

# **DIPLOMADOLGOZAT**

**Tóth Kata**  
**Ellátásilánc-menedzsment MSc.**

**Gödöllő**  
**2023**



**Magyar Agrár- és Élettudományi Egyetem  
Szent István Campus  
Ellátásilánc-menedzsment MSc Szak**

**Kanban bevezetése és működésének elemzése a szoftverfejlesztés területén**

**Belső konzulens:** Dr. Gyenge Balázs  
Egyetemi docens

**Készítette:** Tóth Kata  
CDTU3P  
Levelező

**Intézet:** Agrár- és Élelmiszergazdasági Intézet

**Tanszék:** Agrárlogisztika, Kereskedelem és Marketing Tanszék

**Gödöllő  
2023**

## Tartalomjegyzék

|                                                   |    |
|---------------------------------------------------|----|
| <i>Köszönetnyilvánítás</i> .....                  | 4  |
| 1. Bevezetés .....                                | 5  |
| 2. Lean menedzsment .....                         | 7  |
| 2.1. Lean menedzsment kialakulása.....            | 7  |
| 2.2. Lean menedzsment alapelvei és eszközei ..... | 8  |
| 2.2.1. Lean menedzsment eszközei .....            | 8  |
| 2.2.2. Lean menedzsment stratégiai szintje .....  | 10 |
| 2.2.3. Lean menedzsment operatív szintje .....    | 10 |
| 3. Kanban.....                                    | 12 |
| 3.1. Kanban kialakulása .....                     | 12 |
| 3.2. Kanban módszer ismertetése.....              | 12 |
| 3.3. Kanban módszer alkalmazási területei.....    | 14 |
| 3.4. Kanban eszközei .....                        | 14 |
| 3.5. Kanban főbb típusai .....                    | 17 |
| 3.5.1. Anyagfelvevő kanban .....                  | 17 |
| 3.5.2. Termelési utasítás kanban.....             | 18 |
| 3.6. Kanban előnyei .....                         | 18 |
| 3.7. Kanban kihívásai és korlátai .....           | 20 |
| 3.8. Kanban költségei .....                       | 21 |
| 3.9. Kanban szabályai .....                       | 21 |
| 4. Kanban alapjai a szoftverfejlesztésben .....   | 22 |
| 4.1. Agilis szoftverfejlesztés .....              | 22 |
| 4.2. Kanban.....                                  | 25 |
| 5. Egyéni rész bevezető .....                     | 27 |
| 6. Esettanulmány ismertető.....                   | 28 |
| 7. Kanban.....                                    | 29 |
| 7.1. Folyamat, szerepkörök .....                  | 29 |
| 7.2. Kanban tábla alkalmazása .....               | 31 |
| 7.3. Kanban kártyák mozgása.....                  | 34 |
| 7.4. Kanban alkalmazása WIP limittel .....        | 36 |
| 8. SCRUM.....                                     | 44 |
| 8.1. Fejlesztés SCRUM esetén .....                | 44 |
| 8.2. Scrum és kanban .....                        | 47 |

|      |                                                                     |    |
|------|---------------------------------------------------------------------|----|
| 9.   | Javaslatok a folyamaton belüli fejlesztésekre .....                 | 48 |
| 9.1. | CFD elemzés .....                                                   | 48 |
| 9.2. | További javaslatok az esetpélda folyamatának optimalizálására ..... | 52 |
| 10.  | Összefoglalás .....                                                 | 54 |
|      | Irodalomjegyzék .....                                               | 56 |
|      | Függelék .....                                                      | 58 |

## Ábrajegyzék

|           |                                                                      |    |
|-----------|----------------------------------------------------------------------|----|
| 1. ábra:  | Lean-ház (Blahó et al., 2021) .....                                  | 8  |
| 2. ábra:  | A lean menedzsment 5 alapelve (Demeter, 2016).....                   | 9  |
| 3. ábra:  | Példa kanban kártyákra (Cimorelli, 2013).....                        | 15 |
| 4. ábra:  | Példa kanban áramlási útvonalára (Demeter, 2016).....                | 16 |
| 5. ábra:  | Kanban típusai (Kosztolányi, KaizenPro/kanban, 2016).....            | 17 |
| 6. ábra:  | Agilis megközelítések (ProjectManagementInstitute, 2018).....        | 23 |
| 7. ábra:  | Software Development Life-Cycle (GuideForSoftwareTesting, 2014)..... | 24 |
| 8. ábra:  | Backlog feldolgozása - PULL .....                                    | 29 |
| 9. ábra:  | FreeGame kanban tábla .....                                          | 33 |
| 10. ábra: | Státuszok – PULL.....                                                | 35 |
| 11. ábra: | Státuszok 2.....                                                     | 35 |
| 12. ábra: | FreeGame kanban – PULL .....                                         | 36 |
| 13. ábra: | FreeGame kanban tábla WIP limittel .....                             | 38 |
| 14. ábra: | Scrum sprint.....                                                    | 44 |
| 15. ábra: | Scrum tábla .....                                                    | 46 |
| 16. ábra: | Kumulatív folyamatábra (CFD) .....                                   | 49 |
| 17. ábra: | Státuszokban töltött idő jelölése .....                              | 51 |

## Táblázat

|               |                                                  |    |
|---------------|--------------------------------------------------|----|
| 1. táblázat:  | Kanban - logisztika és informatika .....         | 26 |
| 2. táblázat:  | FreeGame funkció - backlog elemei .....          | 31 |
| 3. táblázat:  | FreeGame fejlesztési folyamat 1. lépés.....      | 39 |
| 4. táblázat:  | FreeGame fejlesztési folyamat 2. lépés.....      | 39 |
| 5. táblázat:  | FreeGame fejlesztési folyamat 3. lépés.....      | 39 |
| 6. táblázat:  | FreeGame fejlesztési folyamat 4. lépés.....      | 40 |
| 7. táblázat:  | FreeGame fejlesztési folyamat 5. lépés.....      | 40 |
| 8. táblázat:  | FreeGame fejlesztési folyamat 6. lépés.....      | 40 |
| 9. táblázat:  | FreeGame fejlesztési folyamat 7. lépés.....      | 41 |
| 10. táblázat: | FreeGame fejlesztési folyamat 8. lépés.....      | 41 |
| 11. táblázat: | FreeGame fejlesztési folyamat 9. lépés.....      | 42 |
| 12. táblázat: | FreeGame fejlesztési folyamat 10. lépés.....     | 42 |
| 13. táblázat: | FreeGame fejlesztési folyamat 11. lépés.....     | 42 |
| 14. táblázat: | FreeGame fejlesztési folyamat 12. lépés.....     | 43 |
| 15. táblázat: | FreeGame fejlesztési folyamat 13. lépés.....     | 43 |
| 16. táblázat: | FreeGame - becslési értékek .....                | 45 |
| 17. táblázat: | kanban vs scrum.....                             | 47 |
| 18. táblázat: | FreeGame fejlesztési adatok CFD elemzéshez ..... | 49 |

## ***Köszönetnyilvánítás***

Ezúton szeretnék köszönetet mondani témavezetőmnek, Dr. Gyenge Balázsnak, aki szakértelmével és a konzultációk során biztosított elengedhetetlen tanácsaival hatalmas segítséget nyújtott diplomamunkám elkészüléséhez.

# 1.Bevezetés

A termelési rendszerek kialakításának folyamatai már évszázadokkal ezelőtt elkezdődtek, amikor nagy mennyiségű termékekre jelentek meg vevői igények. A minőség, rövid szállítási idő és alacsony ár elérése már akkor is előtérben volt. (GyartasTrend, 2009)

Külön nehézséget jelentett a vállalatok számára, hogy a vevői igények folyamatosan változtak. Ezeket figyelembe véve kezdődtek meg a törekvések a gyártási és tömegtermelési folyamatok optimális kialakítására.

Természetesen, ez azóta is végigkíséri a vállalatok életét. Cél, hogy a lehető leghatékonyabb termelést ki tudják alakítani a vállalat minden területére vonatkozóan, a vevői igények kielégítése és minél nagyobb elégedettségi szint elérése mellett. A megfelelő módszer kiválasztása több tényezőtől is függ, nincs olyan, ami az összes vállalat számára tökéletes lenne, így érdemes alapos kutatást végezni és az alapján dönteni.

Számtalan módszer áll rendelkezésünkre, amik választási lehetőségként jelenhetnek meg a folyamataink optimalizálásának tervezésénél. Ezek közül az egyik legelterjedtebb a kanban, amit rendkívül sikeresen alkalmaznak vállalatok milliói.

Az informatika rohamos fejlődésével elkerülhetetlenné vált, hogy ezen a területen is keretbe és rendszerbe foglalják a csapatok működését és a feladatok végrehajtását, ezzel koordinálva a szoftverek fejlesztésének ütemét. A szoftverfejlesztés hatalmassá nőtte ki magát az elmúlt időszakban, ezt az is alátámasztja, hogy a legtöbb vállalkozás nem tudna eredményesen működni informatikai rendszerek megléte nélkül. Nehezen lehetne olyan szolgáltatást találni, ahol nem játszik szerepet semmilyen szoftver, akár csak a háttérfolyamatokban. Ezek a szoftverek, bár nem fizikai termékek, a „legyártásuk”-ban mégis felfedezhetünk pár hasonlóságot a fizikaival összehasonlítva, mint például, hogy a megfelelő módszerek nélkül ugyanúgy káoszt eredményezhet a termelés. Veszteséges lehet mind a szoftverfejlesztő vállalat, az ügyfél és a felhasználók számára is.

A megfelelő módszer kiválasztásán azért is van nagy hangsúly, mivel az kihat az egész fejlesztési folyamatra, ezáltal a hatékonyságra, a csapat teljesítményére és a felmerülő problémák megoldásainak lehetőségeire is. Emellett, meghatározza a csapaton belüli kommunikációt és azok eszközeit, sőt még a csapat hangulatát is befolyásolhatja. Ezek mindegyike kihatással lehet a projektcélok elérésére, a határidők betartására, valamint a

felhasználók vagy ügyfelek igényeinek teljesítésére és elégedettségére.

A mai „rohanó világ”-ban mindenki mindent azonnal szeretne és ez az IT cégekre és ügyfeleikre is igaz. Hatalmas a verseny a vállalatok között és ez megköveteli, hogy minél gyorsabban és rugalmasabban reagáljanak a felmerülő igényekre. A kanban egy igazán jól működő módszer a gyártásoknál, így az informatikában is megjelent idővel. Mivel a szoftverfejlesztés egyértelműen az agilis projektmenedzsment felé tendál a folyamatosan változó ügyféligények miatt, így a kanban ezen a területen is rendkívül népszerű.

Dolgozatomban azt szeretném bemutatni, hogy a kanban hogyan értelmezhető és használható effektíven a szoftverfejlesztés területén, a következő célok mentén:

1. Kanban sajátosságainak és működésének elemzése, hatékonyságának vizsgálata a szoftverfejlesztésben
2. Kanban összehasonlítása egy másik - a szoftverfejlesztés területén használt - népszerű módszertannal (scrum)
3. Kanban használata esetén felmerülő nehézségek elemzése, javaslatok a folyamat fejlesztésére

Az egyes célok esetpéldákon keresztül kerülnek megvalósításra a későbbi fejezetekben.

## 2. Lean menedzsment

### 2.1. Lean menedzsment kialakulása

A kanban kialakulása a lean rendszerhez fűződik, így mindenképpen érdemes először a lean menedzsment megismerésével kezdeni.

A lean rendszer a klasszikus tömegtermelési rendszerrel megy szembe, a neve is az összehasonlításukból származik (Demeter, 2016).

