



# SCRUM

2010 Február

*Scrum: Fejlesztí és karbantartja Ken Schwaber és Jeff Sutherland*

# Köszönetnyilvánítás

## Bevezető

A Scrum az ipar által elfogadott bevett, évtizedek óta használt és bizonyított módszereken alapszik. Ezek alapján lett egy tapasztalati folyamat elméletbe összefoglalva. Ahogy Jim Coplien egyszer Jeff-nek említette: „A Scrum mindenkinek tetszeni fog; ez az amit mindannyian csinálunk amikor falhoz állítanak.”

## Emberek

Több ezer ember közül akik hozzájárultak a Scrumhoz, ki kell emelnünk azokat akik meghatározóak voltak az első tíz évben. Elsőként ott van Jeff Sutherland aki Jeff McKenna-val dolgozik és Ken Schwaber Mike Smith-vel és Chris Martin. A Scrum hivatalosan először az 1995-ös OOPSLA-n lett bemutatva. A következő öt évben Mike Beedie és Martine Devos tett hozzá sokat. Valamint sokan mások, akiknek a segítségével a Scrum nem lenne olyan letisztult mint ma.

## Történelem

A Scrum története már hosszúnak tekinthető a szoftverfejlesztés világában. Hogy megtiszteljük az első helyeket ahol kipróbálták és finomították, kiemeljük az Individual, Inc., Fidelity Investments és IDX (napjainkban GE medical) vállalatokat.

## Fordítás

Ez a kézikönyv a Ken Schwaber és Jeff Sutherland által írt *Scrum Guide* eredeti angol változatának fordítása. A magyar nyelvű fordítást Péntek Gábor készítette, a korrektúrát Radics László és Péntek Bálint végezte.

## Cél

A Scrum-ot az 1990-es évek elejétől használják komplex termékek fejlesztéséhez. Ez az írás bemutatja, hogy hogyan használjuk a Scrum-ot termékfejlesztéshez. A Scrum se nem folyamat, se nem technika a termékek fejlesztéséhez; sokkal inkább egy keretrendszer, melyen belül különböző folyamatokat és technikákat lehet alkalmazni. A Scrum szerepe az, hogy felszínre hozza a fejlesztési gyakorlatok relatív hatékonyságát annak érdekében, hogy javíthassunk rajta, **miközben egy olyan keretrendszert nyújt aminek a keretein belül összetett termékeket lehet fejleszteni.**

## Scrum elmélet

A Scrum, egy tapasztalati folyamatszabályozási elméleten alapul, ami inkrementális megközelítést alkalmaz a becslések optimalizálhatóságára és a

kockázatok kezelésére. A tapasztalati folyamatszabályozás mindegyik megvalósítása három pilléren nyugszik.

### Az első pillér az átláthatóság

Az átláthatóság biztosítja, hogy a folyamat azon részei melyek hatással vannak a végtermékre, láthatóak azok számára akik menedzselik a végterméket. Nem csak átláthatónak kell lenni ezeknek a részeknek, hanem amit látunk azt meg is kell tudni érteni. Ez azt jelenti, hogy amikor valaki egy folyamat vizsgálatakor azt hiszi, hogy valami *elkészült*, akkor ez egybeessen a *kész* általuk használt definíciójával.

### A második pillér a vizsgálat

A folyamat különböző részeit kellő gyakorisággal kell vizsgálni ahhoz, hogy a nem elfogadható eltéréseket a folyamatban észrevegyük. A vizsgálat gyakoriságánál figyelembe kell venni, hogy a vizsgálat minden folyamatra hatással van. Fura dolgok történnek, ha a vizsgálat szükséges gyakorisága meghaladja a folyamat vizsgálatokkal szembeni toleranciáját. Szerencsére úgy tűnik, ez nem jellemző a szoftver fejlesztésre. A gyakoriságon kívül a másik tényező a végeredmény vizsgálatát végző emberek felkészültsége és szorgalma.

### A harmadik pillér az alkalmazkodás

Ha a vizsgáló a vizsgálat eredményeképpen arra az következtetésre jut, hogy egy vagy több tekintetben a folyamat az elfogadható határokon kívül van – és az eredményül kapott termék nem lesz elfogadható – a vizsgálónak módosítania kell a folyamatot, vagy az éppen feldolgozás alatt lévő anyagot. A módosítást olyan gyorsan kell elvégezni, amilyen gyorsan csak lehet, hogy elkerüljük a további eltéréseket.

A Scrum-ban három pont van a vizsgálatra és alkalmazkodásra. A Napi Scrum megbeszélésen (Daily Scrum) megvizsgáljuk, hogy mennyit haladtunk a Sprint cél felé és ennek megfelelően változtatunk, hogy optimalizáljuk a következő munkanapot. A Sprint Áttekintő (Sprint Review) és Sprint Tervező (Sprint Planning) megbeszélések azt a célt szolgálják, hogy megvizsgáljuk a Release Cél (Release Goal) felé haladást és módosításokat eszközöljünk mellyel a következő Sprint eredményességét javíthatjuk. Végül a Sprint Visszatekintő (Sprint Retrospective) megbeszélés megvizsgálja az elmúlt Sprintet és meghatározza, milyen változások tehetik a következő sprintet produktívabbá, kielégítőbbé és szórakoztatóbbá.

### Scrum tartalom

A Scrum keretrendszer **Scrum csapatok** halmazából és hozzájuk rendelt szerepkörökből áll; **Idő-Keretekből**, **Termékekből** és **Szabályokból**

A Scrum Csapatot úgy alakítjuk ki, hogy optimalizáljuk a rugalmasságot és a termelékenységet. Ennek érdekében önszerveződők, polihisztorok és iterációkban dolgoznak. Minden Scrum Csapatban három szerepkör van: 1) a **ScrumMester** (ScrumMaster) aki azért felel, hogy a folyamatot mindenki megértse és betartsa; 2) a **Termék Tulajdonos** (Product Owner) akinek a feladata, hogy maximalizálja a Scrum Csapat által végzett munka értékét; és 3) a **Csapat** aki elvégzi a munkát. A csapat olyan fejlesztőkből áll, akiknek minden tudása megvan ahhoz, hogy a Termék Tulajdonos elvárásai a Sprint végére akár kiadható része legyen a terméknek.

