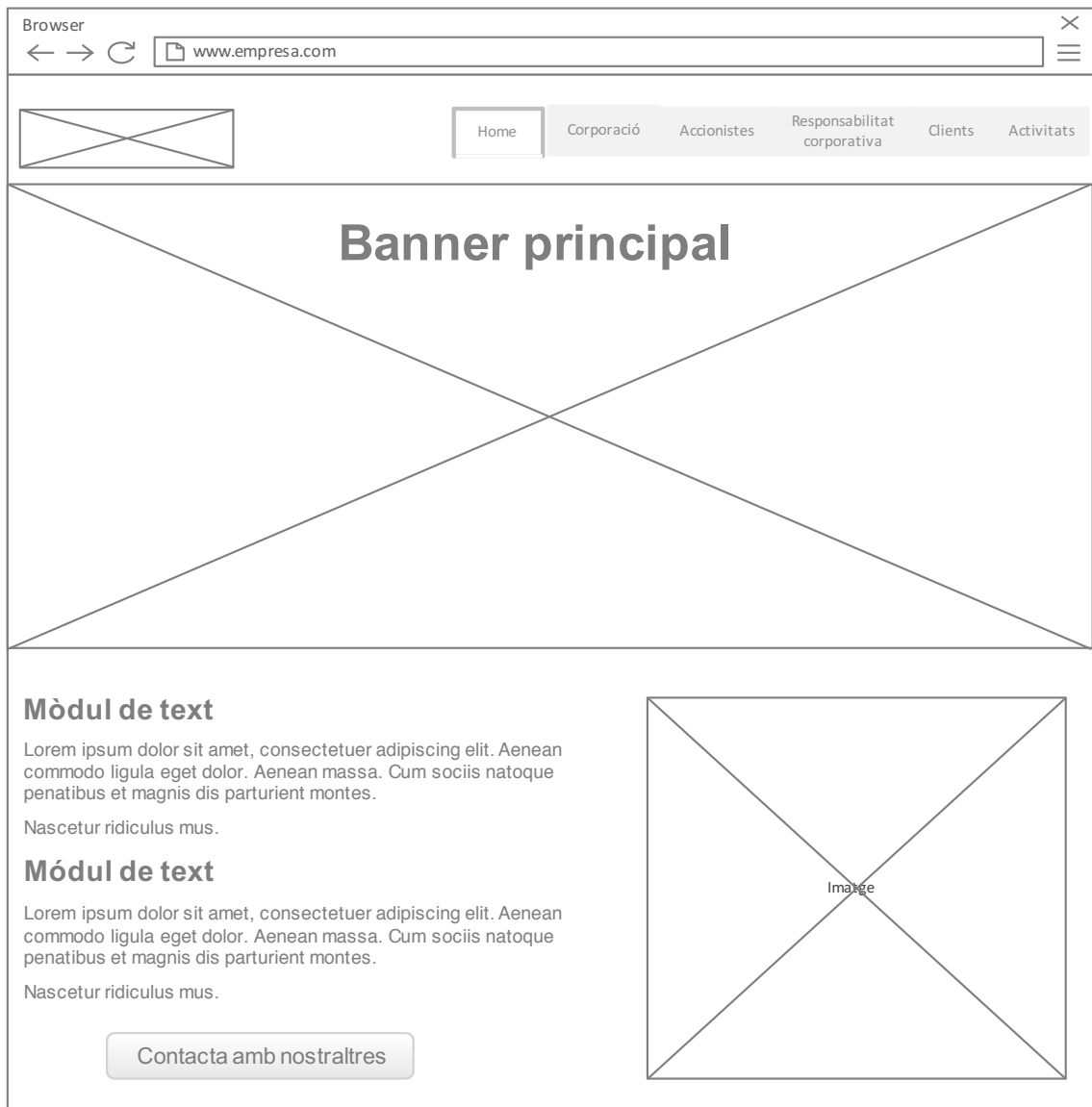
La maquetació ha de ser responsive i complir el nivell doble A d'accessibilitat. Aquest punt ho desenvoluparà un equip independent que entregarà els fitxers de la maqueta (marcatge, estils i javascript) a l'equip de desenvolupament.