A lean gyártási rendszer kialakulása egyértelműen a Toyota gyár nevéhez fűződik, gyökereit az itt kifejlesztett termelési rendszerben találhatjuk meg. Célja a hatékonyabb termelés kialakítása volt. Megfigyelve a korabeli autógyártók termelési folyamatait, az amerikai és nyugat-európai gyártók a Henry Ford féle tömegtermelést alkalmazták, míg a japán gyártók a Just In Time (JIT) rendszerre alapoztak. Daniel T. Jones, James P. Womack és Daniel Ross által kiadott könyvükben (*The Machine that Changed the World*) elemzésre kerül a két módszer elemeinek összehasonlítása, aminek következtetése, hogy a japán gyártók által alkalmazott eszközök sokkal hatékonyabb termelést értek el, mint a készletre termelés. (GyartasTrend, 2009)

A hatékonyabb termelést és nagyobb profitot a produktivitás növelésével és költségek csökkentésével szerették volna elérni – „Profit through cost reduction”. Természetesen a „költség” sokféleképpen értelmezhető, mivel nem csak a gyártási költség tartozik ehhez, hanem például az adminisztratív jellegű költségek, marketing költségek és még sok más. A Toyota Production System fő elképzelése viszont arra irányult, hogy a költségek csökkentését a pazarlás teljes megszüntetésével tudnák elérni a leghatékonyabban. A fő céljuk – költségcsökkentés - mellett további alcélokat határoztak meg, mint például a mennyiség kontrollálhatósága, minőség biztosítása és az emberek iránti tisztelet. (Monden, 2012)

Tehát a lean menedzsment elsődleges célja a vevői értékteremtés, illetve a pazarlás minimumra csökkentése vagy annak teljes megszüntetése (Demeter, 2016). Ez nagyban hozzájárul a hatékony termelés kialakításához.

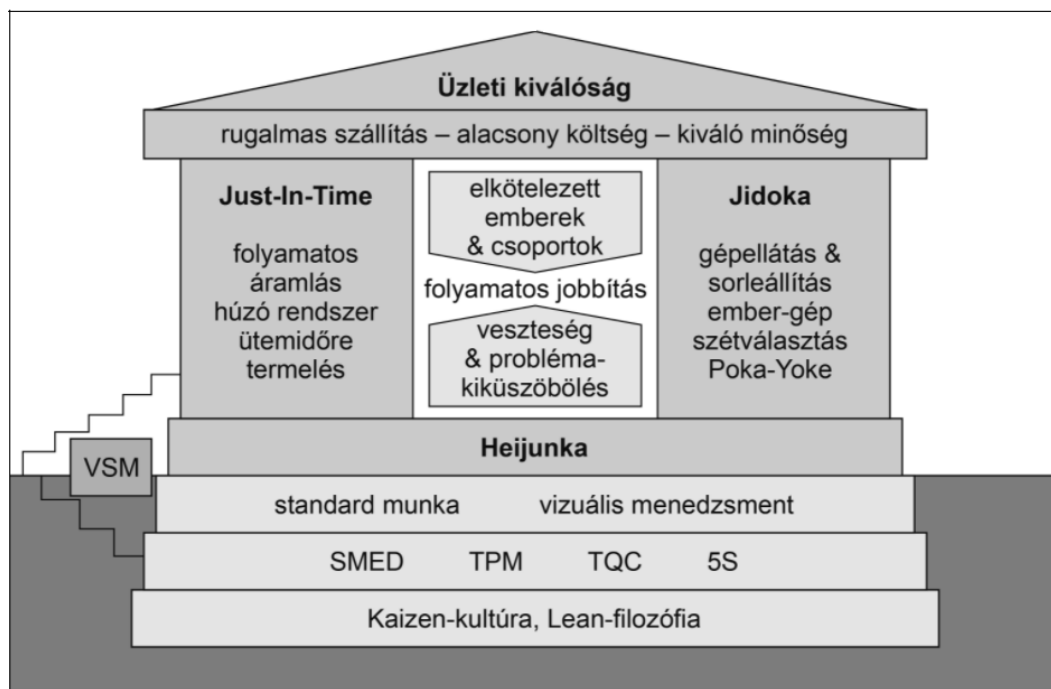
Ahogy az látható, a lean menedzsment kialakulása a Toyota termelési rendszer kiforrásával kezdődött, ami az 1940-es évekre tehető. Ebben az időszakban a Toyota

számos innovációt mutatott be. A lean fogalom 1990-től vált nemzetközileg ismertté és a leghatékonyabb megvalósításának a Toyota termelési rendszert vallották. Az évtizedek alatt a rendszer egyre tovább fejlődött és egyre több területet ölelt fel. Az 1990-es évektől beszélhetünk lean menedzsmentről. A menedzsment részévé váltak különböző szolgáltatások, egyedi igényeket kielégítő folyamatok stb. A lean filozófia és lean vállalat megjelenése is erre az időszakra tehető, ami magában foglalta a szervezeti közeg és szervezeti funkciók összhangját a lean elvekkel. (Kovács, 2017)

## 2.2. Lean menedzsment alapelvei és eszközei

### 2.2.1. Lean menedzsment eszközei

A lean-t tulajdonképpen egy gondolkodásmódként lehet definiálni, ami arra irányul, hogy a vállalat a „veszteségeit” milyen módon tudja csökkenteni. Más szóval azt a kérdést helyezi előtérbe, hogy a vállalat hogyan tud működni úgy, hogy minél kevesebb erőforrást és felszerelést vegyen igénybe, illetve emellett fontos, hogy mindez minél kevesebb helyen és minél kevesebb idővel társuljon megvalósítás szinten. Sokak szerint a fő hangsúly a vevői értékeken és a veszteségek kiiktatásán van. A lean-menedzsmentet egy átfogó rendszerként lehet leírni, ennek a szemléletmódnak az ismertetésére szolgál a lean-ház. (Blahó et al., 2021)

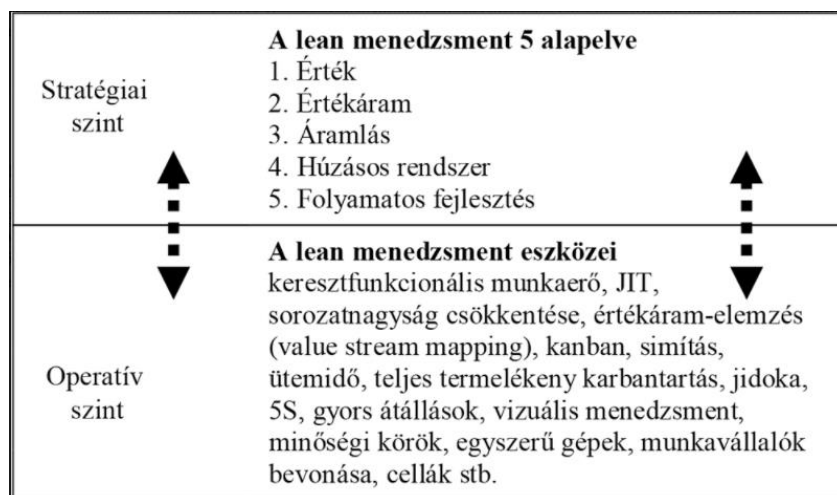


1. ábra: Lean-ház (Blahó et al., 2021)

A lean-háznak többféle elrendezése is lehet, az 1. ábrán látható az egyik elméleti megvalósítás. A Just-In-Time építőelem alá sorolható például a kanban vagy a pull (húzó) termelés, a Jidoka alá pedig példaként említhető a Poka-Joke, Ishikawa, Andon stb.

Az ábrán is látható, hogy a lean-ház eszközei hozzájárulnak ahhoz, hogy rugalmasabb szállítást, alacsonyabb költségeket és kiváló minőséget érjünk el a vállalaton belül, ezzel a vevői elégedettséget nagy mértékben növelve. A ház alapját képezi a Kaizen-kultúra és Lean-filozófia, ezek elsajátítása elengedhetetlen az eszközök hatékony használatához. Továbbá, a ház teteje szimbolizálja a kitűzött célokat, az eszközök pedig a megvalósítás lehetőségét adják, ezek az építőelemek. Az elemek mindegyike fókuszál valamire, például az 5S a munkahely rendezett kialakításában lehet a vállalat segítségére, a heijunka a termelés szinkronizálásához járul hozzá stb. A megfelelő eszközök kombinálása és hosszú távú alkalmazása nagyfokú előrelépést biztosíthat az üzleti kiválóság irányába. Természetesen az eszközök kiválasztása sok mindentől függhet. Alapos elemzést igényel a vállalat jelenlegi működése, ami segít a fejlődési pontok azonosításában és azok prioritizálásában a megfelelő célok elérésének érdekében.

A lean rendszerrel megkülönböztetünk stratégiai és operatív szintet. A stratégiai szint a rendszer 5 alapelvét írja le, az operatív szint pedig a menedzsment eszközeit tartalmazza:



2. ábra: A lean menedzsment 5 alapelve (Demeter, 2016)

## 2.2.2. Lean menedzsment stratégiai szintje

Az 2. ábrán látható stratégia elvek a következőképpen értelmezhetők:

- **Érték:** értéknek nevezzük mindazt, amit a vevő megfizet. Többféleképpen lehet értéket teremteni:
  - Belső pazarlások megszüntetésével
  - Vevői érték megteremtésével
- **Értékáram:** értékteremtő folyamatok értékárammá szervezése
- **Áramlás:** akadályok nélküli folyamatos áramlások biztosítása lean eszközökkel
- **Húzásos rendszer:** a pull stratégia a push stratégiával ellentétben visszafele működik, ebben az esetben tulajdonképpen a felhasználói igény határozza meg a termelést. A pull rendszert a vevői igények irányítják, így a készletezés is elkerülhető.
- **Folyamatos fejlesztés:** kaizen, vagyis a folyamatos tökéletesítésre törekszik a lean menedzsment.

(Koncz, 2017)

## 2.2.3. Lean menedzsment operatív szintje

A fentiekben a lean menedzsment stratégiai szintjei kerültek ismertetésre. Az operatív szint a menedzsment eszközeit írja le. A 1. ábrán látható, hogy a lean-nek kettő alappillére van.

Az egyik alappillére a JIT. A JIT rövidítés a „just in time”-nak felel meg, ami azt jelenti, hogy „éppen időben”. Ez a kifejezés tulajdonképpen annyit tesz, hogy „akkor és amit kell”, nem hamarabb, nem később, mindent ott és akkor, amikor szükség van rá. (Transpack, 2017)

A JIT alap felfogása szintén a készlet nélküli vagy minél kevesebb készlettel történő működést célozza meg. Ezek alapján megkülönböztethetünk JIT beszállítást és JIT termelést. Beszállítás esetén úgy valósul meg, hogy a termékekből kevés készletet tartanak és a beszállító viszonylag gyakran szállít, így elkerülhető a nagyobb készletezés. A JIT termelés esetén viszont a termelési folyamatokon belül is az alacsony készletezésre törekszenek. (Kovács, 2017)

A lean másik alappillére a jidoka. A jidoka fő célja a jobb minőség elérése alacsonyabb költséggel. A jidoka által azt szeretnék elérni, hogy a gépek is felismerjék a hiba

előfordulását és megakadályozzák a hibás termék továbbadását. Ezek alapján a két alapelve, hogy:

- A gép felismeri hibát és megáll
- Az ember és a gép munkájának elkülönülése

Mivel a gép magától felismeri a hibát és lekezeli, így az ember értékteremtőbb feladatokkal is tud foglalkozni, nem kell a gép felett őrködni, csak akkor kell beavatkozni, amikor a gép jelez. (Kosztolányi, Kaizenpro, 2016)

A két alappillér mellett számos eszköz szerepel a lean operatív szintjén, mint például a keresztfunkcionális munkaerő, simítás, ütemidő, munkavállalók bevonása, kanban stb.