A Scrum időkereteket alkalmaz a rendszeresség megvalósításához. A következő elemek vannak időkerethez kötve: a Kiadástervező (Release Planning) megbeszélés, a Sprinttervező (Sprint planning) megbeszélés, a Sprint, a Napi Scrum, a Sprint Áttekintő (Sprint Review) és a Sprint Visszatekintő (Sprint Retrospective). A Scrum lelke a Sprint, ami egy egy hónapos vagy rövidebb iteráció, ami adott fejlesztés alatt állandó hosszúságú. Az összes Sprint a Scrum keretrendszer használja és minden Sprint egy potenciálisan kiadható fejlesztését eredményezi a végleges terméknek (inkremenst). Az egyik Sprint rögtön a másik után kezdődik.

A Scrum négy vezérfonalat használ. A **Terméknapló** (Product Backlog) egy prioritizált lista azokról a dolgokról, amik a termékbe kerülhetnek. A **Sprintnapló** (Sprint Backlog) olyan feladatok listája amely a Terméknaplót egy Sprint erejéig egy szállítható termék inkremenst eredményező listává alakítja. A **Kiadásbontás** (Release Burndown) a Terméknaplóból hátralévő feladatokat méri egy kiadási terv alatt. A **Sprintbontás** (Sprint Burndown) a Sprintnaplóból hátralévő feladatokat méri egy Sprint alatt.

A Scrum időkereteit, szerepköreit és segédelemeit **szabályok** kapcsolják össze. Ezeket a szabályokat fejtjük ki ebben a dokumentumban. Például egy Scrum szabály, hogy csak Csapattagok – olyan emberek akik elkötelezettek abban, hogy a Terméknaplóból egy inkremenst hozzanak létre – beszélhetnek a Napi Scrum megbeszélések alatt. Az olyan javaslatok a Scrum megvalósításához, melyek nem szabályok, Tipp dobozokban szerepelnek.

## Scrum Szabályok

### Tipp

*Amikor a szabályok nincsenek lefektetve, a Scrum felhasználóinak kell kitalálni, hogy mit tegyenek. Ne próbáljuk tökéletes megoldást kitalálni, mert a probléma általában gyorsan változik. Inkább próbáljuk ki azt ami eszünkbe jut és nézzük meg, hogy működik? A Scrum elemző-alkalmazkodó tulajdonsága majd segíteni fog.*

A Scrum Csapat egy ScrumMesterből, egy Termék Tulajdonosból és a Csapatból áll. A Scrum Csapat tagjait „malacoknak” hívják. A Termék Tulajdonos a Terméknapló „malaca”. A Csapat a Sprint munka „malaca”. A ScrumMester a Scrum folyamat „malaca”. Mindenki más „csirke”. A csirkék és malacok a következő történetből jönnek:

*„A csirke és a malac együtt mendegélnek, amikor a csirke azt mondja, „Nyissunk egy éttermet!”. A malac elgondolkodik a dolgon majd megkérdezi, „Rendben, de hogy nevezzük el?”. A csirke erre azt válaszolja, „Hívjuk Ham&Eggs –nek! (Sonka és tojás)”. A malac erre azt válaszolja, „Nem, kösz! Én el lennék kötelezve, de te csak be lennél vonva!”*

## A ScrumMester

A ScrumMester felelős azért, hogy a Scrum Csapat pontosan betartsa a Scrum értékeket, praktikákat és szabályokat. A ScrumMester segíti a Scrum Csapatot és a szervezetet a Scrum alkalmazásában. A ScrumMester tanítja a Scrum Csapatot úgy, hogy felügyeli és vezeti azt annak érdekében, hogy produktívabb legyen és magasabb minőségű termékeket állítson elő. A ScrumMester segíti a csapatot az önszerveződés és a keresztfunkcionalitás alkalmazásában. A ScrumMester ezen kívül segíti a csapatot olyan szervezeti környezetben ami még nem készült fel komplex termékek fejlesztésére. Amikor a ScrumMester elősegíti ezeket a változásokat, azt úgy nevezzük, hogy az „akadályok eltávolítása”. A ScrumMester szerepe igazából egy szolgáló-vezető a Scrum Csapat számára.

## A Termék Tulajdonos

A Termék Tulajdonos az egyetlen személy, aki felelős a Terméknapló kezeléséért, valamint a Csapat által végzett munka értékének biztosításáért. Ez az illető tartja karban a Terméknaplót és biztosítja, hogy ez mindenki számára látható legyen. Mindenki tudja, minek van a legnagyobb prioritása, tehát mindenki tudja, miken kell dolgozni.

### Tipp

*A ScrumMester a megrendelővel és a menedzsmenttel dolgozik együtt azon, hogy meghatározza és kijelölje a Termék Tulajdonost. A ScrumMester feladata a Termék Tulajdonost megtanítani a teendőire. A Termék Tulajdonostól elvárjuk, hogy tudja hogyan kell értéket optimalizálni a Scrum keretein belül. Ha ezt nem tudja, azért a ScrumMester a felelős.*

### Tipp

*A ScrumMester lehet a Csoport egy tagja; például egy fejlesztő aki Sprint feladatokat hajt végre. Viszont ez gyakran konfliktushoz vezet, mikor a ScrumMesternek döntenie kell, hogy az akadályokat hárítsa el, vagy a feladataival haladjon inkább. A ScrumMesternek soha nem szabad Termék Tulajdonosnak lenni.*

A Termék Tulajdonos egyetlen ember, nem egy bizottság. Létezhetnek olyan bizottságok, amelyek tanácsot adnak, illetve hatással vannak erre az illetőre, de ha valaki meg szeretné változtatni egy elem prioritását, akkor meg kell győznie erről a Termék Tulajdonost. A Scrumot bevezető vállalatok gyakran úgy találják, hogy ez idővel befolyásolja a prioritálási és követelmény felvételi módszereiket.

