

Szoftvertchnológia és –technikák

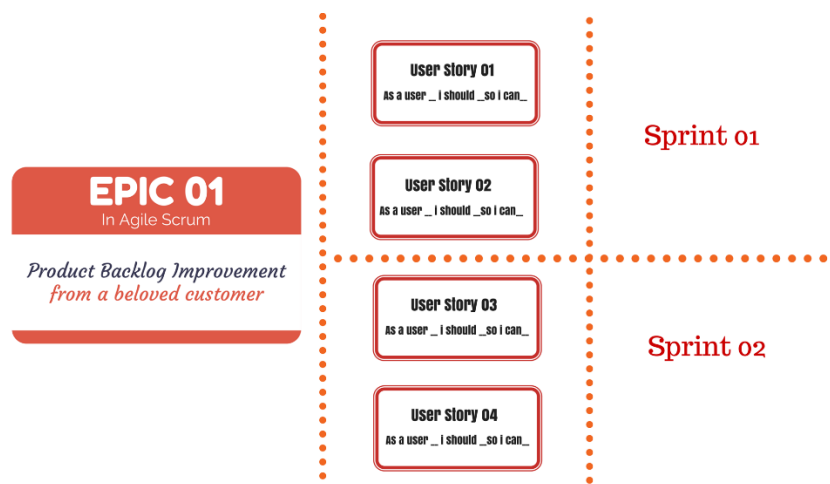
Szoftverfejlesztési módszertanok a gyakorlatban labor

1. Elméleti áttekintés

A laboron az agilis módszertanokból a Scrum-ot tekintjük át, (4-5 fős) csoportokba szerveződve. Mivel a kapcsolódó előadás később van a legtöbb csoportnak, mint a gyakorlat, kérjük, hogy nézzék át előre a fogalmakat.

1.1. Scrum

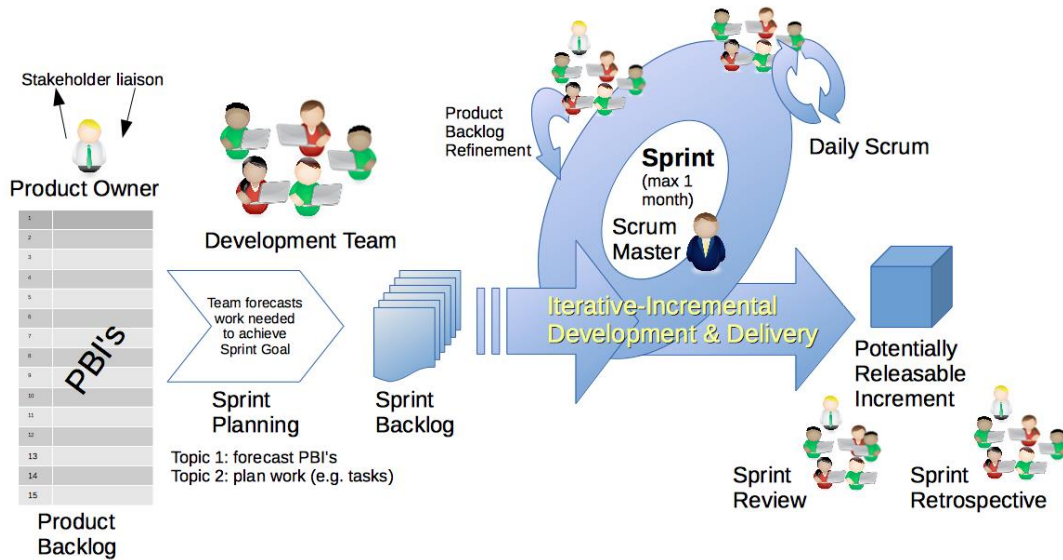
Scrum egy agilis módszertan, ahol a feladatot (EPIC) alfeladatokra (User story) bonjuk fel:



1. ábra Scrum tervezés lépései,

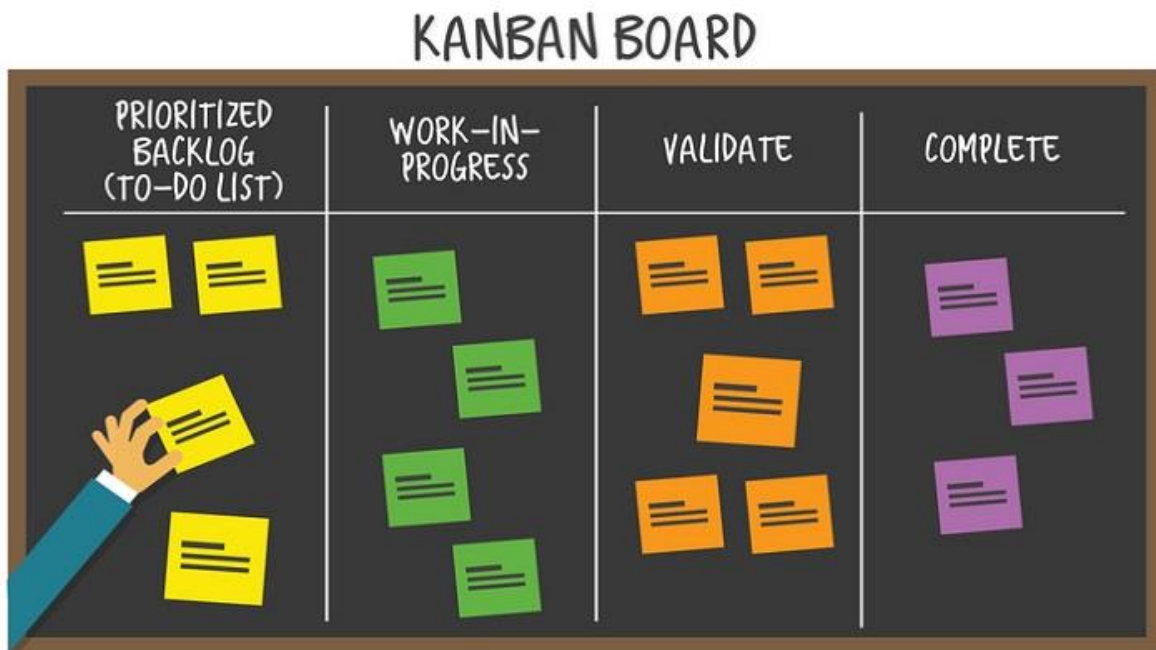
forrás: <https://www.yodiz.com/blog/what-is-epic-in-agile-methodology-definition-and-template-of-epic/>

A user story-k lesznek a Product Backlog Item-ek (PBI), azok az elemeket amiket fontosságuk alapján egy-egy sprint során elkészülnek. A Product Owner (PO) tartja a kapcsolatot a megrendelővel, kialakítja a User Story-kat (PBI) és prioritást rendel hozzájuk. A legnagyobb prioritással rendelkező PBI-ket kiválasztja a csapat. Megbecsülik, hogy mennyi idő telik azok lefejlesztése. Az fejlesztési idő becsléséhez a csapat story point-okat rendel minden user story-hoz. Ez tipikusan úgy történik, hogy egy két user story-hoz rendel egy átlag pontszámot és a többi story-t ahhoz képest becsüli. Egy sprinthez annyi PBI-t választanak ki amennyi egy Sprint ideje alatt lefejleszthető. A PBI-khez meghatározzák az alfeladatokat. Majd elindul a sprint, amit a Scrum master felügyel. A sprint végén megbeszéljük a tapasztaltakat, megmutatják az elkészült munkát a megrendelőnek.



1.2. Kanban tábla

Az adott sprintben elvégzendő feladatok állapotát szokás Kanban táblán követni:



Az adott sprint-re kiválasztott use story-k a TODO oszlopban kezdenek, majd jobbra haladnak. Tipikusan max 4 elem lehet a work-in-progress oszlopban, hogy a csapat ne kezdjen bele egyszerre sok feladatba. Fontos még meghatározni egy 'Definition of Done'-t, ami azt mondja meg, hogy mikor számít egy user story késznek.