Les urls han de ser semàntiques i s'ha de tenir en compte que el portal està en 3 idiomes (castellà, català i anglès). També s'han de complir les pautes de seo on-page recomanades per google per tal de posicionar el portal als resultats de cerca dels buscadors.

La gestió dels usuaris i del flux de edició i publicació de continguts ve controlada pel propi gestor de continguts que es fa servir con a base. Te rols diferents pels usuaris en base a la seva tasca al procés de publicació.

Així, primer de tot es defineix l'arquitectura de continguts, que és entregada a l'equip de disseny per a analitzar els elements de cada pàgina i dissenyar els wireframes. Aquesta documentació és entregada a l'equip de desenvolupament. D'aquí és d'on extraurem realment la funcionalitat a desenvolupar al gestor de continguts.

Els wireframes seran com es mostra a la següent imatge:



Il·lustració 20 - Mostra de wireframe del projecte

En aquest supòsit assumim que:

- L'equip responsable de gestionar els continguts és independent i forma part de l'equip de client. Ells seran els responsables de carregar el contingut una vegada estigui finalitzat el desenvolupament.
- L'equip de disseny, en base als wireframes generarà un disseny i el validarà amb el client per a entregar-lo al equip de maqueta.
- L'equip de maqueta treballa de manera semi-independent del equip de desenvolupament, però les seves tasques no seran contemplades en aquest supòsit. El product owner informarà a l'equip de maqueta dels ajustos que pugui necessitar l'equip de desenvolupament.

Els requisits del framework base:

- Tenim una entitat principal que és la pàgina, sobre la que la s'aplicarà una plantilla. Les plantilles tipus seran:
 - Home
 - Contingut
 - Funcionalitat

Es a dir, quan comencem a crear una pàgina, el primer que haurem d'escollir és si es tracta d'una home, una pàgina de continguts o una funcionalitat a mida.

Per a la primera versió del portal, la plantilla corresponent a funcionalitat es contemplarà com a un iframe a on indicar la url de la funcionalitat a integrar. En una fase posterior es desenvoluparan les funcionalitats a mida.

- Mòduls de contingut
 - Tindrem diferents mòduls que anirem incorporant a les pàgines per tal d'aconseguir la estructura de continguts que volem.
 - Els mòduls seran:
 - Text html. Permet introduir text i imatges (el gestor de continguts controla el seu propi repositori d'imatges).
 - Baners
 - Caixes amb icones o imatges
 - Columnes de contingut
 - Portfoli
 - Àrea lateral de la pàgina customizable
 - Taules
 - Pestanyes
 - Caixes de call to action
 - Formulari de contacte

Aquests requeriments conformen el desenvolupament base del projecte per a poder construir portals amb aquests elements comuns definits.

Tenint clars els requeriments, passarem a detallar la planificació del projecte, creant el backlog i la planificació dels sprints, juntament amb la estimació d'esforços inicial i el càlcul del cost del projecte.

Planificació del projecte

Amb els requisits que tenim, es defineix el backlog inicial del projecte.

Al backlog s'ha contemplat per cada funcionalitat, la seva prioritat indicant si es alta, mitja o baixa i una estimació inicial assignant un valor fent servir la planificació de poker.

ID	Descripció	Prioritat	Estimació
1	Configuració inicial del gestor de continguts	Alta	40
2	Creació de menú principal	Alta	13
3	Creació de sub-menú	Alta	8
4	Plantilla de pàgina tipus home	Alta	20
5	Plantilla de pàgina tipus contingut	Alta	20
6	Plantilla de pàgina tipus funcionalitat	Baixa	3
7	Configuració de urls semàntiques	Alta	40
8	Configuració multi-idioma	Mitja	20
9	Configuració de rols d'usuari	Mitja	5
10	Alta inicial d'usuaris de la plataforma	Mitja	3
11	Integració de la maqueta lliurada	Alta	40
12	Mòdul columnes	Mitja	20
13	Mòdul text html	Alta	8
14	Mòdul banner	Alta	8
15	Mòdul caixa amb icona o imatge	Mitja	8
16	Mòdul portfoli	Mitja	13
17	Mòdul Taula	Baixa	5
18	Mòdul Pestanyes	Baixa	8
19	Mòdul Caixes de call to action	Baixa	5
20	Mòdul Àrea lateral customitzable	Mitja	20

D'aquesta estimació inicial tenim una estimació total de 307 punts.