A Termék Tulajdonos sikeréhez szükséges, hogy a szervezeten belül mindenki tiszteletben tartsa a döntéseit. Senki nem mondhatja a Csapatnak, hogy más prioritások szerint dolgozzon és a Csapat nem hallgathat senki másra. A Termék Tulajdonos döntéseit a Terméknapló tartalma és prioritási sorrendje mutatja. Ez az átláthatóság arra ösztönzi a Termék Tulajdonost, hogy legjobb tudása szerint járjon el és magát a szerepkört egyszerre teszi megerőltetővé és hálássá.

## A Csapat

Fejlesztők csapatai készítik a funkciók potenciálisan leszállítható növekményét (inkremenst) minden Sprintben. A csapatok ezen kívül keresztfunkcionálisak; Minden csapattagnak megvan a képessége, hogy előállítson egy növekményt. A csapattagoknak gyakran van speciális szakértelme egy adott területen, mint pl. programozás, minőségbiztosítás, üzleti analízis, tervezés, felhasználói felület tervezés vagy adatbázis tervezés. Ellenben az a tudás amin a csapattagok osztoznak – nevezetesen a készség, hogy egy követelményt megfogjanak és egy használható inkremensté változtassanak – fontosabb, mint az amiben nem. Az olyan emberek akik nem hajlandók kódolni mert ők tervezők vagy dizájnerek, nem passzolnak megfelelően a Csapatba. Mindenki hozzájárul valamivel még akkor is, ha ehhez új ismereteket kell elsajátítani, vagy régiakat feleleveníteni. A Csapaton belül nincsenek titulusok és ez alól nincs kivétel. A Csapatok nem tartalmaznak adott szakterületre specializálódott – pl. tesztelés, vagy üzleti analízis – al-Csapatokat sem.

A Csapatok ezen kívül önszerveződők. Senki – még a ScrumMaster se – mondja meg a csapatnak, hogy hogyan készítsen a Terméknaplóból egy szállítható funkció inkremenst. Minden csapattag hozzáteszi a tudását az összes probléma megoldásához. Az ennek eredményeképpen létrejövő szinergia fejleszti az egész

### Tipp

*A Termék Tulajdonos lehet Csapattag, aki fejlesztési tevékenységet is végez. Ez a plusz felelősség viszont beleszólhat a Termék Tulajdonos felelősökhöz fűződő viszonyába. A Termék Tulajdonos viszont soha nem lehet a ScrumMester.*

### Tipp

*Szerződéses fejlesztésnél a Termék Tulajdonos lehet a termékmenedzser. Belső fejlesztéseknél a Termék Tulajdonos lehet az éppen automatizálás alatt lévő üzleti folyamat a menedzsere.*

Csapat eredményességét és hatékonyságát.

Egy Csapat optimális mérete hét ember, plusz mínusz kettő. Ha ötnél kevesebb csapattag van egy csapatban akkor kevesebb az interakció és így kevesebb a termelékenységi nyereség. Mi több, a Csapat szakértelmi korlátokba ütközhet a Sprint folyamán és nem lesz képes elkészíteni egy leszállítható terméket a Sprint végére. Ha kilenc embernél többen vannak, akkor egyszerűen túl sok koordináció szükséges. A nagy csapatok túl nagy komplexitást képeznek egy tapasztaláson alapuló folyamatban ahhoz, hogy az kezelhető legyen. Ennek ellenére találkoztunk már olyan sikeres Csapatokkal, melyek túlléptek ezeken a korlátokon. A Termék Tulajdonos és a ScrumMester nem számít bele ebbe a keretbe, hacsak nem ők is a Sprintnaplóban található feladatokon dolgozó „malacok”.

A Csapat szervezete a Sprintek végén változhat. Minden esetben amikor a Csapat tagsága megváltozik az önszerveződésből fakadó termelékenység csökken. Éppen ezért körültekintőnek kell lenni amikor változtatunk egy csapaton.

## Időkeretek

A Scrum időkeretei a **Kiadás Tervező Megbeszélés**, a **Sprint**, a **Sprint Tervező Megbeszélés**, a **Sprint Áttekintő**, a **Sprint Visszatekintő** és a **Napi Scrum**.

## Kiadás Tervező Megbeszélés

A Kiadás Tervező Megbeszélés célja egy olyan terv és célok felállítása amely a Scrum Csapat és a szervezet többi részének is érthető és kommunikálható. A Kiadás Tervező Megbeszélés a következő kérdésekre ad választ: „Hogyan fordíthatjuk az elképzelést egy nyerő terméké a lehető legjobb, lehetséges módon? Hogyan érhetjük el vagy haladhatjuk meg a kitűzött megrendelői elégedettséget illetve megtérülést (Return on Investment, RoI)?” A kiadási terv határozza meg a kiadás célját, a legmagasabb prioritású Terméknapló elemeket, a fő kockázatokat és az összes funkciót és sajátosságot amit a kiadás tartalmazni fog. Szintén tartalmaz egy valószínűsíthető szállítási időpontot és egy időkeretet, amik tarthatók ha semmi nem változik. Így a szervezet vizsgálhatja a haladást és módosíthatja a kiadási tervet Sprintenkénti alapon.

A kiadás tervezés egyáltalán nem kötelező. Ha a Scrum Csapat ennek megbeszélése nélkül áll neki a munkának, az eredmények hiánya nyilvánvalóan akadályként jelentkezik majd, amit el kell hárítani. Az akadályt eltávolító feladat egy bejegyzés lesz a Terméknaplóban.