## **3. Kanban**

### **3.1. Kanban kialakulása**

A kanban történetének kezdete a 20. század közepéhez és a Toyota gyárhoz köthető. A japán kanban szó jelentése kártya vagy tábla, de jelent jel-t is, mivel egyfajta jelzésként szolgál. Kialakulása Taiichi Ohno nevéhez fűződik. (PromanConsulting, 2020)

Sokan a Toyota Production Systems-re kanban rendszerként hivatkoznak, de valójában az a pontosabb megfogalmazás, hogy a kanban egy olyan rendszer, ami kontrollálja a gyártási mennyiségeket az összes folyamatban. (Monden, 2012)

### **3.2. Kanban módszer ismertetése**

A kanban a „Just in time (JIT) termelési rendszer megvalósításának eszköze. A JIT lényege: „a szükséges terméket a szükséges mennyiségben, a szükséges időben gyártani, mozgatni. A JIT-rendszer működtetésének három alapelve az ütemidővel történő gyártás, a folyamatos anyagáram és a húzó rendszer. Ehhez szükség van a vevői igények kiegyenlítésére és a legismertebb JIT-eszköz, a kanban használatára. A JIT célja a termelési átfutási idő csökkentése.” (LeanSzotar, 2012-2021) Az átfutási idő alatt azt az időintervallumot értjük, ami a gyártás kezdetét és befejezését öleli körbe.

A kanban módszer segít irányítani a JIT elvét és elősegíti, hogy tényleg csak azokat a termékeket gyártsuk le, amikre szükség van. „Húzó elv”-ként is hivatkoznak rá, vagyis csak azt a terméket és mennyiséget állítják elő, amit a vevői igény megkövetel, ellentétben a „toló elv”-vel.

Ezzel a gyártási megközelítéssel szinte a minimumra csökkenthető a raktárkészlet. A raktárkészlet felhalmozása azért nem célszerű, mert ha túl sok terméket gyártanak le előre, akkor a pénz úgymond „áll” benne. Egy cég életciklusában számtalan olyan helyzet léphet fel, amikor pénzt kell mozgósítani. Akár egy beruházás miatt, akár az alap-költségek fedezésének érdekében. A legjobb esetben a vállalat saját tőkéből tudja fedezni a pénz szükségleteket. Amennyiben túl nagy raktárkészletet tart egy vállalat, úgy azzal a lépéssel csökken a forrás-kapacitás, így adott helyzetben kénytelen lesz külső forrásból megoldani,

ami legtöbbször költségesebb. Másik hátránya a raktárkészlet felhalmozásának, hogy a termékek amortizálódhatnak ez alatt az időtartam alatt. Illetve, az sem elhanyagolható tényező, hogy a készlet tárolásának költsége általában nagyon magas, így az is tőkét köt le. Ezek mind költségtényezőként szerepelnek a vállalatnál.

Mindent egybevéve, azt az állapotot érdemes elérni, ahol minél kisebb felhalmozott raktárkészlettel minden vevői igényt időben ki tudnak elégíteni.

A kanban egyik célja, hogy optimális szinten tartsa a készleteket. A pillanatnyi készletgazdálkodást célozza meg, amik a vevői igények alapján kerülnek meghatározásra. Emellett fontos szerepe van a szükséges feladatok megfelelő időben történő végrehajtására is. Tehát tulajdonképpen a valós igényekre építi a termelés koordinálását.

A kanban ezen kívül lehetővé teszi, hogy a vevői igények változásához igazítsa a termelést. Az előre legyártott termékeken már legtöbbször nehéz változtatni és a legtöbb vállalat célja a vevői igények kielégítése megfelelő nyereség elérése mellett, amibe érdemes belekalkulálni, hogy az igények változhatnak. A kanban alkalmazásával a gyártás és termelési terv ehhez igazodik. Erre legegyszerűbb példaként szolgálhat az autógyártás, ami esetén a vevők az alap felszereltséghez kedvük szerint választhatnak további extra funkciókat felár ellenében. Amennyiben az autók előre le vannak gyártva, úgy utólag sokkal nehezebb és költségesebb a gyártó számára módosítani a meglévő felszereltséget (például egy manuális klíma kicserélése digitális klímára, vagy vevő által igényelt színösszeállítások), mint már az elején az adott konstrukciókkal legyártani. A vevők ez által egy olyan autót kapnak kézhez, amiben a számukra preferált funkciók adottak. (Gelei, 2016)

A kanban akkor tud a legjobban működni, amikor a jövőbeli igények könnyen megjósolhatóak és a jelenlegi információk birtokában arra számít a vállalat, hogy viszonylag stabil marad. Természetesen, emellett további tényezők is számítanak, mint például költségek, éves mennyiség, átfutási idők stb. Egy vállalat csak úgy tud dönteni a megfelelő módszer kiválasztását tekintve, ha minden tényezőt figyelembe vesz. (Cimorelli, 2013)

A módszer alkalmazásának megértése egyszerű és emellett nagy fejlődés érhető el vele a vállalat működésében. Ez nem csak a vállalat számára szolgálhat pozitív eredményekkel, de a vevő részére is.

### 3.3. Kanban módszer alkalmazási területei

A kanban módszert számos helyen alkalmazzák sikeresen. Szinte bármilyen területen felmerülhet opcióként a bevezetése, ahol gyártás és szállítás történik. Nem csak fizikai termékek gyártásáról lehet szó, szolgáltatások és feladatok koordinálásában is segítségünkre lehet.

Főbb területeiként említhető például az autóipar, vasúti szállítás, de szintén megtalálható az informatika területén is, ahol manapság a projektmenedzsment egyik leghatékonyabb és leggyakrabban használt eszköze. Ebben az iparágban szívesen ötvözik más módszerekkel is (például scrum), így elérve egy hibrid rendszert, ami csak részben kanban. Ennek nagy előnye, hogy a kanban lehetséges hátrányait és nehézségeit leküzdhetjük egy másik módszer rész-megvalósításával.

Egyszerű alkalmazhatósága és könnyű tanulhatósága miatt a legtöbb iparágban bevezethető és hatékonyan alkalmazható, viszont ez nem azt jelenti, hogy minden esetben a kanban a legmegfelelőbb választás.

### 3.4. Kanban eszközei

A kanban alapja az információáramlás. Erre általában segítségül szolgálnak az úgynevezett információs kártyák, amik által könnyű koordinálni, hogy:

- mit
- mikor
- milyen ütemben
- és mennyit

kell gyártani és szállítani ahhoz, hogy fenntartsuk a legmegfelelőbb szintet. A kártyák segítségével alacsony készlet mellett is tudjuk biztosítani a rendelkezésre állást, tehát a csökkentett készlet szint nem megy az igények kielégítésének a rovására.

Sokféle módszer használható a jelölésre, de általában kártyák segítségével valósul meg az információáramlás. A kártyák egyfajta szükséglet vagy igényt jeleznek, ez által irányítható a termelés vagy szállítás.

A fent említett feladatok koordinálásának érdekében egy tipikus kanban kártyának a következő információkat mindenképpen tartalmaznia kell:

- cikkszám
- megnevezés
- mennyiség
- melyik folyamatba kell továbbítani (ez tulajdonképpen a következő állomás)
- sorozatnagyság és standard mennyiség (jel kanbannál)

Előfordulhat, hogy a kártyán több információ is szerepel, amennyiben ezt megkövetelik az állomások és szükséges a hatékony áramláshoz (például tárolótípus, honnan érkezik a kanban stb). (Kosztolányi, KaizenPro/kanban, 2016)

|                                                                                   |               |             |                                               |
|-----------------------------------------------------------------------------------|---------------|-------------|-----------------------------------------------|
| P/N: XXXXX-XX                                                                     |               |             | O<br>R<br>D<br>E<br>R<br><br>C<br>A<br>R<br>D |
|  |               |             |                                               |
| Desc: Lock Washer                                                                 |               |             |                                               |
| Note: Inspection Required                                                         |               |             |                                               |
| U/M: Pcs                                                                          | ABC: A        | LT: 5       |                                               |
| Supplier: ABC Fasteners                                                           |               | Code: 12345 |                                               |
| Cont: Box                                                                         | Loc: AAA X 5  |             |                                               |
| Card 1 of 3                                                                       | Card Qty: 100 | OP: 388     |                                               |
|                                                                                   |               |             |                                               |
|                                                                                   |               |             |                                               |

Figure 1.1 Example of external supplier order kanban card.

|                                                                                     |               |  |                                          |
|-------------------------------------------------------------------------------------|---------------|--|------------------------------------------|
| P/N: XXXXX-XX                                                                       |               |  | M<br>O<br>V<br>E<br><br>C<br>A<br>R<br>D |
|  |               |  |                                          |
| Desc: Lock Washer                                                                   |               |  |                                          |
| From Location: AAAX 5                                                               |               |  |                                          |
| U/M: Pcs                                                                            | Card Qty: 100 |  |                                          |
| Cont: Bin Type 1                                                                    |               |  |                                          |
| To Location: ABD X 1                                                                |               |  |                                          |
| Card 1 of 3                                                                         |               |  |                                          |
|                                                                                     |               |  |                                          |
|                                                                                     |               |  |                                          |

Figure 1.2 Example of move kanban card.

### 3. ábra: Példa kanban kártyákra (Cimorelli, 2013)

Ahogy a fenti példa-kártyákon is látszik, a kanban kártyák mindig csak azt a minimális mennyiségű információt tartalmazzák, ami szükséges a megfelelő információáramláshoz. (Cimorelli, 2013)

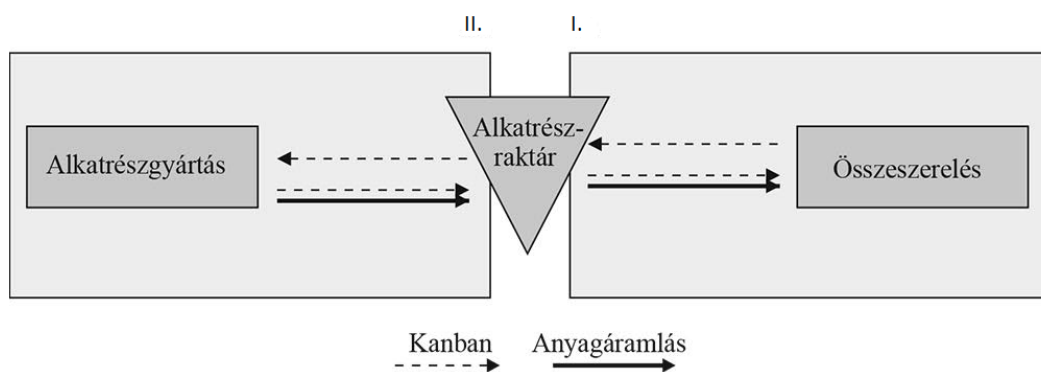
A kártyának a következő szabályokat célszerű követni ahhoz, hogy az állomások munkáját elősegítse:

- egyszerű
- egyértelmű utasítást ad

- vizuális irányítás alapelveit követi
- színmenedzsment szabályai érvényesülnek

(Kosztolányi, KaizenPro/kanban, 2016)

A kártyák hozzájárulnak a termelésirányítás megkönnyítéséhez, segítségükkel egyértelműen nyomon követhetők, hogy milyen tevékenységeket kell elvégezni. Ezen kívül, hozzájárul ahhoz, hogy a készletszint bármikor könnyen ellenőrizhető legyen. Továbbá, a készletszint növelése vagy csökkentése rendkívül egyszerűen elérhető a kanban kártyák számának módosításával. (Demeter, 2016)



4. ábra: Példa kanban áramlási útvonalára (Demeter, 2016)

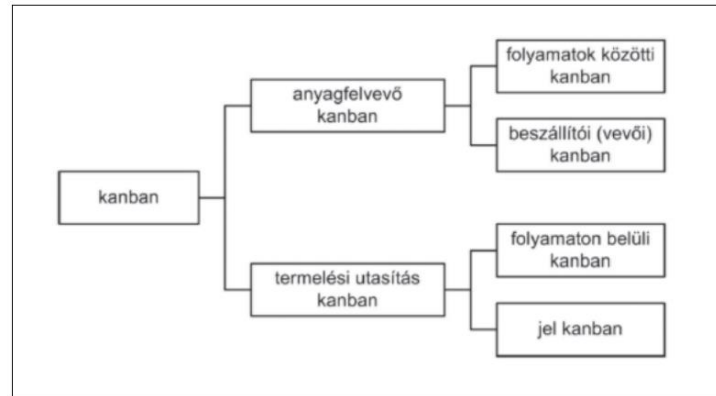
A 4. ábrán látható, hogy a kanban kártyák – és ezzel egyben az információ – mindkét irányba áramlik, anyagáramlás pedig az alkatrészgyártástól az összeszerelés folyamatáig történik. A folyamat azzal kezdődik, hogy megjelenik a vevői igény és az összeszerelés fázis megindul – I. fázis -. Ez által, az összeszerelés folyamata elhasznál egy doboz alkatrészt, így az üres doboz kanbanja átkerül az alkatrész raktárba. A kártya leadása azt eredményezi, hogy az összeszerelés fázis kap egy tele doboz alkatrészt az alkatrész gyár pedig látja, hogy mit kell pótolni. A kanban átkerül az alkatrészgyártásba, ahol legyártják az “elhasznált” dobozt. A kész dobozt visszajuttatják a kanbannal együtt az alkatrész raktárba, ami akkor kerül tovább az összeszerelés fázisba, amint jelentkezik igény rá. (Demeter, 2016)

A fent szemléltetett példa bemutatja a kanban kártya egy használati esetét. A folyamatból látható, hogy tulajdonképpen a vevői igény megjelenése generálja az utánpótlásokat és az elvégzendő feladatokhoz a kártyák adnak iránymutatást az állomásoknak.