S'ha configurat el següent equip per a portar a terme el projecte:

Oficina de Barcelona

- Product owner
- Scrum master
- 1 analista tècnic
- 1 programador sènior
- 1 programador junior

Oficina de Saragossa

- 1 programador senior
- 2 programadors juniors

L'analista tècnic no estarà dedicat al desenvolupament, sinó que serà el responsable d'assegurar la qualitat del codi lliurat a cada entrega, donar suport tècnic a l'equip i resoldre dubtes o qüestions que sorgeixin en el dia a dia sobre el desenvolupament.

Per tant, l'equip de desenvolupament està format per 2 programadors sènior i 3 programadors junior.

Per a aquest supòsit tindrem en compte que l'equip ha treballat anteriorment i per tant, podem fer una estimació de la càrrega de feina que l'equip pot assumir en un sprint. A cada sprint poden assumir-se entre 35 i 40 punts i els sprints tindran 10 dies de duració.

Amb això se'n pot deduir que es necessitaran 9 sprints per a finalitzar el projecte, per tant la seva duració serà de 90 dies laborables.

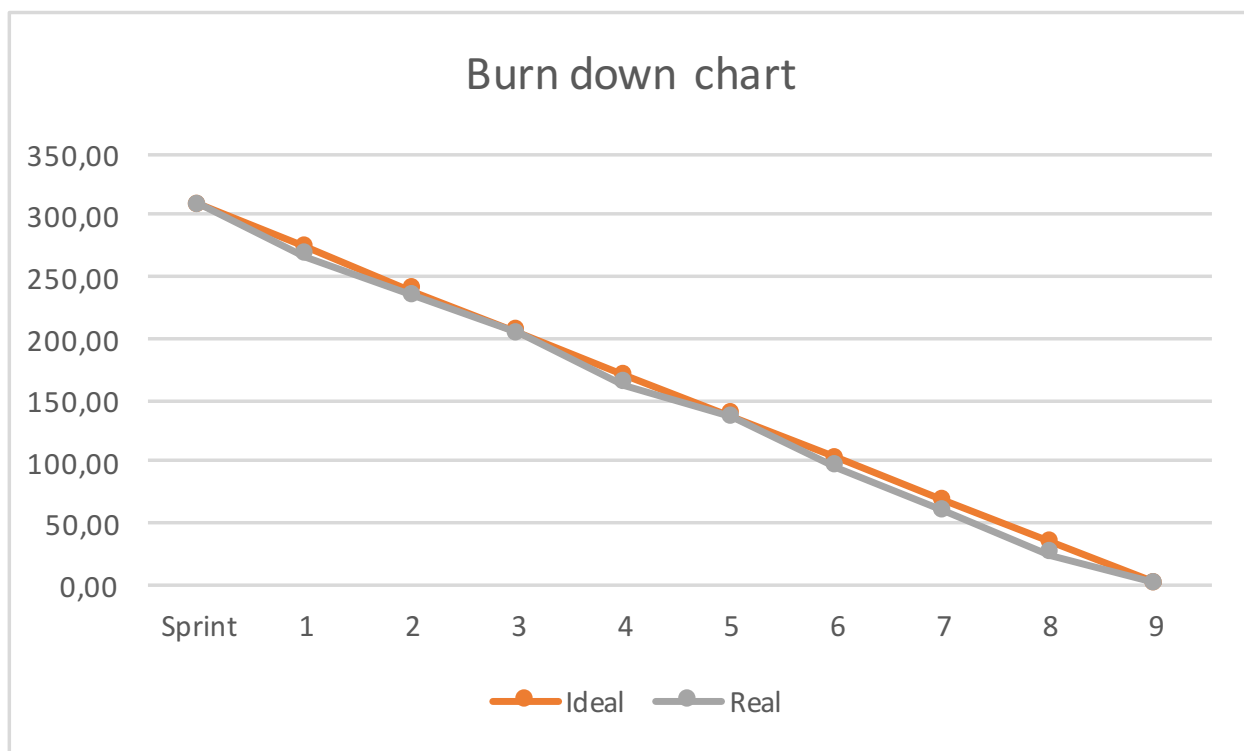
Ara podem abordar la estimació de costos del projecte:

Ubicació	Perfil	Dedicació al projecte	Cost jornada	Dies dedicació al projecte	Cost total
Barcelona	Product owner	50%	100,00 €	90	4.500,00 €
Barcelona	Scrum master	50%	90,00 €	90	4.050,00 €
Barcelona	Analista tècnic	100%	80,00 €	90	7.200,00 €
Barcelona	Programador S.	100%	60,00 €	90	5.400,00 €
Barcelona	Programador J.	100%	40,00 €	90	3.600,00 €
Saragossa	Programador S.	100%	50,00 €	90	4.500,00 €
Saragossa	Programador J. 1	100%	35,00 €	90	3.150,00 €
Saragossa	Programador J. 2	100%	35,00 €	90	3.150,00 €
Total					35.550,00 €

Com a provisió per a gestions de canvis i ajustos a les estimacions, calculem un 10% del total del projecte, que es sumarà al pressupost final.

Amb aquest detall, podem definir la estimació del que seria l'avanç ideal del projecte, dividint les tasques del backlog en sprints i definint la primera burndown Chart del projecte:

ID	Descripció	Prioritat	Estimació	Sprint
1	Configuració inicial del gestor de continguts	Alta	40	0
2	Creació de menú principal	Alta	13	1
3	Plantilla de pàgina tipus home	Alta	20	1
4	Creació de sub-menú	Alta	8	2
5	Plantilla de pàgina tipus contingut	Alta	20	2
6	Plantilla de pàgina tipus funcionalitat	Baixa	3	2
7	Configuració de urls semàntiques	Alta	40	3
8	Configuració multi-idioma	Mitja	20	4
9	Configuració de rols d'usuari	Mitja	5	4
10	Alta inicial d'usuaris de la plataforma	Mitja	3	4
11	Integració de la maqueta lliurada	Alta	40	5
12	Mòdul text html	Alta	8	6
13	Mòdul columnes	Mitja	20	6
14	Mòdul banner	Alta	8	6
15	Mòdul caixa amb icona o imatge	Mitja	8	7
16	Mòdul portfoli	Mitja	13	7
17	Mòdul Taula	Baixa	5	7
18	Mòdul Pestanyes	Baixa	8	7
19	Mòdul Caixes de call to action	Baixa	5	8
20	Mòdul Àrea lateral customitzable	Mitja	20	8



Il·lustració 21 - Burndown chart inicial del projecte

Seguiment

Aplicant la metodologia d'Scrum, estan definits els següents punts de seguiment al projecte:

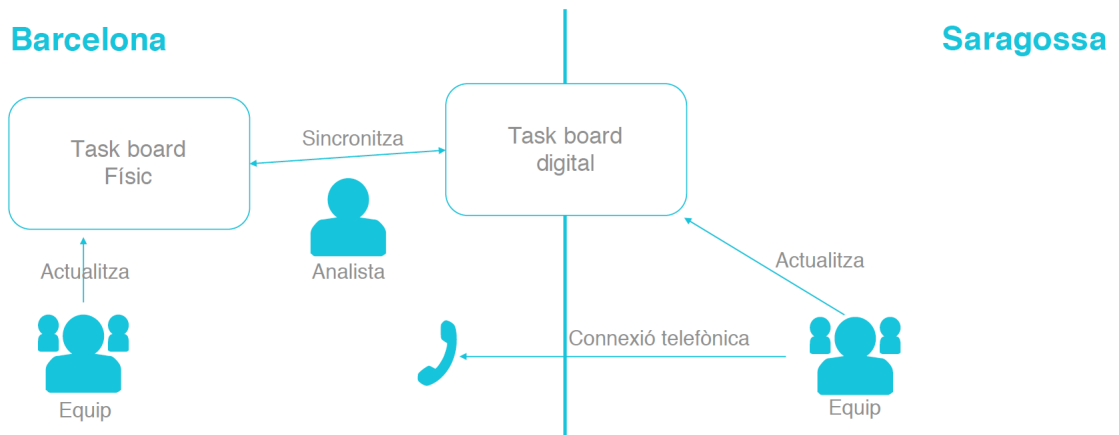
Reunió	Periodicitat	Objectiu	Participants
Sprint planning	Al inici de cada sprint	Definir tasques del proper sprint i valorar-les	Tot l'equip
Daily meeting	Diàriament	Seguiment diari del sprint en curs	Equip de desenvolupament
Sprint review	Al final de l'sprint	Demo al product owner del treball finalitzat	Tot l'equip
Sprint retrospective	Al final de l'sprint	Analitzar com ha anat l'sprint	Tot l'equip

En aquest supòsit tenim la particularitat de comptar amb un equip distribuït en dos ubicacions geogràfiques: Barcelona i Saragossa.

Per tal d'optimitzar el procés, s'han pres les següents decisions:

- Donat que el client i els seus representats al projecte es troben ubicats a Barcelona, l'equip de Saragossa no participarà a les reunions de Sprint review. En finalitzar aquestes reunions, seran informats del seu resultat i de les conclusions.
- Per a les reunions d'sprint Planning, es comptarà amb un servei de vídeo conferència per tal de que tot l'equip pugui participar.
- De la mateixa manera, per a les reunions de retrospectiva també es farà servir la vídeo conferència per a que estigui present tot l'equip.

El seguiment diari és el punt més crític, ja que excloure a una part de l'equip d'aquestes reunions pot posar en risc l'èxit del projecte. Per a fer aquest seguiment es comptarà amb el següent:



II·l·lustració 22 - Organització del seguiment diari de l'equip

- Un task board d'Scrum físic a les oficines de Barcelona, al costat dels llocs de treball de l'equip.
- Aquest task board estarà replicat digitalment amb l'aplicació Trello, a on tot l'equip tindrà accés.
- L'equip de Saragossa actualitzarà les seves tasques al task board digital i l'equip de Barcelona actualitzarà tant el task board digital com el físic.
- L'analista tècnic de Barcelona serà el responsable de mantenir sincronitzat el task board digital, replicant la informació del task board físic i a l'inrevés, afegint la informació de l'equip de Saragossa al task board físic per tal de que tot l'equip disposi de la mateixa informació.
- Els daily meetings es faran amb tot l'equip, incloent per via telefònica a l'equip de Saragossa amb un telèfon en mans lliures. D'aquesta manera es permet fer participants a tots els membres de l'equip a la reunió diària i portar un seguiment adequat del sprint.

Anàlisi

Després de valorar aquest procés plantejat per a organitzar el dia a dia amb l'equip, tenint en compte que esta a dos ubicacions físiques diferents s'aprecia el següent:

- Es possible mantenir la metodologia i aplicar-la correctament fent servir eines de comunicació virtual.
- L'equip que no es troba en persona es pot perdre a les reunions diàries o distreure's amb més facilitat que l'equip que està en persona.
- Poden haver-hi problemes per a coincidir tots en el mateix moment d'iniciar la reunió diària, sobre tot per a l'equip que es connecta per telèfon.
- Les reunions poden allargar-se en fer-se per telèfon si surten imprevistos, dificultats o temes a comentar que generen debat a l'equip.
- Alguns membres de l'equip poden descuidar-se d'informar correctament les tasques al task board físic i al digital a la vegada per a mantenir la informació sincronitzada.