A Scrumban a termék iteratívan fejlődik. Minden Sprint elkészíti egy teljesebb növekményét a terméknek, a legértékesebb és legkockázatosabb résszel kezdve. A következő sprintek mindig egy teljesebb inkremenst hoznak létre a termékből.

Minden növekmény egy akár leszállítható változata a végleges terméknek. Ha már elég inkremens készült ahhoz, hogy a termék a finanszírozó számára használható értéket képviseljen, a termék kiadásra kerül.

A legtöbb szervezet jelenleg is rendelkezik egy kiadástervezési folyamattal és a legtöbb ilyen folyamatban a tervezés nagy részét a folyamat elején végzik és a későbbiekben nem változtatnak rajta. A Scrum kiadástervezés során egy átfogó cél és egy valószínűsíthető végeredmény kerül meghatározásra. Ez a kiadástervezés általában egy hagyományos kiadási terv elkészítési idejének 15-20%-ánál nem igényel többet. Viszont a Scrum minden Sprint Áttekintéskor és Sprint Tervezéskor végrehajt bizonyos kiadás tervezést csakúgy, mint a Napi Scrum alkalmával. Összességében tehát a Scrum lényegesen több erőfeszítést tesz a kiadás tervezés érdekében, mint a hagyományos kiadás tervezési folyamatok.

A kiadástervezés teszi szükségessé a Terméknapló tételeinek becslését és prioritizálását. Sokféle technika van ennek elvégzésére amik nem tartoznak szorosan a Scrum bemutatásához, de ettől függetlenül hasznosak, ha azzal együtt alkalmazzuk.

## A Sprint

A sprint egy iteráció. A Sprintek időben korlátozottak. A Sprint alatt a ScrumMester biztosítja, hogy semmi olyan változtatás nem történik, ami befolyásolná a Sprint Cél. A Csapat összetétele és a minőségi célok is változatlanok egy sprint alatt. A Sprintek a Sprint Tervező Megbeszélésből, a fejlesztő munkából, a Sprint Áttekintőből és a Sprint Visszatekintőből állnak. A Sprintek közvetlenül egymás után következnek, köztük lévő szünetek nélkül.

Egy projektet arra használunk, hogy elérjünk valamit; a szoftverfejlesztésben például, hogy felépítsünk egy terméket, vagy rendszert. Minden projekt tartalmaz egy definíciót arról, hogy mit kell elkészíteni, egy tervet ami szerint készül, a munkát, amit a terv alapján elvégeztek és végül az elkészült terméket. Minden projektnek van egy horizontja ami az az időkeret amire a terv jó. Ha a horizont túl nagy, változhat a definíció, túl sok paraméter kerülhet be, a kockázat túl

### Tipp

*Ha a Csapat úgy érzi, hogy túlterhelt, találkoznak a Termék Tulajdonossal, hogy csökkentsék a termék hatáskörét (scope). Ha a Csapat úgy érzi, hogy marad ideje, a Termék Tulajdonossal közösen kiválaszthatnak további elemeket a Terméknaplóból.*

### Tipp

*Amikor egy Csapat elkezdi a Scrum-ot, a kéthetes Sprintek lehetővé teszik a beletanulást anélkül, hogy bizonytalanságban lennének. Az ilyen hosszú Sprintekkel könnyen szinkronban lehetünk a többi Csapattal úgy, hogy két Sprint hosszát összeadjuk.*

magas lehet stb... A Scrum egy olyan projekt számára készült keretrendszer, aminek az időhorizontja nem több egy hónapnál, ahol a távolabbi horizont már túl kockázatos volna. Minden hónapban figyelni kell a projekt előreláthatóságát, továbbá azt a kockázatot, hogy a projekt kezelhetetlenné vagy bizonytalanná válik legalább havonta kezeljük.

A Sprintek lefújhatók az időkeret letelte előtt. Csak a Termék Tulajdonosnak van joga lefűjni egy sprintet, viszont ő teheti ezt a részvényesek, a Csatat vagy a ScrumMester nyomására. Milyen körülmények vezethetnek egy Sprint lefűjéséhez? A menedzsmentnek szüksége lehet a sprint lefűjésére amennyiben a Sprint céljai már elévültek. Ilyen akkor történhet ha a cég irányt változtat vagy ha a piaci vagy technológiai feltételek megváltoznak. Általánosságban egy Sprintet akkor kell lefűjni, ha adott körülmények között már nincs értelme. Azonban a Sprintek rövid hossza miatt ritkán van ennek értelme.

Ha egy Sprintet lefűjnek, minden befejezett és „kész” Terméknapló elemet ellenőrzünk. Akkor fogadjuk el őket, ha egy potenciálisan szállítható inkremenst képviselnek. Az összes többi Terméknapló elemet visszatesszük a Terméknaplóba az eredeti becslésekkel. A velük töltött munkát elveszítettnek tekintjük. A Sprint megszakítások erőforrásokat igényelnek, mivel mindenkinek újra kell szervezni a munkáját egy Sprinttervező megbeszélésen, hogy egy másik Sprint induljon. A Sprint megszakítások gyakran jelentenek traumát és nagyon szokatlanok.

### **Sprint Tervező Megbeszélés**

A Sprint Tervező Megbeszélés akkor van amikor egy iterációt tervezünk. Egy egy hónapos Sprint esetén a hossza korlátozva van 8 órára. Rövidebb sprintekhez arányosan kevesebb időt foglaljunk (pl. egy két hetes Sprinthez négy órás Sprint Tervező Megbeszélés tartozik). A Sprint Tervező Megbeszélés két részből áll. Az első részben döntjük el, hogy mikor, mit fogunk tenni a sprint során. A második részben (egy négy órás időkorláttal havi Sprint esetén) a Csatat kitalálja, hogy fogja ezt a funkcionalitást egy működő növekménnyé alakítani a Sprint során.