## 3.5. Kanban főbb típusai

A kanban által kiadott utasítás alapján két fő típust különböztethetünk meg:

- anyagfelvevő kanban
- termelési utasítás kanban



5. ábra: Kanban típusai (Kosztolányi, KaizenPro/kanban, 2016)

### 3.5.1. Anyagfelvevő kanban

Anyagfelvevő kanban anyagmozgatásra vonatkozó utasításokat tartalmaz, ez további kettő típusra bontható.

Altípusai:

- Folyamatok közötti kanban
  - Lényege, hogy a jelenlegi folyamat az előző folyamattól át tud venni munkadarabokat. Ez a korábban említett kártyás módszer segítségével érhető el.
- Beszállítói/vevői kanban
  - Ennek esetében, a kártyák segítségével beszállítóktól tudunk rendelni. A vevői kanban annyiban tér el a beszállítótól, hogy a vevő beszállítóira vonatkozik a kártya.

(Kosztolányi, KaizenPro/kanban, 2016)

### 3.5.2. Termelési utasítás kanban

Ebben az esetben termelési utasításokra vonatkoznak a kanban kártyák.

Altípusai:

- Folyamaton belüli kanban
  - Ahogyan arra a neve is utal, ebben az esetben a kártyák segítségével a folyamaton belül tudjuk koordinálni a gyártást
- Jel kanban
  - Például hosszú átállási idő esetén használják, így egyszerre nagyobb mennyiséget gyártanak. A kártyát típusonként helyezik el. A sorozatnagyság jelzi, hogy hány dobozt kell gyártani a kártya lekerülésekor, a standard mennyiség pedig azt határozza meg, hogy hova kell elhelyezni a kártyát.

(Kosztolányi, KaizenPro/kanban, 2016)

A kanbant azért is nevezik „húzó rendszer”-nek, mert a kártyák áramoltatásával a mennyiség fogyását jelezzük és ez idézi elő a következő termék legyártását. A gyártási folyamatláncban tulajdonképpen ezekkel a kártyákkal kommunikálnak az egyes láncszemek és ezzel válik egyértelművé, hogy mire van szükségük.

A kanban lényegében egy jelként szolgál, amivel egyszerűen felismerhető, hogy egy termék jelen van-e a polcon vagy hiányként szerepel. Amennyiben a munkaállomás azt az információt kapja a kanbantól, hogy a termék hiányzik, úgy maga után húzza a gyártást.

Ezzel ellentétben a toló rendszer esetén a gyártósor folyamatosan termel, függetlenül attól, hogy a következő munkafolyamat igényli-e. Ennek következménye lehet, hogy felhalmozódnak a termékek feleslegesen, mivel a vevői igény nem követeli meg a legyártott mennyiséget.

### 3.6. Kanban előnyei

A kanban bevezetésének és használatának számtalan pozitív hatása van a vállalat működésére vonatkozóan.

Néhány példa a kanban legfőbb előnyeire:

- Költségek csökkentése
  - A kanban bevezetésével jelentős költségcsökkentéseket érhetünk el a vállalat működésében, mivel a termékek csak akkor kerülnek legyártásra, amikor arra szükség van. A kanban termelés-szabályozásának köszönhetően elkerülhetjük a túltermelést, mivel a „húzó elv” lévén csak a ténylegesen szükséges termékmennyiség kerül legyártásra. További költségcsökkentéshez vezetnek például a rövid átfutási idők is.
- Veszteségek csökkentése
  - A Just-In-Time model a veszteségek csökkentését is előtérbe helyezi. Valójában ez a pont összefügg a költségek csökkentésével is, mivel a veszteségek csökkentése költség-csökkentést is előidéz és a vállalatok pénzügyi biztonsága ezáltal nő.
- Flexibilitás
  - A kanbannal egy rugalmas rendszert építhetünk ki, ami könnyen kezeli a változó igényeket és gyorsan reagál azokra. A folyamatokban nagyfokú flexibilitást biztosít.
- Magasabb színvonalú vevőkiszolgálás
  - A hatékony termelési folyamatok által jelentősen rövidebb kézbesítési időt eredményezhetnek. Ennek köszönhetően a vásárlók elégedettebbek lesznek és nő a vállalat/márka iránti elköteleződésük.
- Gyorsan és könnyen optimalizálható

(LogAp, 2021)

Ezen kívül még számtalan előnyt köthetünk a kanbanhoz. Például – ahogy a korábbiakban is volt szó róla - a kártyák egyszerű és átlátható rendszert eredményeznek. Valamint, a folyamatok vizualizálása nagyban hozzájárul a követhetőséghez és megkönnyíti a résztvevők feladatainak azonosítását. Ezen túl, könnyen mérhető és szabályozható, ami lehetővé teszi, hogy felismerjük a rendszerünk hibáit.

### 3.7. Kanban kihívásai és korlátai

A kanbant nem minden esetben érdemes bevezetni a vállalat működésébe, nem jelent megoldást minden helyzetben. Ahogy az a korábbiakban is említésre került, érdemes részletes elemzést készíteni a vállalat működésére vonatkozóan (például a vásárlói igények ingadozásáról), ahhoz, hogy tisztábban lássuk, előnyt jelent-e a vállalat számára a kanban bevezetése vagy épp ellenkezőleg.

A kanban fő célja a készletek csökkentése, ennek megvalósításának érdekében viszont minél pontosabban kell tudni meghatározni (illetve megbecsülni) a várható igényt. Amennyiben az igény szignifikánsan változik állandóan, úgy nehezen számítható ki előre, így a kártyák nem tudják hatékonyan irányítani a gyártási folyamatokat.

E mellett a kártyákat folyamatosan igazítani kell a megnövekedett vagy akár lecsökkent vásárlói igényekhez mérten, ez alapján újra kell kalkulálni annak érdekében, hogy a gyártás megfelelően legyen koordinálva.

Egyértelműen kikövetkeztethető, hogy nagymértékű igény ingadozása esetén a kanban használata nem viszi előre a vállalat működését, sőt. Ebben az esetben leginkább többletmunkával jár a folyamatos kalkulálások miatt és a gyártási folyamatokat is megbonyolítja.

A fentiekből következik, hogy a kanban azokban az esetekben éri meg, amikor a kereslet ingadozása nem számottevő.

(Kosztolányi, KaizenPro/kanban, 2016)

További hátrányai például:

- szigorú munkafegyelmet igényel
- előfordulhat, hogy a résztvevőknek nem átlátható az egész folyamat, csak a szomszédos tevékenységek
- vevői igények nem megfelelő meghatározása esetén készlethiány léphet fel
- munkafolyamatok közötti átállások miatt idővesztés léphet fel

### **3.8. Kanban költségei**

Mint minden új rendszer, így a kanban bevezetése is költségekkel járhat az elején, de hosszú távon jelentős költségcsökkentés érhető el vele. Plusz költségként jelenhet meg kanban használata esetén a munkafolyamatok közötti váltás és az információáramlással eltöltött idő. Az egyes munkaállomások nem megállás nélkül termelnek, így hosszabb időbe telhet, míg elvégzik feladataikat. További költségként jelenhet meg a rendszer bevezetése, kialakítása, monitorozása, karbantartása és a munkavállalók betanítása is.

### **3.9. Kanban szabályai**

A kanban főbb szabályai 5 pontban írhatóak le, melyek a következők:

- Egy későbbi folyamat mondja meg egy korábbi folyamatnak, hogy mikor szükségesek új termékek/alkatrészek stb.
- A korábbi folyamat gyártja le azt, amire a későbbi folyamatnak szüksége van
- Nem történik gyártás vagy mozgatás kanban nélkül
- Hibás termékek nem mennek tovább a következő állomásra
- A kanbanok számát óvatosan csökkenteni kell a kisebb készlet elérésének érdekében és az esetleges rendszerhibák felfedése miatt. (Skarrin, 2015)

## 4. Kanban alapjai a szoftverfejlesztésben

Ahogy az a korábbiakban már említésre került, a kanban módszer olyan sikeresnek bizonyult a termelés területén, hogy idővel több iparág is elkezdte alkalmazni. A mai világban az egyik leggyorsabban fejlődő terület az IT területe, így természetesen itt is próbálnak működő újításokat bevezetni a projekt sikerességének érdekében, ehhez pedig hozzátartozik a módszertanokkal való haladás is. A kanban egy jó jelöltnek számított a korábbi tapasztalatok alapján, így az IT területeken is előszeretettel próbálták ki. A módszer megfelelő alkalmazása elősegítette a projektcélokat, így az egyik legnépszerűbb módszer lett azóta a projektek életében.

### 4.1. Agilis szoftverfejlesztés

A projektmunkák nagy eltérést mutathatnak bizonytalanságukat tekintve, a könnyen meghatározótól terjedhet a teljesen bizonytalan munkáig. Az előbbihez egyértelmű elvárások tartoznak, ami mindenki számára tiszta és előre meghatározott. Ezek általában olyan projektek, amikhez hasonlóak már korábban sikeresnek bizonyultak. A teljesen bizonytalan projektek általában valami újnak a tervezéséhez kapcsolódnak, ahol kisebb a tapasztalat, nem tisztázottak az elvárások, vagy csak „high-level” elvárások ismertek. Ezekben az esetekben jelentős a változás, a bonyolultság és a kockázat. Ezek a körülmények nehézséget jelentenek az előrejelzésekben, becslésekben, így agilis megközelítést hoztak létre a projekt menedzselésére. Ez segít megvizsgálni a rövid ciklusú, gyors kialakítás megvizsgálhatóságát, emellett pedig értékelés és visszajelzés-alapú. (ProjectManagementInstitute, 2018)

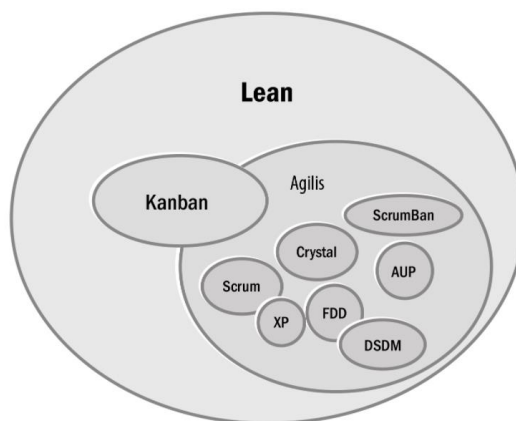
Az agilis szoftverfejlesztés rendkívül népszerűvé vált, mivel ennek segítségével jobb szoftvert fejleszthetnek, jobban tudnak együtt dolgozni a csapatok és a csapattagok, nő az ügyfélelégedettség és egy kellemesebb és élvezhetőbb munkakörnyezet kialakítását támogatja. Így mind a vállalat és a munkavállalók számára is számos előnnyel szolgál. Természetesen, a kérdés adott, hogy hogyan is lehet mindezt elérni és hogyan használjuk az agilis szoftverfejlesztés eszközeit megfelelően ahhoz, hogy ilyen eredményeket érjünk el.