Com a millores al procés s'introduiria:

- Possibilitat de que la daily Meeting es pugui fer per vídeo conferència, bé sigui en una sala a on es pugui penjar el task board o fent-lo servir digital.
- El fet de fer servir vídeo conferència provoca que l'equip es vegi en persona, cosa que facilita el seguiment de la reunió i la implicació per part de tots els membres de l'equip a la reunió, independentment de la seva ubicació geogràfica.
- Una altra opció a valorar seria fer la daily Meeting en dos equips per separat, assignant a un responsable de cada equip que en finalitzar les reunions fessin posta en comú del temes sorgits.

5. Conclusions

El projecte realitzat tracta els diferents punts de vista per a la gestió d'un projecte aplicant metodologies àgils, com és Scrum i mostra les principals diferències entre altres mètodes de gestió de projectes tradicionals i aquestes noves metodologies.

Sens dubte, les metodologies àgils ens mostren un nou paradigma per a la gestió de projectes i semblen la evolució lògica cap a la que dirigir-se en el camp de la gestió de projectes.

Malgrat tot, existeixen dos factors que són especialment crítics i que depenen de factors culturals, per tant, no han estat analitzats detalladament al desenvolupament del projecte, però que no podem perdre de vista.

El primer d'aquests factors es la constància i dedicació que requereix aquesta metodologia per tal de que el projecte funcioni i es gestioni amb èxit.

No es tracta d'una metodologia que es pugui aplicar un cop a la setmana per a fer un seguiment. O la segueixes o no, però fer-ho a mitges pot comportar més problemes que beneficis a un projecte.

En la meua experiència personal, aquest ha estat un dels factors pels quals vàrem tornar a aplicar metodologies tradicionals després d'haver implantat Scrum a un dels projectes als que participava.

L'equip va començar amb molts ànims i tothom veia que la informació i el seguiment que ens proporcionava Scrum era realment àgil i beneficiava a tot l'equip. Ens proporcionava un control detallat diari que facilitava moltíssim les tasques de gestió.

Els problemes van sorgir quan es van prioritzar urgències del dia a dia per sobre de l'aplicació de la metodologia. Això va portar al descontrol intern del task board en no fer-se rigorosament les daily Meetings. Aquests dos factors van ser decisius per a decidir tornar al plantejament de gestió anterior.

La maduresa de l'equip ha de ser un factor clau a l'hora de decantar-nos per a aplicar metodologies àgils de gestió a un projecte. En un equip immadur, poc compromès o que no entén el procés pot portar el projecte a un fracàs rotund.

El segon factor crític és la negociació del contracte amb el client. Com s'ha vist durant el document, els tres factors clau per a un projecte són: l'abast, el temps i el cost.

En un model tradicional, el cap de projecte és la figura que es troba al mig d'aquest triangle per tal de vetllar per a que aquest tres factors funcionin correctament.

Un dels punts d'Scrum i que no tots els clients poden acceptar és que algun d'aquests tres paràmetres ha d'estar obert. Per a que la metodologia d'Scrum funcioni correctament hauríem de tenir tancat només un parell de factors:

- Abast i temps tancats i cost variable segons avanç i canvis aplicats al projecte
- Temps i cost tancats i abast variable en funció dels altres dos factors
- Cost i abast tancats i planificació flexible

En un escenari amb Scrum, seria ideal no tenir tancat el factor del cost i anar-lo revisant amb el client segons avancés el projecte. Tancar només unes bases en el moment inicial en quant a tarifes per hora de dedicació, per no haver de negociar els preus a cada facturació. Però això comportaria que el client no sabria el cost total del projecte fins a que no estigués finalitzat.

Aquest punt, pot ser implantejable per a la majoria de clients avui dia. No tenir un cost tancat pot fer que el projecte no es validi internament i no surti o no s'adjudiqui.