Két része van a Sprint Tervező Megbeszélésnek: a „Mit?” rész és a „Hogyan?” rész. Bizonyos Scrum csapatok vegyítik a kettőt. Az első részben a Scrum Csatat a „Mit?” kérdést célozza meg. Itt a Termék Tulajdonos előadja a Terméknapló legfontosabb elemeit a Csatatnak. Közösen kitalálják, hogy milyen funkcionalitást kell kifejleszteni a következő Sprint során. Ennek a megbeszélésnek az alapja a Terméknapló, az előzőleg fejlesztett termék inkremens, a Csatat terhelhetősége és a Csatat múltbéli teljesítménye. Az, hogy a terméknaplóból a Csatat mennyi tételt választ ki, csak a Csataton múlik. Csak a Csatat tudja megbecsülni, hogy mit tud megcsinálni a következő Sprint alatt.

A Terméknaplóból válogatva körvonalazódik a Sprint Cél. A Sprint Cél egy olyan cél, amihez a Terméknapló implementálásán keresztül jutunk. Ennek a meghatározása segíti a Csatatot abban, hogy megértse, miért készíti az

inkremenst. A Sprint Cél a Kiadási cél egy részhalmaza.

A Sprint Cél meghatározásának oka, hogy a Csapatnak legyen valami fogalma a funkcionalitásról. Például a fenti Sprint Célja lehetne ez is: „Automatizáljuk az ügyfélazonosítók módosítását egy biztonságos, hibatűrő köztesréteggel.”. A Csapat munka közben ezt a célt tartja szem előtt. A cél megvalósítása érdekében valósítja meg a funkciókat és a technológiát. Ha a feladat nehezebbnek bizonyul, mint a Csapat eredetileg számította, akkor a Termék Tulajdonossal együttműködve csak részlegesen implementálják a funkcionalitást.

A Sprint Tervező Megbeszélés második felében a Csapat a „Hogyan?” kérdést vizsgálja. Ebben a szakaszban (ami egy havi Sprint esetén 4 órás időkorláttal rendelkezik) a Csapat kitalálja, hogyan fogja a Sprint Tervezéskor kiválasztott Terméknaplót („Mit?”) egy kész inkremenssé alakítani. A Csapat általában azzal kezd, hogy megtervezi a munkát. A tervezés folyamán a Csapat azonosítja a feladatokat. Ezek a feladatok részletezett munkarészek, amik ahhoz kellenek, hogy a Terméknaplót működő szoftverré alakítsák. A feladatokat annyira le kell bontani, hogy egy napon belül elvégezhetőek legyenek. Ezt a feladat listát nevezzük Sprintnaplónak. A Csapat megszervezi magának a munka végrehajtását akár már a Sprint Tervező megbeszélés alatt, vagy akár a Sprint közben folyamatosan.

A megbeszélés második felében a Termék Tulajdonos jelen van azért, hogy tisztázza a Terméknapló céljait és hogy segítsen kompromisszumokra jutni. Ha a Csapat úgy ítéli meg, hogy túl sok vagy túl kevés feladata van, újratárgyalhatja a Terméknaplót a Termék Tulajdonossal. A Csapat ezen kívül meghívhat másokat is,

hogy technikai és szakmai tanácsokat adjanak. Egy új Csapat gyakran itt szembesül először azzal, hogy egy emberként vagy felszínen marad, vagy elmerül. A Csapat ekkor ismeri fel, hogy csak magára számíthat. Amint ez bekövetkezik, megkezdődik az önszerveződés és elkezd olyan tulajdonságokat és viselkedést felvenni, mint egy igazi Csapat.

Tipp

*Általában a Terméknapló csak 60-70% -a kerül feldolgozásra a Sprint Tervező Megbeszélésen. A maradék félre lesz téve későbbi részletezésre, vagy nagyvonalú becsléssel lesz ellátva és később lesz lebontva a Sprint során.*

## Sprint Áttekintő

A Sprint végén egy Sprint Áttekintő Megbeszélést (Sprint Review Meeting) tartunk. Ez egy négyórás időkorláttal rendelkező megbeszélés egy hónapos Sprintek esetén. Rövidebb Sprinteknél arányosan rövidebb a megbeszélés (például egy két hetes Sprint esetén a Sprint Áttekintő két órá). A Sprint Áttekintő Megbeszélés alatt a Scrum Csapat és a közreműködők értekeznek az éppen elvégzett dolgokról. Ez alapján és a Terméknaplóban bekövetkezett

változások alapján értekeznek arról, hogy milyen következő feladatokkal kellene foglalkozni. Ez egy informális megbeszélés az elkészült funkciók bemutatásával annak érdekében, hogy elősegítse az együttműködést a következő teendőkkel kapcsolatban.

A megbeszélés legalább a következő elemeket foglalja magába. A Termék Tulajdonos azonosítja, hogy mi készült el és mi nem készült el. A Csapat megvitatja, hogy mi ment jól az elmúlt Sprintben, milyen problémákba ütköztek és ezeket hogyan oldották meg. Ezután a Csapat bemutatja az elkészült munkát és megválaszolja a kérdéseket. A Termék Tulajdonos ezután megbeszéli a Terméknapló állását. Megbecsüli a lehetséges befejezési dátumokat különböző sebességek feltételezésével. Az egész csoport megbeszéli a látottakat és hogy ez mit jelent a következő teendőkre nézve. A Sprint Áttekintő értékes információként szolgál a soron következő Sprint Tervező Megbeszéléshez.