Definition of Done példa:

- Implementáltam
- Lefordul
- Tesztek futnak
- Gitre pusholtam
- Többiek átnézték
- Többi komponenssel jól integrálódik
- Master branchbe merge-elve lett
- PO elfogadta

2. A labor menete

A labor első max. 10 percében az oktató a hallgatókkal átbeszéli a laboron előforduló alap fogalmakat. Ezt követően kezdődik el a valódi feladat megoldása. Olvassa el mindenki az 1. fejezetben leírtakat, hogy a fogalmakkal tisztában legyen!

A hallgatók megalapítják a 4-5 fős csapatokat. Csapatonként kijelölnek egy Product owner-t. A feladat egyszerűsége miatt Scrum Master szerepet most nem veszi fel senki, a csapat önmagát felügyeli. A laboron kód nem készül, csak képernyőképek és modellek. A labor végén jegyzőkönyvet adnak be a hallgatók, vagy bemutatják az összes elkészült eredményt a Laborvezetőnek (ez a jelenlét feltétele). A laboron alkalmazás mockup-okat (képernyő terveket) és modelleket (jellemzően UML diagrammokat), unit teszt specifikációkat kell készíteni. Mockup készítéshez több alkalmazás is elérhető a piacon. Mi a draw.io-t fogjuk használni. A hallgatók rajzolhatnak papírra is amit scannelve csatolnak a jegyzőkönyvhöz, vagy személyesen a labor alkalmával bemutatják a laborvezetőnek. Az egyes csapatok, a feladataik állását Kanban kártyákkal követik, ehhez minden csapat készít egy Kanban táblát a <https://trello.com/> weboldal segítségével. A labor során a Laborvezető a megrendelő szerepét veszi fel. Egy sprint 40 percig tart: 1-1 csapat Product Owner-jei minden sprint során tárgyalhatnak a megrendelővel. A sprintek időbeosztása a következő:

- (i) 5 perc egyeztetés a PO és a Laborvezető közt (az első sprintnél ez 10 perc)
- (ii) 10 perc becslés a csapatban (az egyes user story-khoz tartozó pontok megállapítása), a megvalósítandó user story kiválasztása, Kanban feladatok felvitele
- (iii) 15-20 perc munka (feladatkidolgozás)
- (iv) 5-10 perc beszámoló (PO – Laborvezető) és tapasztalatok összefoglalása

*Az első sprint előtt 5-10 percet számoljunk a labor ismertetésére, a fogalmak tisztázására és a csapatok megalakítására! Ezek után az első sprint 45 percet, a továbbiak ~40 percet vesznek igénybe. Négy sprint fér bele a laborba (10+10+45+3*40 perc), az utolsó sprint lehet rövidebb.*

Vigyünk magunkkal csapatonként 1-1 A4-es lapot, hogy a PO-k tudjanak mire jegyzetelni (ha nem lesz laptop náluk).

A laboron a következő feladatokat kell megoldani:

1. Feladat felosztása (feladat epic-ekbe és user story-kba osztása) – laborvezetővel közösen
2. Sprint 1 kivitelezése – részben laborvezetővel közösen

3. Sprint 2 kivitelezése – önállóan
4. 2 további funkció elkészítése (megrendelő további igényei) - önálló munka

3. A projekt

A projekt során a következő feladatot kell megoldani: BKV megkereste a cégünket, hogy fejlesszünk neki egy mobilos e-ticket rendszert. Ezt a fejlesztési folyamatot szimuláljuk.

3.1. Scrum: Initiate phase

A PO a Megrendelővel egyeztetve megállapítja a projekt víziót (avagy 'elvator pitch'), ami tisztázza, a projekt célját, stakeholdereket.

For (target customer)

Who (statement of need or opportunity)

The (product name) **is a** (product category)

That (key benefit, reason to buy)

Unlike (primary competitive alternative)

Out product (statement of primary differentiation)

A Laborvezető segítségével a PO-k a továbbiakban azonosítják a feladat nagyobb részeit, beosztják azt két epicbe, majd minimum hat user story-ba (összesen), amiket később tovább finomítanak a csapatok (ezt már önállóan), subtaskok formájában. Amíg a PO a Laborvezetővel tárgyal, a csapat többi tagja átnézi a Mockup készítő és Kanban tábla kezelő eszközöket, valamint megpróbál elkészíteni egy mobilalkalmazás képernyőt tetszőleges tartalommal!