El desenvolupament d'aquest projecte s'ha fet seguint la metodologia Scrum i encara i ser un projecte individual, he pogut comprovar en primera persona alguns punts forts de la metodologia i d'altres febles:

- El seguiment diari fa que en cap moment es perdi la visió global del projecte i del que queda pendent de fer per a finalitzar.
- La estimació de tasques ha estat bastant encertada en general, cosa que ha permès que no haguessin imprevistos respecte a la càrrega de feina.
- El major problema trobat ha estat la gestió de factors externs que han impactat a la dedicació planificada. En tenir una data d'entrega tancada i no negociable, l'única opció viable es fer un sobre esforç per a finalitzar els sprints planificats, donat que en aquest cas, no es plantejable tampoc ampliar l'equip.

Per a finalitzar, encara que la hipòtesi plantejada al cas pràctic està basada en la meva experiència personal amb equips distribuïts i aplicant Scrum, té mancances que són difícils de valorar. El factor humà es imprescindible per a aquesta metodologia i com ja s'ha comentat la maduresa de l'equip també.

Quan parlem d'equip, pensem en els desenvolupadors, en les persones de perfil tècnic, però l'equip són tots els integrants. Qui no desenvolupa,

per exemple el product owner o cap de projecte, també ha d'estar d'acord amb la metodologia i ser conscient del rigorós que s'ha de ser per a l'aplicació correcta del procés.

De la mateixa manera, el client ha de ser particip en tot el procés, conèixer la metodologia de la mateixa manera que l'equip i tenir el mateix nivell de consciència respecte de l'aplicació.

En el moment en que alguna de les parts integrants de l'equip perdi la rigurositat o deixi de prioritzar el procés per sobre d'altres factors, l'aplicació de la metodologia serà un fracàs i afectarà al projecte.

En els moments actuals és difícil trobar clients que estiguin disposats a aplicar aquesta metodologia al 100%.

Crec que el primer pas ha de ser un canvi de mentalitat per part de tots els agents que intervenen i gestionen els projectes i a partir d'aquí podrà ser plantejable l'aplicació de metodologies àgils de manera generalitzada. Fins a que no arribi aquest moment, serà difícil aplicar-ho de manera correcta.

6. Glossari

Burndown Chart – Gràfica que mostra el grau d'avanç o bé de l'sprint en curs o bé de tot el backlog del producte a desenvolupar. Eina que serveix per a mesurar el grau d'avanç i fer el seguiment i control del projecte.

Daily Meeting – Reunió diària de seguiment del sprint en curs que manté l'equip de projecte a on cada membre de l'equip respon a tres preguntes bàsiques: Què vas fer ahir? Què faràs avui? Quins impediments t'has trobat que no t'han permès assolir el teu objectiu?

Equip de desenvolupament – Equip encarregat del desenvolupament de les tasques del projecte planificades a un sprint.

Historia d'usuari – Fitxa a on es recull el requeriment a desenvolupar i els passos per a validar que el desenvolupament és correcte.

Iteració o desenvolupament iteratiu – Període de duració que té un sprint. Desenvolupament plantejat en blocs de curts períodes de temps anomenats sprints dintre de la metodologia Scrum. Cada sprint es pot considerar una iteració del cicle de desenvolupament.

Kanban - metodologia de desenvolupament de software enfocat a la filosofia de gestió de projectes àgils. Similar al task board d'Scrum, però amb la principal diferència que té una limitació respecte al nombre de tasques que poden estar en curs a la vegada. Enfoc més adequat per a la gestió de serveis.

Metodologies àgils – Marcs de treball enfocats a la filosofia àgil, basada en aquests quatre punts: Individus 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 de cenyir-se a una planificació.

Planificació de pòquer – Eina que es fa servir per part de l'equip per a estimar les tasques d'un sprint. Rep el seu nom de pòquer de les cartes que es fan servir per a donar les estimacions de les tasques, valorades amb un nombre i del fet que aplica un toc de gamificació al procés d'estimació.

Product backlog – Llistat del requeriments del producte a desenvolupar. Conté tant requeriments funcionals definits pel client, com requeriments de l'equip necessaris per a portar a terme el projecte (tasques tècniques, de configuració, etc).