### **Sprint Visszatekintő**

A Sprint Áttekintő (Sprint Retrospective) után és a következő Sprint Tervező Megbeszélés előtt a Scrum Csapat tart egy Sprint Visszatekintő Megbeszélést. Ez egy három órában korlátozott megbeszélés havi Sprintek esetén (ennél arányosan kevesebb ideig tart rövidebb sprintek esetén). Ennek a megbeszélésnek a folyamán a ScrumMester arra ösztönzi a Scrum Csapatot, hogy a Scrum keretrendszer technikáival gondolja át a fejlesztési módszertanát annak érdekében, hogy hatékonyabbá és szórakoztatóbbá tegye azt a következő Sprintre. Sok könyv ír le technikákat, melyek segítenek a Visszatekintő során.

A Visszatekintő célja, hogy megvizsgálja, hogy ment a legutóbbi sprint az emberek, kapcsolatok, folyamatok és eszközök tekintetében. A vizsgálatnak azonosítani kell azokat az elemeket amik jól mentek és azokat az elemeket – amiket ha másképp csinálunk – még jobban mennek a dolgok. Ide értve a Scrum Csapat felépítését, megbeszélés egyeztetéseket, eszközöket, a „kész” definícióját, kommunikációs technikákat, és a folyamatokat amik a Terméknaplót „kész” terméké teszik. A Sprint Visszatekintő végére a Scrum Csapatnak azonosítania kell azokat a fejlődési szempontokat amiket a következő Sprint során alkalmazni fog. Ezek a változtatások a tapasztalati elemzés részévé válnak.

### **Napi Scrum**

Minden Csapat minden nap találkozik egy 15 perces elemző és reagáló megbeszélésre amit Napi Scrum-nak hívunk. A napi Scrum a sprint során mindig ugyanabban az időben és ugyanazon a helyen van. A megbeszélés során az összes csapattag elmondja a következőket:

1. Mit csinált a legutóbbi megbeszélés óta?

2. Mit fog csinálni a következő megbeszélésig?
3. Milyen akadályok állnak az útjába?

A Napi Scrum megbeszélés elősegíti a kommunikációt, feleslegessé tesz további megbeszéléseket, azonosítja és eltávolítja a fejlesztés útjába kerülő gátakat, kiemeli és elősegíti a gyors döntéshozatalt és mindenkinek emeli a projekttel kapcsolatos ismereteinek szintjét.

A ScrumMesternek kell felügyelni, hogy a Csapat megtartsa a megbeszélést. A Napi Scrum levezetéséért a Csapat a felelős. A ScrumMester megtanítja a Csapatnak, hogy hogyan legyen rövid a Napi Scrum, betartatja a szabályokat és arra ösztönzi az embereket, hogy tömören beszéljenek. A ScrumMester azt a szabályt is betartatja, hogy a csirkék nem beszélhetnek illetve semmilyen módon nem zavarhatják meg a Napi Scrum-ot.

A Napi Scrum nem egy státuszmegbeszélés. Senkire sem tartozik azokon az embereken kívül, akik a Terméknaplóból egy termék inkremenst készítenek (A Csapat). A Csapat elkötelezett a Sprint Cél és a hozzá kapcsolódó Terméknapló elemei iránt. A Napi Scrum a Sprint Cél felé haladásra ad rálátást (a három kérdés). Az ezt követő megbeszélések rendszerint csak azért vannak, hogy felkészüljünk a Sprint közelgő feladataira. A cél annak a valószínűségnek a növelése, hogy a Csapat teljesíti a tervet. A napi Scrum egy kulcsfontosságú elemző és alkalmazó megbeszélés a tapasztalat alapú Scrum folyamatban.

### Scrum hozzávalók

A Scrum hozzávalói között a Terméknapló, a Kiadásbontás, a Sprintnapló és a Sprintbontás szerepel.

### Terméknapló és Kiadásbontás

A Csapat(ok) által fejlesztett termék követelményeit a Terméknapló tartalmazza. A Termék Tulajdonos felelős a Terméknapló tartalmáért, elérhetőségéért

#### Tipp

*A Terméknapló elemek általában Felhasználói Leírások (User Story). A Használati Esetek (Use Case) is megfelelők, de azokat inkább élet- vagy üzemkritikus rendszerek fejlesztésénél használjuk.*

#### Tipp

*A Scrum Csapatok gyakran idejük 10%-át a Terméknapló előkészítésével (grooming) töltik, hogy az megfeleljen a fenti definíciónak. Amikor ilyen részletesen rendbetettük, a Terméknapló tetején lévő elemek (legmagasabb prioritás, legnagyobb érték) olyannyira le vannak bontva, hogy beférjenek egy Sprintbe. Az átfésülési folyamat alatt megvizsgáljuk és végiggondoljuk ezeket. Amikor eljön e Sprint Tervező Megbeszélés, ezek a Napló elemek már közérthetőek és könnyen kiválaszthatók.*

és elemeinek prioritálásáért. A Terméknapló sosem teljes. Az első verzió kidolgozásánál csak a kezdetben ismert, legjobban megértett követelményeket gyűjti össze. A Terméknapló úgy fejlődik, ahogy a termék és a környezet - amiben használni fogják - fejlődik. A napló dinamikus abban az értelemben, hogy folyamatosan változik annak érdekében, hogy azonosítsa, mire van szüksége a terméknek ahhoz, hogy megfelelő, versenyképes és hasznos legyen. Ameddig egy termék létezik, Terméknapló is létezik.

A terméknapló képvisel mindent, ami ahhoz szükséges, hogy kifejlesszünk és elindítsunk egy sikeres terméket. Ez azon feladatok, sajátosságok, funkciók, technológiák, fejlesztések és hibajavítások listája amikből összeállnak a következő kiadásig elvégzendő változások. A Terméknapló elemeinek van leírása, prioritása és becslése. A prioritást a kockázat, érték és a szükségesség határozza meg. Sok technika létezik ezeknek a tulajdonságoknak a meghatározására.