Az agilitás tulajdonképpen metódusok és metodológiák gyűjteménye, ami segít a csapatnak abban, hogy hatékonyabban dolgozzanak és gondolkozzanak, Ezek magukba foglalják a szoftverfejlesztés minden területét, például a projektmenedzsmentet, szoftverdesign-t, architektúrát és folyamat fejlesztéseket. Az agilitás egy gondolkodásmód is, ami segít az embereknek az egymás közötti információ áramlásában, így képesek lesznek fontos döntéseket meghozni együtt. Ez azért is hasznos, mert így nincs szükség arra, hogy egy menedzser egyedül hozza meg ezeket a döntéseket, hanem a csapat képes önállóan is működni. Csapatok, amik sikeresen alkalmazkodtak az agilitáshoz, sokkal hatékonyabban fejlesztik le a termékeket, magasabb minőséget biztosítanak és mindezt gyorsabban elérik, mint az agilitás bevezetése előtt. (Greene-Stellman, 2014)

Az agilitás elvei közé tartoznak például az ügyfélelégedettség elérése a szoftver leszállítási határidejének betartásával és ezen belül is a folyamatos szállítással. Az agilis szoftverfejlesztési modellek elfogadják és hatékonyan le tudják kezelni a követelmények változását (még a fejlesztés végefelé is). A terméket nem egyben szállítják, hanem kisebb darabokban (például funkcióként). Támogatja (és elvárja) az üzleti szakemberek és a szoftverfejlesztők/tesztelők szoros kooperációját. Ezek mellett, elengedhetetlen az ütemtervek tartása, folyamatos információáramlás, egyszerűség, folyamatos fejlődés stb. A 6. ábra az agilis és kanban módszert a lean részeként mutatja, mivel a lean gondolkodási elveket követik, mint például:

- értékre összpontosítás
- kis tételszámok
- selejt kiküszöbölés

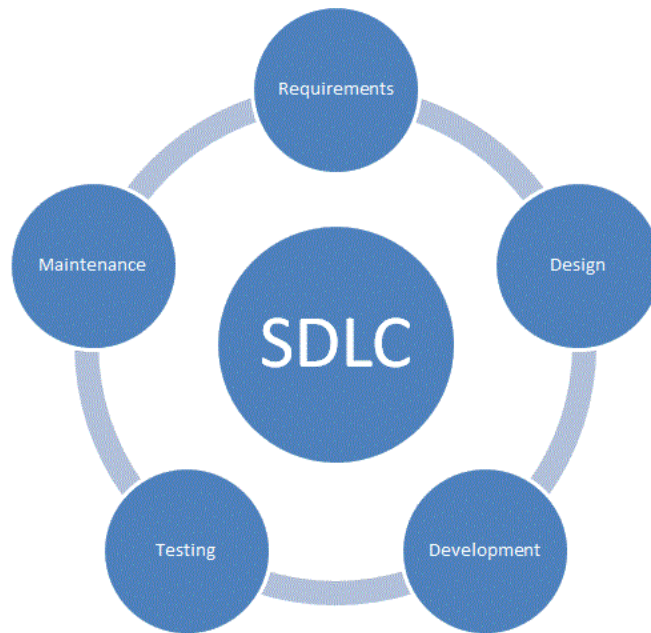
(ProjectManagementInstitute, 2018)



6. ábra: Agilis megközelítések (ProjectManagementInstitute, 2018)

A 6. ábrán látható, hogy az agilis és a kanban módszer a lean gondolkodásmód leszármazottja. A kanban a lean gyártási környezetben alakult ki, de agilis környezetben is használják. “Ez a közös örökség nagyon hasonló, és az értékteremtésre, az emberek tiszteletére, a selejt minimalizálására, az átláthatóságra, a változáshoz való alkalmazkodásra és a folyamatos fejlődésre összpontosít.” (ProjectManagementInstitute, 2018, 5.o.)

Az agilis szoftverfejlesztés során iterációkban gondolkozunk, ami az SDLC-n (Software Development Life-Cycle) keresztül valósul meg.



7. ábra: Software Development Life-Cycle (GuideForSoftwareTesting, 2014)

A 7. ábra szemlélteti az SDLC folyamatát. Az iterációk a következő elemekből állnak:

- Követelmények elemzése
- Design
- Fejlesztés
- Tesztelés
- Karbantartás

Az agilis módszertanok tulajdonképpen egy keretrendszert biztosítanak a csapatoknak, ami fel van készítve a folyamatos változásokra, támogatja a csapatok önálló működését és a folyamatos feedback-eket tartják a siker egyik kulcsának.

## 4.2. Kanban

Az agilis szoftverfejlesztés egyik legnépszerűbb metodológiája a kanban. A vizuális projektmenedzsment eszközök jelentősen meg tudják könnyíteni a munkát és a projektcélok elérését. A kanban segítségével az információk minden érintett számára könnyen láthatóvá tehetőek, így hatékonyan tudnak együttműködni a csapattagok. Az információ áramlását az úgynevezett „kanban board” segíti elő.

A kanban bevezetésével a következő előnyök érhetőek el:

- **Idő megtakarítása:** A kanban board által a csapattagoknak lehetősége van megosztani minden projekttel kapcsolatos releváns információt egy helyen. Mivel rendkívül egyszerűen elérhető, így számos felesleges meeting vagy e-mail váltás elkerülhető.
- **Gyorsabb problémamegoldás:** Egy jól működő kanban munkafolyamattal könnyű felismerni, ha egy feladat nem halad az elvárt tempóban. Ez segít minél hamarabb meghatározni a hibákat, blokkereket stb.
- **Hatékonyság növelése:** A kanban segít csökkenteni az erőforrás-veszteséget. Emellett, segít abban, hogy a csapat tagjai egyszerre csak egy feladatra koncentráljanak, feladatról-feladatra ugrálás helyett.
- **Növeli az egyének szerepét:** a csapattagok meghozhatják saját döntéseiket és szerepelhetnek „vezetőként” egyes helyzetekben. A kanban board pedig lehetővé teszi, hogy saját munkájukat a körülmények szerint szervezzék.
- **Fejleszti a csapat teljesítményét:** A kanban a folyamatos fejlődésre biztatja a csapattagokat.
- **A csapat teljesítményének folyamatos növekedése egy jobb termékhez és elégedettebb ügyfélhez vezet.** (Sammons, 2020)

A kanbannak a logisztikában és informatikában történő felhasználása között számos hasonlóságot lehet felfedezni, amire példa található az 1. táblázatban:

| KANBAN                          | LOGISZTIKA                       | INFORMATIKA                                                          |
|---------------------------------|----------------------------------|----------------------------------------------------------------------|
| Push/Pull                       | Pull                             | Pull                                                                 |
| Eszköz, információáramlás       | Információs kártya               | Információs kártya, menedzsment eszközök (pl. JIRA)                  |
| Munka szabályozása              | kártyával                        | WIP limit, de ez opcionális, lehet limit nélkül is kanbant használni |
| Termék eljuttatása az ügyfélhez | Kész terméket kap                | A terméket „darabokban” szállítjuk (verziók)                         |
| Stratégia                       | JIT, kaizen                      | JIT, kaizen                                                          |
| Mérőszámok                      | átfutási idő, ciklusidő, taktidő | átfutási idő, ciklusidő, WIP                                         |

*1. táblázat: Kanban - logisztika és informatika*

## 5. Egyéni rész bevezető

A dolgozat egyéni alkalmazás részében a kanban informatikai területen történő használatát szeretném bemutatni, ezen belül is a szoftverfejlesztést kiemelve.

Ezen a területen nem fizikai termékek áramlásáról van szó, hanem szellemi termékek gyártásáról, viszont ugyanúgy le kell „szállítani” ezeket a termékeket is az ügyfélnek, amennyiben ez egy megrendelés. Ahogy az már az előző fejezetben kifejtésre került, lényeges különbség, hogy szoftverek esetében általában nem feltétlen akkor kerül át a megrendelőhöz a termék, amikor teljesen kész van, hanem az ügyfélnek bizonyos időközönként rálátása van, hogy hogyan áll a fejlesztés. Tehát az ügyfél tulajdonképpen kisebb darabokban kapja meg a szoftvert, mindig hozzátéve új funkciókat. A szoftverek fejlesztési ideje terjedhet pár hónapos időciklustól akár több évig is, sőt nagyobb projektek esetén akár 10 éves szerződésekről is előfordulhatnak – a folyamatos karbantartási munkálatok és esetleges új funkciók megvalósítása miatt -. Illetve, ez az időtartam attól is függ, hogy egy már meglévő rendszert kell továbbfejleszteni vagy az elejéről kell felépíteni.

Az előző bekezdésben ábrázolt helyzetben létezik egy megrendelő és egy szoftverfejlesztő cég. Ilyenkor általában a folyamat úgy megy végbe, hogy az ügyfél megosztja a szoftverfejlesztő céggel, hogy milyen terméket szeretne (például pénzügyi folyamatokat lebonyolító alkalmazást, sportszert árusító webáruházat, marketingcég weboldalát stb). A részletek tisztázása és rögzítése után elindulhat a csapat(ok) felépítése, szerepek meghatározása és megkezdődhet a szoftver fejlesztése.

Lényeges pont, amikor a menedzsment meghatározza, hogy milyen módszertan alapján valósuljon meg a fejlesztés és a kanban egy rendkívül gyakran választott módszer. Természetesen rengeteg tényezőtől függ, hogy melyik módszer lehet a legalkalmasabb az adott projekt esetén, ennek elemzése a későbbi fejezetekben kerül előtérbe.

## 6. Esettanulmány ismertető

A diplomadolgozat gyakorlati része esetpéldákon keresztül kerül bemutatásra. Az esettanulmány keretein belül egy funkció lefejlesztésének folyamata kerül elemzésre, ami kanban módszer által valósul meg.

A diplomadolgozatban szereplő fiktív vállalat online szerencsejáték weboldalak fejlesztésével és üzemeltetésével foglalkozik. A vállalat célja, hogy a piac minél nagyobb részét lefedjék folyamatosan megújuló és bővülő oldalakkal. A vállalat számtalan online gaming adta lehetőséget rejt magában, így casino, sportfogadás és live casino lehetőségek is adottak az oldalakon.

A vállalat egy fejlesztés alatt lévő szerencsejáték oldalukra egy új funkció bevezetését tűzte ki célul. Az új funkció teljesítésére 3 hét áll rendelkezésre, így addig szükséges és elengedhetetlen minden tervezett rész-projekt megvalósítása, hogy az újonnan debütáló online gaming oldal megfelelően működjön a felhasználók számára.

A diplomadolgozatban elemzett projekt a szóban forgó weboldal egy új funkciójának implementálását vizsgálja kanban módszert követve. A funkció neve a „FreeGame”, ami tulajdonképpen regisztráció nélküli játékot enged a felhasználóknak. Természetesen ezzel a játékmóddal nem lehet pénzt nyerni, de a felhasználó számos játékot kipróbálhat tét nélkül és eldöntheti, hogy szeretne-e regisztrálni az oldalra, ezzel megkezdve a tényleges játszási folyamatot. Az ingyenes játék lehetőségével a felhasználónak nem kell feleslegesen időt töltenie a regisztrációs folyamattal csupán azért, hogy pár játékot kipróbáljon. A kipróbált játékok alapján a felhasználó szabadon dönthet, hogy regisztrál-e a weboldalra.

A projekt célja a felhasználó-bázis növelése, aminek eredményeképp profit-növekedés is várható.