Product Owner – Propietari del producte. Normalment és una figura representada pel client o pel contacte de l'equip amb el client.

Punts d'història – Mesura de magnitud d'una història d'usuari a on la seva estimació es relativa a la estimació d'altres tasques.

Scrum - Marc de treball para gestionar projectes segons la filosofia de gestió àgil de projecte. Metodologia complerta per a la gestió de projectes àgils.

Scrum master – Facilitador del procés d'Scrum. Responsable de vetllar pel bé del procés i de fer complir la metodologia.

Sprint - Cicle de treball aplicat a la metodologia Scrum. Període de desenvolupament (entre una i quatre setmanes de duració) que es repeteix iterativament fins a la finalització del projecte.

Sprint backlog - Llistat del requeriments del producte a desenvolupar durant l'sprint en curs. L'sprint backlog és un subconjunt del product backlog i es defineix a les reunions d'sprint Planning.

Sprint Planning – Reunió dintre de la metodologia Scrum a on l'equip defineix quines tasques es treballaran a la propera iteració (en base a la prioritat definida pel product owner) i es fa una valoració amb tot l'equip de cada una de les tasques incloses.

Sprint retrospective – reunió dintre de la metodologia Scrum a on l'equip de desenvolupament intercanvia impressions respecte de com a anat l'sprint finalitzat i quines àrees de millora es poden aplicar per als propers sprints.

Sprint Review – reunió dintre de la metodologia Scrum a es presenten les tasques finalitzades a l'sprint i es fa una demo d'aquestes mostrant el producte real desenvolupat.

Stand up Meeting – Sinònim de Daily Meeting. Reunió diària de seguiment que porta a terme l'equip de projecte.

Task Board – Tauler visual a on es recullen totes les tasques en les que l'equip està treballant durant un sprint. Recull les tasques, l'assignació dels membres de l'equip i l'estat en el que es troba cada tasca.

Trello – Eina que permet gestionar un task board dinàmic d'escum de manera digital. <https://trello.com/>

Xtreme programming – metodologia de desenvolupament de software enfocat a la filosofia de gestió de projectes àgils.

7. Bibliografia

Novembre 2015 - Manifiesto for Agile Software Development

<http://www.agilemanifesto.org/>

Novembre 2015 - Microsoft Developer Network

Article: *"Principios y valores de Agile, por Jeff Sutherland"*

[https://msdn.microsoft.com/es-es/library/dd997578\(v=vs.120\).aspx](https://msdn.microsoft.com/es-es/library/dd997578(v=vs.120).aspx)

Novembre 2015 – Northware software development

<http://www.northware.mx/desarrollo-en-cascada-waterfall-vs-desarrollo-agile-Scrum/>

Novembre 2015 – Scrum

<https://es.wikipedia.org/wiki/Scrum>

Novembre 2015 – Framework Scrum

<http://xsfera.com/>

Desembre 2015 – Mountaing goat software

<https://www.mountaingoatsoftware.com/agile/Scrum/sprint-backlog>

Desembre 2015 – Scrum Institute

http://www.Scrum-institute.org/Burndown_Chart.php

Desembre 2015 – Agile software development

<http://agilesoftwaredevelopment.com/files/apostimages/Scrum/simple-sprint-backlog.png>

Desembre 2015 – Eclipse.org

http://epf.eclipse.org/wikis/Scrum/Scrum/workproducts/taskboard_2DC4A0DF.html

Documentació i llibres de consulta:

- Scrum: El nuevo y revolucionario modelo organizativo que cambiará tu vida de Jeff Sutherland
- Gestión práctica de proyectos con Scrum: Desarrollo de software ágil para el Scrum Master de Antonio Martel
- Scrum y XP desde las trincheras de Henrik Kniberg
- Documentació entregada al curs "Gestió de projectes àgils amb Scrum" impartir per Marc Florit (Agile Coach)

8. Annexos

No s'inclou cap annex.