A Terméknapló prioritás szerint van tárolva. A legnagyobb prioritású Termék Napló bejegyzésekből következnek az azonnali fejlesztési teendők. Minél magasabb a prioritás, annál sürgősebb, annál többet gondolkoztunk róla és annál inkább konszenzus van az általa képviselt értékről. A magasabb prioritású bejegyzések világosabbak és részletesebb információkat tartalmaznak, mint az alacsonyabb prioritásúak. A részletesebb és világosabb információkból jobb becslések születnek. Minél alacsonyabb a prioritása egy elemnek, annál kevésbé részletesen van kidolgozva és ez így is marad egészen addig, amíg előrébb nem kerül a Terméknaplóban és nem lesz idő részletesen kidolgozni.

Amint a terméket használatba veszik, az értéke megnövekszik és ahogy a piacról érkeznek a visszajelzések a Terméknapló megnő egy hosszabb és több mindenre kiterjedő listává. A követelmények változása soha nem áll meg. A Terméknapló egy élő dokumentum. Az üzleti követelményekben, piaci viszonyokban, technológiában, személyi

#### Tipp

*Az Elfogadási Tesztek (Acceptance Test) is gyakran egy tulajdonságai a Terméknapló elemeknek. Gyakran helyettesíthetik a részletes szöveges leírást egy tesztleírással, hogy mit kell a Terméknapló tételnek teljesíteni, ha elkészült.*

#### Tipp

*Néhány szervezetben több munkát vesznek fel a Naplóba, mint amennyit elvégeznek. Ez egy lapos, vagy enyhén emelkedő trend görbét eredményezhet. Ennek kompenzációjaképpen illetve, hogy az átláthatóságot megőrizzük, létrehozhatunk egy új viszonyítási alapot amikor munkát adunk hozzá, vagy veszünk el a Naplóból. Az alap csak érezhető változásokat adhat hozzá, vagy vehet el és jól dokumentálnak kell lennie.*

állományban bekövetkezett változások változást eredményeznek a Terméknaplóban. Hogy elkerüljük az átdolgozást, mindig csak a legmagasabb prioritással rendelkező elemeket kell kidolgozni részletesen. Azok a bejegyzések, amelyekkel a Csapatok elfoglalják magukat a következő Sprintek alatt, részletesen ki vannak dolgozva és le vannak bontva annyira, hogy minden egyes elem elkészíthető legyen egy sprint alatt.

Gyakran több Scrum Csapat dolgozik együtt ugyanazon a terméken. Ilyen esetben egy terméknaplót használunk a Terméken végzendő jövőbeni munka leírására. Ezután egy olyan Terméknapló elem tulajdonságot alkalmazunk, ami csoportosítja az elemeket. A csoportosítást végezhetjük funkció csoport, technológia, architektúra szerint és gyakran ez a módja, hogy Scrum Csapatonként szervezzük a munkát.

A Kiadásbontás diagram feljegyzi a fennmaradó Terméknapló becsült ráfordításait az idő függvényében. A becsült ráfordítás tetszőleges egységben lehet amiről a Scrum Csapat és a szervezet előzetesen megegyezett. Az időegységek általában Sprintek.

A Terméknapló tételeinek a becslései kezdetben a Kiadás Tervezés során készülnek, majd később ahogy létrehozzuk azokat. A Terméknapló átfésülése során ezeket felülvizsgáljuk és korrigáljuk, habár bármikor módosíthatók. A Csapat felelős az összes becslésért. A Termék Tulajdonos befolyásolhatja a Csapatot úgy, hogy segíti a kompromisszumok megkötését és megértését, de a végső becslést a Csapat mondja ki. A Termék Tulajdonos tartja napra készen a Terméknapló listát és a Kiadásnapló Bontását az érintettek számára elérhetően. A fennmaradó munkában bekövetkező változások alapján felrajzolható egy tendencia görbe.

Tipp

*A trend görbe megbízhatatlan lehet a Kiadás első két-három Sprintjében, hacsak nem a Csapat dolgozott már korábban együtt, ismeri jól a terméket és a használt technológiát.*

## Sprintnapló és Sprintbontás

A Sprintnapló (Sprint Backlog) azokat a feladatokat tartalmazza amiket a Csapat elvégez annak érdekében, hogy Terméknapló elemeket egy „kész” inkremenssé változtassa. Sok ezek közül a Sprint Tervező Megbeszélésen születik. A Sprintnapló azon munka összessége amit a Csapat szükségesnek tart a Sprint cél elérése érdekében. A Sprintnapló elemeket le kell bontani. A lebontás akkor elégséges, ha a fejlesztési lépcsők megérthetők a Napi Scrum alkalmával. Egy végrehajtható Sprintnapló tétel hossza jellemzően egy nap, vagy kevesebb.

A Csapat az egész Sprinten keresztül módosíthatja a Sprintnaplót, ahogy új Sprintnapló tételek bukkannak fel a Sprint során. Az egyéni feladatokra bontáskor kiderülhet, hogy több vagy kevesebb feladat szükséges vagy, hogy az

adott feladat több vagy kevesebb időt vesz igénybe, mint ahogy azt eredetileg becsültük. Ha új teendőre van szükség, a Csapat felveszi a Sprintnaplóba. Ahogy a feladatok haladnak, illetve elkészülnek, frissítjük a fennmaradó teendők becsléseit. Ha bizonyos feladatokat szükségtelennek találunk, töröljük őket. Csak a Csapat változtathat a Sprintnaplóján a Sprint alatt. Csak a Csapat változtathatja meg a tartalmat vagy a becsléseket. A Sprintnapló egy jól látható valós képe a munkának amit a Csapat a Sprint alatt el szeretne végezni és kizárólag a Csapatra tartozik.