## 7. Kanban

### 7.1. Folyamat, szerepkörök

A folyamatok megértéséhez fontos ismertetni a különböző szerepköröket, ezek közül a legjellemzőbbek a következők:

- Termékfelelős (product owner): A termékfelelős határozza meg, hogy mit kell kifejleszteni az ügyfél igények alapján. Ő állítja össze az erre vonatkozó listát.
- Fejlesztési csapat (development team): A fejlesztők feladata, hogy fejlesszék az adott funkciókat vagy javítsák a hibákat
- Tesztelési csapat (quality analyst/testing team): A tesztelők felelőssége, hogy a funkciók a megfelelő minőségben legyenek kiadva
- Ügyfél (customer): az ügyfél a megrendelő

A backlogban szerepelnek azok feladatok, amik arra várnak, hogy elkészüljenek. Feladat lehet például egy hiba kijavítása vagy akár egy teljesen új funkció kifejlesztése. Ez egy prioritizált lista, tehát minden tételhez tartozik egy prioritás, ami által könnyen látható, hogy melyik a legfontosabb.

A húzó rendszernek itt is fontos szerepe van. Célja, hogy a backlog-ban szereplő tételeket minél hamarabb feldolgozzuk:



8. ábra: Backlog feldolgozása - PULL

### Esetpélda1

A FreeGame funkció esetén a backlog összeállítása a követelmények rögzítésével kezdődik. A funkciót kisebb feladatokra bontottam, amik alapján könnyebben tervezhető és megvalósítható. A kisebb részfeladatokat story-nak nevezzük, az ezeket összefogó nagy elemet pedig epic-nek.

**Epic:** FreeGame funkció

**Story:** A részfeladatokat a 2. táblázat tartalmazza

| Story/Részfeladat                  | Leírás/követelmény                                                                                                                                                                                                                                                                                                           | Előfeltétel | Prioritás |
|------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|-----------|
| <b>1:<br/>FreeGame blokk</b>       | <p>A honlapon belül új menüpont létrehozása “FreeGame” névvel. Ha a felhasználó rákattint, megnyílik a FreeGame oldal.</p> <p>Alkategóriák létrehozása a FreeGame oldalon belül.</p> <p>Kategóriák:</p> <ul style="list-style-type: none"> <li>• Népszerű játékok</li> <li>• Új játékok</li> <li>• Casino játékok</li> </ul> | -           | magas     |
| <b>2:<br/>FreeGame játékkeret</b>  | Az elindított játék egy játékkeretbe betöltve indul el. A keret különböző gombokat tartalmaz: átugrás a fizetős módra, kilépés stb. Ezen kívül játékpénz is megjelenítésre kerül.                                                                                                                                            | 1           | magas     |
| <b>3:<br/>„Mutasd mindet” gomb</b> | A kategóriáknál a “Mutasd mindet” gomb teszi lehetővé, hogy egy új lapon megnyitva az adott kategória összes játékát lássuk.                                                                                                                                                                                                 | 1           | közepes   |
| <b>4:<br/>Lapozás</b>              | A kategóriáknál egyszerre csak 5 játék látszódik. A lapozás funkciót egy nyíl teszi lehetővé, ami akkor jelenik meg, ha több, mint 5 játék van feltöltve az adott kategóriában.                                                                                                                                              | 1           | közepes   |

|                                                |                                                                                                                                                                           |   |          |
|------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|----------|
| <b>5:</b><br><b>SEO ellenőrzés -<br/>teszt</b> | A SEO szövegek hozzáadása a SEO csapat feladata, így ebben az esetben csak ellenőrzésre van szükség a tesztelői csapattól.                                                | 1 | közepes  |
| <b>6:</b><br><b>Rendezés</b>                   | Kategóriák játéklistának rendezése: <ul style="list-style-type: none"> <li>• Népszerű játék: népszerűség alapján</li> <li>• Új játék/Casino játék: ABC sorrend</li> </ul> | 1 | alacsony |

2. táblázat: FreeGame funkció - backlog elemei

A 2. táblázatban szereplő részfeladatok alkotják a backlog-ot, prioritás szerint rendezve. Továbbá látható az egyes részfeladatok előfeltételei. Ez azt a tevékenységet vagy tevékenységeket jelenti, amiknek el kell készülniük, mielőtt az adott story fejlesztése elkezdődne. Látható, hogy a FreeGame blokk megvalósítása nélkül a többi rész-funkció sem készülhet el.

A funkció kiadása éles rendszerre (amit a valódi felhasználók is látnak már) egyben történik meg (tehát mind a 6 story-t tartalmaznia kell), aminek előfeltétele, hogy minden FreeGame epic-hez tartozó story eljusson a Kész állapotba. Az éles rendszerre történő kiadás előtti munkálatok fejlesztői és tesztelői környezetben történnek.

## 7.2. Kanban tábla alkalmazása

A kanban egyik sajátossága és egyben nagy előnye a vizuális megvalósítás. A csapat és a vezetőség tagjai számára rendkívül praktikus, ha mindenki pontosan tudja, hogy mi történik körülötte és egyszerűen követni tudja. A menedzsmentnek szintén az az elvárása, hogy egyértelműen és gyorsan látható legyen minden feladat jelenlegi státusza, becslése, hátralévő idő és így tovább.

A fent leírt igények kielégítésére tökéletesen szolgál a kanban tábla. Ennek többféle megvalósítása létezik.

### 1. A tényleges fizikai tábla

Ebben az esetben, egy táblát helyeznek el a csapat közelében, ahol post-it jelzi az egyes feladatokat. A papírokon alap információk szerepelnek, például:

- Feladat: nagyon rövid cím, de mégis egyértelműen beazonosítható általa az adott feladat
- Prioritás: magas, közepes, alacsony
- Felelős/illetékes: kinek a felelőssége az adott feladat elvégzése, ki dolgozik éppen rajta
- Színek:
  - Jelentheti a prioritást: így természetesen ez az információ leghagyható a kártyáról, mert a szín által egyből beazonosításra kerül
  - Felelős: akár mindenkinek a csapatban lehet egyéni színe, így ebben az esetben nem kell feltüntetni a nevet a kártyán
- Oszlopok:
  - Az oszlopok a különböző státuszokat mutatják, könnyen beazonosítható általuk, hogy az adott feladat milyen fázisban van éppen.

A kanban nagyon rugalmas, így itt sincsenek fix megkötések. Mind a státuszok, a post-it-en lévő tartalmak, színek projektre szabhatóak.

2. A második és egyben modernebb megközelítése a vizuális szemléltetésnek, ha valamilyen projekt menedzsment eszközt alkalmazunk. A leggyakrabban használt a JIRA. A JIRA-ban lehetőségünk van létrehozni a csapat kanban tábláját (JIRA kanban board-ként is lehet rá hivatkozni). Célja, működési elve és felhasználási módja megegyezik a fizikai táblával.

Nyilvánvalóan megfelelő jogosultsággal a board-ot teljes mértékben a projektre és csapatra lehet szabni, így a csapat preferenciáján múlik, hogy mi és hogyan kerül megjelenítésre.

A JIRA-hoz hasonló menedzsment eszközök még számos előnyt hordoznak magukban, amik segítségünkre lehetnek a projekt haladásában, folyamatok fejlesztésében, riportok készítésében stb. Segítségükkel könnyebben nyomon követhetőek a verziók is, illetve a feladatok közötti kapcsolatok és függőségek is egyértelműen láthatóak.

Jellemző, hogy több csapat is együttműködik a projektcél megvalósításának érdekében, ilyenkor több kanban tábla is létrehozható, ami által mindegyik csapat a saját feladatait tudja menedzselni.

Ahogy a korábbiakban említésre került, a kanban lehetővé teszi, hogy viszonylag egyszerűen és kevés limitációval koordináljuk a folyamatokat. A backlog listája bármikor kibővítésre kerülhet további feladatok hozzáadásával. Amennyiben nincs limitáció megadva, úgy bármennyi feladat lehet egyszerre folyamatban akármelyik státuszban. A feladatok jelenthetnek például egy új funkció lefejlesztését vagy egy hiba kijavítását.

## Esetpélda2

A FreeGame funkció részfeladatai a 9. ábrán látható kanban táblán kerültek elhelyezésre. Az aktuális kanban tábla a fenti feladatokkal tehát a következőképpen alakul:

| Teljesítendő (6)                                             | Fejlesztés alatt (0) | Tesztelésre vár (0) | Tesztelés alatt (0) | Kész (0) |
|--------------------------------------------------------------|----------------------|---------------------|---------------------|----------|
| FreeGame blokk<br>Illetékes: -<br>Prioritás: magas           |                      |                     |                     |          |
| FreeGame játékkeret<br>Illetékes: -<br>Prioritás: magas      |                      |                     |                     |          |
| Lapozás<br>Illetékes: -<br>Prioritás: közepes                |                      |                     |                     |          |
| "Mutasd mindet" gomb<br>Illetékes: -<br>Prioritás: közepes   |                      |                     |                     |          |
| SEO ellenőrzés - teszt<br>Illetékes: -<br>Prioritás: közepes |                      |                     |                     |          |
| Rendezés<br>Illetékes: -<br>Prioritás: alacsony              |                      |                     |                     |          |

9. ábra: FreeGame kanban tábla

A kanban táblát a következő oszlopok alapján határoztam meg:

- Teljesítendő (backlog): teljesítendő feladatok listáját jelöli
- Fejlesztés alatt: fejlesztés folyamatban
- Tesztelésre vár: tesztelésre váró feladatok
- Tesztelés alatt: tesztelés folyamatban
- Kész: a feladat lefejlesztve, letesztelve

A kanban táblán továbbá megfigyelhető az oszlopnevek mellett zárójelben az adott oszlopban jelenleg lévő feladatok száma.

Az ábrán látható a kiinduló állapot, ahol az összes FreeGame epic alá tartozó részfeladat a Teljesítendő oszlop elemeit alkotják.

A kártyákra felkerült a story neve, az illetékes és a prioritás. A Teljesítendő és Tesztelésre váró oszlopokban lévő elemeken jelenleg nem dolgozik senki, a Kész oszlopban pedig már nem, így az illetékes ezekben az esetekben üres.

A prioritás színkódokkal is jelölve van az egyszerűbb beazonosítás végett:

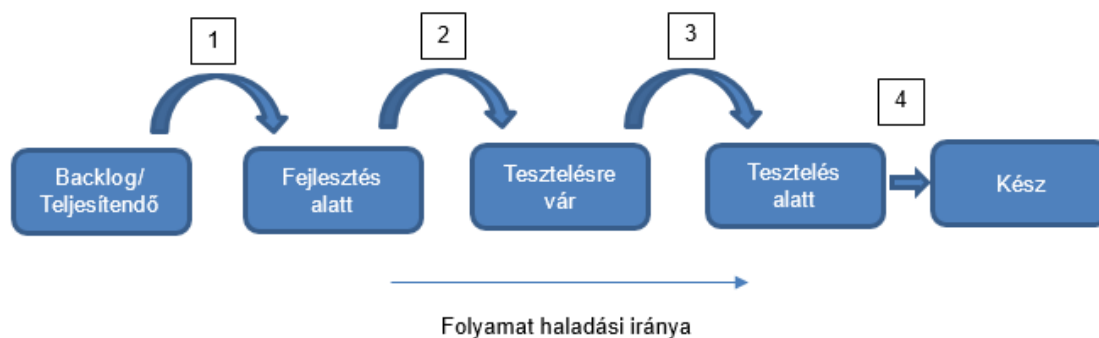
- Piros: magas
- Sárga: közepes
- Kék: alacsony

A feladatok továbbhaladásának sorrendjénél a prioritást érdemes alapul venni, viszont egyéb limitáció ebben az esetben nincs. A kanban ezen verziója nem szabja meg, hogy egyes oszlopokban hány tétel lehet.

A teljesítendő oszlop sajátossága kanban rendszer esetén, hogy nem egy fix lista, hanem bármikor kerülhet be új tétel. A backlog teljes listáját adja a teljesítendő oszlop, tehát a csapat tagjai számára láthatóvá teszi az összes teljesítendő feladatot.