A Sprintnapló bontás egy diagram ami a Sprintnaplóban lévő teendők mennyiségét mutatja a Sprint során eltelt idő függvényében. Ennek a diagramnak az elkészítéséhez a Sprint minden napján meg kell határozni, hogy mennyi a fennmaradó teendők becsléseinek összege. A Sprintben elvégzendő munka mennyisége megegyezik a Sprintnaplóban található tételek elvégzéséhez szükséges munka összegével. Kövessük nyomon ezt az összeget nap mint nap és készítsünk belőle grafikont ami időben mutatja a fennmaradó munka mennyiségét. Ha a pontokon keresztül egy vonalat húzunk, a Csapat látja és kézben tartja a haladást a Sprint teljesítése felé. Az időtartamot a Scrum nem veszi figyelembe. A fennmaradó munka és a dátum az egyetlen érdekes paraméterek.

#### Tipp

*Amikor csak lehetséges kézzel rajzoljuk a Sprint Bontás diagramot egy nagy papírlapra ami a Csapat munkahelyén van elhelyezve. A Csapatok szívesebben nézegetnek egy nagy, látható ábrát, mint egy Excel diagramot, vagy valami erre való eszközt.*

A Scrum szabályainak egyike a Sprint céljához tartozik mégpedig, hogy növekményeket állítsunk elő potenciálisan szállítható funkciókból amelyek megfelelnek a „kész” fogalomnak.

## Kész

A Scrum elvárja a Csapatoktól, hogy minden Sprintben elkészítsék a termék funkcióinak egy inkremensét. Ennek az inkremensnek akár szállíthatónak is kell lennie, a Termék Tulajdonos akár azonnal kérheti a funkció implementálását. Ennek érdekében az inkremensnek a termék egy teljes szeletének kell lennie. „Készen” kell lennie. Minden inkremenst hozzá kell tudni adni az előző inkremenshez és alaposan le kell tesztelni, hogy meggyőződjünk róla, hogy az inkremensek együttműködnek.

A termékfejlesztés során azt feltételezve, hogy egy funkció kész, valakiben azt a benyomást keltheti, hogy minimum letisztult a kód, át van nézve, tesztelve van (unit test), le van fordítva és az elfogadási tesztek is megvannak. Valaki másban pedig csak annyit, hogy a kód le lett fordítva. Ha nem tudja mindenki mit jelent a „kész” definíciója, a tapasztaláson alapuló folyamat szabályozás másik két pillére

nem működik. Ha valaki úgy jellemez valamit, hogy „kész” akkor mindenki másnak tudnia kell, mit jelent az, hogy „kész”.

A „kész” határozza meg, hogy mit csinál a Csapat, amikor elkötelezi magát egy Terméknapló elem „elkészítésére” a Sprint alatt. Némely termék nem tartalmaz dokumentációt, tehát a „kész” fogalma sem tartalmazza a dokumentációt. Egy teljesen „kész” inkremens magában foglalja az inkremens és az azt felépítő összes Terméknapló elem analízisét, tervezését, felülvizsgálatát, programozását, dokumentációját és tesztelését. A tesztelésbe beleértendő az egység-, rendszer-, felhasználói- és regressziós-teszt ugyanúgy, ahogy a nem funkcionális tesztek, mint pl. a teljesítmény-, stabilitás-, biztonsági- és integrációs tesztek. A kész tartalmazza a nyelvi fordítást is. Egyes Csapatok még nincsenek felkészülve, hogy minden szükséges megvalósítandót beleértsenek a „kész” definíciójába. Ezt világossá kell tenni a Termék Tulajdonos számára. Ezt a fennmaradó munkát majd el kell végezni, a termék implementálása és használatba kerülése előtt.

#### Tipp

*Az „elvégzetlen” feladatokat gyakran egy „Elvégzetlen munka” vagy „Implementálandó feladat” nevű Terméknapló elembe gyűjtjük. Mivel így összefoglaljuk ezeket, a Termékbontás pontosabb marad, mintha nem fognánk össze ezeket az elemeket.*

## VÉGSŐ GONDOLATOK

Némely szervezet képtelen egy teljes inkremenst felépíteni egy Sprint alatt. Lehet, hogy még nem rendelkezik az automatikus tesztelési infrastruktúrával az összes teszt végrehajtásához. Ebben az esetben minden inkremenshez két kategória készül, a „kész” munka és az „elvégzetlen” munka. Az „elvégzetlen” munka azon része az egyes inkremenseknek amit még később be kell fejezni. A Termék Tulajdonos tudja, hogy mit vizsgál egy Sprint végén mivel az inkremens megfelel a „kész” definíciójának és tudja, hogy mi a „kész” definíciója. Az „elvégzetlen” munka egy „Elvégzetlen munka” nevű Terméknapló elembe kerül ahol összegyűlik és tisztán látszik a Kiadásbontás diagramon. Ez a technika átláthatóságot teremt a kiadás felé haladó folyamatban. A vizsgálat és alkalmazás a Sprint Áttekintőben annyira pontos, mint ez az átláthatóság.

Például ha egy Csapat nem tud teljesítmény-, regresszió-, stabilitási-, biztonsági- és integrációs tesztelést végezni minden Terméknapló elemre, kiszámítjuk ennek a munkának az arányát az elvégezhető munkához (analízis, tervezés, átalakítás, programozás, dokumentáció, egység és felhasználói tesztelés) képest. Tegyük fel, hogy ez az arány hat egység „kész” és négy egység „elvégzetlen” munka. Amikor a Csapat befejezett hat egység munkát (a Csapat az alapján becsül, amiről tudja, hogy kell csinálni), négyet hozzáadunk az „elvégzetlen munka”

Terméknapló elemhez.

Sprintről Sprintre növekszik az egyes inkremensek „elvégzetlen munka” mennyisége és ezzel foglalkozni kell, mielőtt kiadnánk a terméket. Ez a munka lineárisan növekszik, habár igazából van némi exponenciális jellege ami függ az egyes szervezetek jellemzőitől. Kiadási Sprintek kerülnek minden kiadás végére, hogy befejezzük ezeket az „elvégzetlen feladatokat”. A szükséges Sprintek száma olyan fokig nem becsülhető, amilyen fokig az „elvégzetlen” munka növekedése nem lineáris.