## 7.3. Kanban kártyák mozgása

A feladatok oszlopról-oszloptra haladása akkor lehetséges, ha van rá kapacitás. A feladatok ideális esetben balról jobbra haladnak, amiből észrevehető, hogy a PULL rendszer itt is érvényesül.



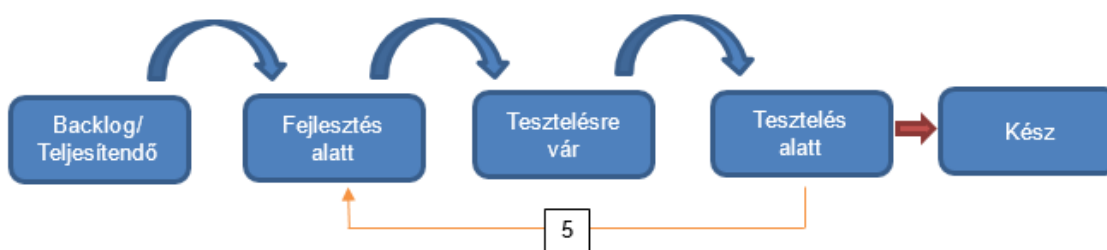
10. ábra: Státuszok – PULL

Az 10. ábrán látható az 1-3. folyamatok közötti pull rendszer.

- 1-es folyamat: abban az esetben, ha szabadul erőforrás a fejlesztői csapatban, a teljesítendő tételek oszlopából áthúzható újabb elem
- 2-es folyamat: a tesztelésre váró oszlopba a fejlesztés oszlopából kerülnek át azok az elemek, amelyek fejlesztői szemszögből késznek nyilváníthatóak és validálásra várnak a tesztelők részéről
- 3-as folyamat: a tesztelők a tesztelésre váró feladatokból tudnak áthúzni újat a „Tesztelés alatt” lévő oszlopba, amennyiben van rá kapacitás

A 4-es folyamatban lezárásra kerülnek azok a feladatok, melyek a tesztelés folyamán pozitív eredménnyel végződtek, tehát azok a feladatok, amik a rögzített elvárásoknak megfelelően lettek lefejezve és sikeresen átmentek a tesztelés fázison.

Ugyanakkor, lehetséges a folyamatban visszafele áramlás is. Tegyük fel, hogy a tesztelés nem hoz pozitív eredmény, így az adott feladatot nem lehet lezárni. Ebben az esetben, a kártya visszakerül a fejlesztői csapathoz javításra (11.ábra).

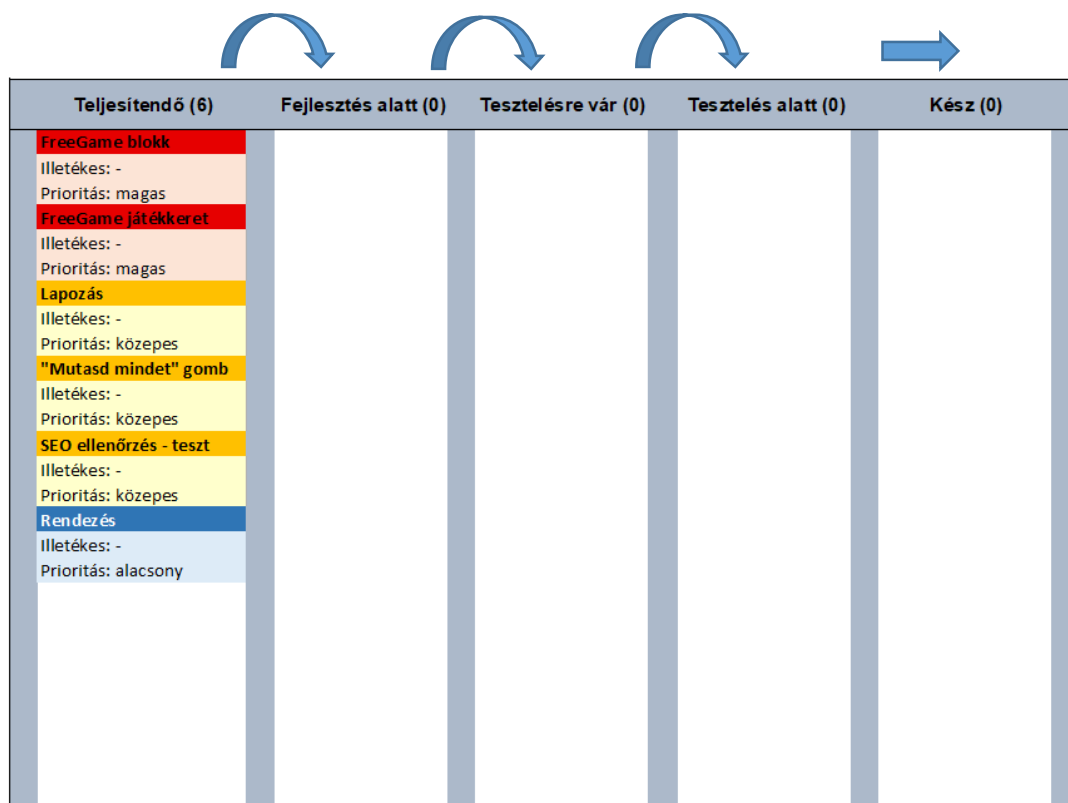


11. ábra: Státuszok 2.

A fent látható 11. ábrán az 5. folyamat jelzése mutatja a kártya visszafele áramlását. Ebben az esetben az adott kártyának újból végig kell mennie a folyamatokon a fejlesztéstől a tesztelésig, egészen addig, míg nem lehetséges a sikeres lezárás.

### Esetpélda3

A gyakorlatban a FreeGame funkció részfeladatait tartalmazó kanban táblán végig nyomon követhető a kártyák mozgása:



12. ábra: FreeGame kanban – PULL

A kártyák haladásának részletes ismertetése a 7.4. fejezetben kerül elemzésre az esetpélda által. A folyamat ugyanaz lenne ennél az esetnél is, csak limitáció nélkül.

## 7.4. Kanban alkalmazása WIP limittel

Ahogy a korábbiakban említésre került, a kanbannak számos felhasználási módja van a gyakorlatban. Ezen fejezetben a WIP limittel történő megvalósítást szeretném bemutatni.

A WIP limit a „work in progress”-ből ered, aminek segítségével korlátozni tudjuk, hogy hány tétel lehet egy adott oszlopban, tehát WIP limitet rendelhetünk azokhoz az oszlopokhoz, ahol a folyamatban lévő feladatokat limitálni szeretnénk. A limitálás előnye, hogy arra ösztönzi a csapatot, hogy a feladatokat minél hamarabb befejezzék annak érdekében, hogy újat tudjanak felvenni. Továbbá, nem előnyös, ha nagyon sok elem áll egy adott státuszban. Ezekben az esetekben előfordulhat, hogy megfelelnek párról vagy a

kevésbé vonzó feladatok teljesítését túl sokáig elhúzzák. Ösztönző lehet a WIP limit bevezetése, mivel kevésbé oszlik meg a fókusz. Projekt menedzsment eszközök használata esetén (pl. JIRA) a WIP limit túllépésével az oszlop jellemzően pirosra színeződik, ezzel felhívva a figyelmet a feladatok túlesordulására. Ez jelzés a csapatnak arra vonatkozóan, hogy szükséges valamiféle közbelépés, hogy visszaállítsák a meghatározott egyensúlyt. A WIP limit tapasztalat alapján bármikor megváltoztatható, kivonható vagy akár bevezethető további oszlopokra.

A limit általában tapasztalat alapján alakul ki és nagyban függ a csapat létszámától és összetételétől. Ennek következtében, ha például a csapat létszáma vagy összetétele megváltozik, akkor ez magával húzza a WIP limitek változását is.

A következő esetpéldából (esetpélda4) látható, hogy hogyan kerülnek megvalósításra a feladatok. Az első lépés, amikor felmerül az igény egy feladat megvalósítására és ez által rögzítésre kerül – bekerül a backlogba -, majd ez a feladat az előre definiált státuszokon keresztül eljut a megfelelő csapatokhoz, amíg kész nem lesz. A folyamat alatt a prioritizálási értékek figyelembe vétele és megfelelő kezelése elengedhetetlen.

A WIP limitnek egy lehetséges felhasználási módja került szemléltetésre, ami talán a leggyakoribbnak is tekinthető.

Ahogy a példákon is látható, kanban esetén a feladatok fejlesztése/tesztelése folyamatos, illetve a funkciók vagy javítások kiadása –“release”- is. A módszertan lehetővé teszi és egyben jól kezeli a változtatásokat, az éles rendszerre történő kiadások nincsenek keretek közé szorítva, ez is adja a rendszer rugalmasságát.

### **Esetpélda4**

A FreeGame funkció fejlesztése esetén WIP limit a folyamatban lévő elemek oszlopaira került bevezetésre, így a korábbi FreeGame kanban táblát két helyen egészítettem ki korlátozással (13. ábra):

- a fejlesztés alatt lévő elemek oszlopa
- és a tesztelési folyamatban szereplő tételek oszlopa

| Teljesítendő (6)                                             | Fejlesztés alatt (0)<br>WIP limit: 3 | Tesztelésre vár (0) | Tesztelés alatt (0)<br>WIP limit: 2 | Kész (0) |
|--------------------------------------------------------------|--------------------------------------|---------------------|-------------------------------------|----------|
| FreeGame blokk<br>Illetékes: -<br>Prioritás: magas           |                                      |                     |                                     |          |
| FreeGame játékkeret<br>Illetékes: -<br>Prioritás: magas      |                                      |                     |                                     |          |
| Lapozás<br>Illetékes: -<br>Prioritás: közepes                |                                      |                     |                                     |          |
| "Mutasd mindet" gomb<br>Illetékes: -<br>Prioritás: közepes   |                                      |                     |                                     |          |
| SEO ellenőrzés - teszt<br>Illetékes: -<br>Prioritás: közepes |                                      |                     |                                     |          |
| Rendezés<br>Illetékes: -<br>Prioritás: alacsony              |                                      |                     |                                     |          |

13. ábra: FreeGame kanban tábla WIP limittel

A „Teljesítendő”, „Tesztelésre vár” és „Kész” oszlopokra nem került limit bevezetésre, mivel ezek nem a folyamatban lévő feladatokat tartalmazzák.

A fejlesztési és tesztelési csapat limitje a csapatlétszám alapján:

A fejlesztési csapat limitje: 3

A tesztelési csapat limitje: 2

A limit mind az oszlop nevénél, mind határcsíkkal jelezve van a kanban táblán.

A fejlesztés folyamatának megkezdésénél figyelembe kell venni a limitet és a prioritási értékeket. Mivel a fejlesztési csapat WIP limitje 3, így maximum 3 tételen dolgozhatnak egyszerre. A csapat az előfeltételeket figyelembe véve először csak a FreeGame blokk fejlesztésével tud elkezdni foglalkozni, mivel a többi story megkezdéséhez szükség van a blokk befejezésére.

| Teljesítendő           | Fejlesztés alatt<br>WIP limit: 3 | Tesztelésre vár | Tesztelés alatt<br>WIP limit: 2 | Kész |
|------------------------|----------------------------------|-----------------|---------------------------------|------|
| FreeGame játékkeret    | FreeGame blokk                   |                 |                                 |      |
| Lapozás                |                                  |                 |                                 |      |
| “Mutasd mindet” gomb   |                                  |                 |                                 |      |
| SEO ellenőrzés - teszt |                                  |                 |                                 |      |
| Rendezés               |                                  |                 |                                 |      |
|                        |                                  |                 |                                 |      |
|                        |                                  |                 |                                 |      |
|                        |                                  |                 |                                 |      |

3. táblázat: FreeGame fejlesztési folyamat 1. lépés

A 3. táblázaton megfigyelhető, hogy a FreeGame blokk story átkerült a Fejlesztés alatt lévő elemek oszlopába. A story lefejlesztése után bekerül a Tesztelésre váró elemekhez (4. táblázat).

| Teljesítendő           | Fejlesztés alatt<br>WIP limit: 3 | Tesztelésre vár | Tesztelés alatt<br>WIP limit: 2 | Kész |
|------------------------|----------------------------------|-----------------|---------------------------------|------|
| FreeGame játékkeret    |                                  | FreeGame blokk  |                                 |      |
| Lapozás                |                                  |                 |                                 |      |
| “Mutasd mindet” gomb   |                                  |                 |                                 |      |
| SEO ellenőrzés - teszt |                                  |                 |                                 |      |
| Rendezés               |                                  |                 |                                 |      |
|                        |                                  |                 |                                 |      |
|                        |                                  |                 |                                 |      |
|                        |                                  |                 |                                 |      |

4. táblázat: FreeGame fejlesztési folyamat 2. lépés

A folyamat viszonylag gyorsan tud haladni ezen a ponton, mivel a WIP limit sehol se akasztja meg, illetve nagyobb a kapacitás, mint amit ez az egy feladat megkövetel. A FreeGame blokk story-t egyből elkezdheti tesztelni a tesztelői csapat, tehát nem kell várakoznia a „Tesztelésre vár” oszlopban, amit az 5. táblázat szemléltet.

| Teljesítendő           | Fejlesztés alatt<br>WIP limit: 3 | Tesztelésre vár | Tesztelés alatt<br>WIP limit: 2 | Kész |
|------------------------|----------------------------------|-----------------|---------------------------------|------|
| FreeGame játékkeret    |                                  |                 | FreeGame blokk                  |      |
| Lapozás                |                                  |                 |                                 |      |
| “Mutasd mindet” gomb   |                                  |                 |                                 |      |
| SEO ellenőrzés - teszt |                                  |                 |                                 |      |
| Rendezés               |                                  |                 |                                 |      |
|                        |                                  |                 |                                 |      |
|                        |                                  |                 |                                 |      |
|                        |                                  |                 |                                 |      |

5. táblázat: FreeGame fejlesztési folyamat 3. lépés

A FreeGame blokk tesztelés után átkerül a Kész oszlopba, ami lehetővé teszi, hogy további tételeken kezdjünk el dolgozni a backlog-ból.

A következő tételek kiválasztása továbbra is a prioritás és előfeltételek alapján történt. Az előfeltétel teljesült az összes fennmaradó story-ra, így prioritás alapján a FreeGame

játékkeret egyértelműen bekerült a fejlesztendő tételek alá. A WIP limit korlátozásai miatt még két elem választható a listából. A közepes prioritási tételek a következők a sorban. Alapvetően bármelyik lehetne a három közül, viszont a SEO ellenőrzés storyt a SEO csapat fogja végrehajtani és csak tesztelendő feladat van ennél. A SEO story még nem készült el ezen a ponton, így a Lapozás és a „Mutasd mindet” gomb lefejlesztése a sorban következők:

| Teljesítendő           | Fejlesztés alatt<br>WIP limit: 3 | Tesztelésre vár | Tesztelés alatt<br>WIP limit: 2 | Kész           |
|------------------------|----------------------------------|-----------------|---------------------------------|----------------|
| SEO ellenőrzés - teszt | FreeGame játékkeret              |                 |                                 | FreeGame blokk |
| Rendezés               | Lapozás                          |                 |                                 |                |
|                        | „Mutasd mindet” gomb             |                 |                                 |                |
|                        |                                  |                 |                                 |                |
|                        |                                  |                 |                                 |                |
|                        |                                  |                 |                                 |                |
|                        |                                  |                 |                                 |                |

6. táblázat: FreeGame fejlesztési folyamat 4. lépés

A FreeGame játékkeret fejlesztése több időt vesz igénybe, mint a másik kettő. A Lapozás és a „Mutasd mindet” gomb rövid idő után átkerül a Tesztelésre vár oszlopba. A fejlesztés pedig felveszi a Rendezés feladatot (7. táblázat).

| Teljesítendő           | Fejlesztés alatt<br>WIP limit: 3 | Tesztelésre vár      | Tesztelés alatt<br>WIP limit: 2 | Kész           |
|------------------------|----------------------------------|----------------------|---------------------------------|----------------|
| SEO ellenőrzés - teszt | FreeGame játékkeret              | Lapozás              |                                 | FreeGame blokk |
|                        | Rendezés                         | „Mutasd mindet” gomb |                                 |                |
|                        |                                  |                      |                                 |                |
|                        |                                  |                      |                                 |                |
|                        |                                  |                      |                                 |                |
|                        |                                  |                      |                                 |                |
|                        |                                  |                      |                                 |                |

7. táblázat: FreeGame fejlesztési folyamat 5. lépés

A 7. táblázatban megfigyelhető, hogy a Tesztelésre váró feladatok oszlopában pont annyi story váraozik, mint amennyin a tesztelői csapat egyszerre dolgozhat, így mindkettő átvehető, amit a 8. táblázat szemléltet. Időközben, egy új elem is bekerült a backlogba, a Regisztrációs pop-up.

| Teljesítendő           | Fejlesztés alatt<br>WIP limit: 3 | Tesztelésre vár | Tesztelés alatt<br>WIP limit: 2 | Kész           |
|------------------------|----------------------------------|-----------------|---------------------------------|----------------|
| SEO ellenőrzés - teszt | FreeGame játékkeret              |                 | Lapozás                         | FreeGame blokk |
| Regisztrációs pop-up   | Rendezés                         |                 | „Mutasd mindet” gomb            |                |
|                        |                                  |                 |                                 |                |
|                        |                                  |                 |                                 |                |
|                        |                                  |                 |                                 |                |
|                        |                                  |                 |                                 |                |
|                        |                                  |                 |                                 |                |

8. táblázat: FreeGame fejlesztési folyamat 6. lépés

Regisztrációs pop-up: Felmerült az igény, hogy a felhasználónak megjelenjen egy

regisztrációs pop-up. A story követelmény, hogy a pop-up 10 perc FreeGame után felugorjon, és megkérdezze a felhasználót, hogy szeretne-e regisztrálni. Amennyiben igen, átvizsgálja a regisztrációs oldalra, ha nem, akkor pedig tudja folytatni az ingyen játékot. Prioritás: medium

Időközben a SEO ellenőrzés – teszt story is elkészül a SEO csapat által, így az bekerül a tesztelésre váró elemek oszlopába. A Regisztrációs pop-up story-n pedig elkezd dolgozni a fejlesztői csapat (9. táblázat).

| Teljesítendő | Fejlesztés alatt<br>WIP limit: 3 | Tesztelésre vár        | Tesztelés alatt<br>WIP limit: 2 | Kész           |
|--------------|----------------------------------|------------------------|---------------------------------|----------------|
|              | FreeGame játékkeret              | SEO ellenőrzés - teszt | Lapozás                         | FreeGame blokk |
|              | Regisztrációs pop-up             |                        | "Mutasd mindet" gomb            |                |
|              | Rendezés                         |                        |                                 |                |
|              |                                  |                        |                                 |                |
|              |                                  |                        |                                 |                |
|              |                                  |                        |                                 |                |
|              |                                  |                        |                                 |                |

9. táblázat: FreeGame fejlesztési folyamat 7. lépés

A 9. ábrán az is megfigyelhető, hogy a backlogban szereplő tételek elfogytak, így mindegyik story vagy folyamatban van vagy kész. A tesztelésre váró SEO ellenőrzést még nem tudja elkezdni a tesztelői csapat a WIP limit elérése miatt.

A 10. táblázatban az látható, hogy a csapat sikeresen lezárja a Lapozás storyt, a „Mutasd mindet” gomb viszont nem felelt meg a leírt követelményeknek, így az visszakerül a fejlesztői csapathoz javításra. Az így felszabadult tesztelői csapat el tud kezdeni dolgozni a SEO ellenőrzési feladaton, a fejlesztői csapat viszont – a „Mutasd mindet” gomb visszamozgatásával - túlmént a meghatározott WIP limiten.

| Teljesítendő | Fejlesztés alatt<br>WIP limit: 3 | Tesztelésre vár | Tesztelés alatt<br>WIP limit: 2 | Kész           |
|--------------|----------------------------------|-----------------|---------------------------------|----------------|
|              | FreeGame játékkeret              |                 | SEO ellenőrzés - teszt          | FreeGame blokk |
|              | Regisztrációs pop-up             |                 |                                 | Lapozás        |
|              | Rendezés                         |                 |                                 |                |
|              | "Mutasd mindet" gomb             |                 |                                 |                |
|              |                                  |                 |                                 |                |
|              |                                  |                 |                                 |                |
|              |                                  |                 |                                 |                |

10. táblázat: FreeGame fejlesztési folyamat 8. lépés

A FreeGame játékkeret időközben elkészült, majd a Rendezés story is, így a fejlesztési oszlop újra a limit alá kerül (11. táblázat).

| Teljesítendő | Fejlesztés alatt<br>WIP limit: 3 | Tesztelésre vár     | Tesztelés alatt<br>WIP limit: 2 | Kész           |
|--------------|----------------------------------|---------------------|---------------------------------|----------------|
|              | Regisztrációs pop-up             | FreeGame játékkeret | SEO ellenőrzés - teszt          | FreeGame blokk |
|              | "Mutasd mindet" gomb             | Rendezés            |                                 | Lapozás        |
|              |                                  |                     |                                 |                |
|              |                                  |                     |                                 |                |
|              |                                  |                     |                                 |                |
|              |                                  |                     |                                 |                |
|              |                                  |                     |                                 |                |

11. táblázat: FreeGame fejlesztési folyamat 9. lépés

A tesztelési csapat a prioritás alapján elkezd dolgozni a FreeGame játékkereten is, és eközben befejeződik a Regisztrációs pop-up és a „Mutasd mindet” gomb fejlesztése is (12. táblázat).

| Teljesítendő | Fejlesztés alatt<br>WIP limit: 3 | Tesztelésre vár      | Tesztelés alatt<br>WIP limit: 2 | Kész           |
|--------------|----------------------------------|----------------------|---------------------------------|----------------|
|              |                                  | Regisztrációs pop-up | FreeGame játékkeret             | FreeGame blokk |
|              |                                  | "Mutasd mindet" gomb | SEO ellenőrzés - teszt          | Lapozás        |
|              |                                  | Rendezés             |                                 |                |
|              |                                  |                      |                                 |                |
|              |                                  |                      |                                 |                |
|              |                                  |                      |                                 |                |
|              |                                  |                      |                                 |                |

12. táblázat: FreeGame fejlesztési folyamat 10. lépés

A továbbiakban a tesztelői csapaton van a fókusz, akik prioritás alapján ellenőrzik a hátralévő story-kat. A FreeGame játékkeret és SEO ellenőrzés után a Regisztrációs pop-up és a „Mutasd mindet” gomb tesztelése következik (13. táblázat)

| Teljesítendő | Fejlesztés alatt<br>WIP limit: 3 | Tesztelésre vár | Tesztelés alatt<br>WIP limit: 2 | Kész                   |
|--------------|----------------------------------|-----------------|---------------------------------|------------------------|
|              |                                  | Rendezés        | Regisztrációs pop-up            | FreeGame blokk         |
|              |                                  |                 | "Mutasd mindet" gomb            | Lapozás                |
|              |                                  |                 |                                 | FreeGame játékkeret    |
|              |                                  |                 |                                 | SEO ellenőrzés - teszt |
|              |                                  |                 |                                 |                        |
|              |                                  |                 |                                 |                        |
|              |                                  |                 |                                 |                        |

13. táblázat: FreeGame fejlesztési folyamat 11. lépés

A 13. táblázatban megfigyelhető, hogy a Rendezés story még tesztelésre vár, viszont a tesztelői csapat WIP limitje nem engedi, hogy ezen a storyn elkezdjen dolgozni a csapat. A 14. táblázatban sikeresen lezárásra került a „Mutasd mindet” gomb story újratestelése, így ez átkerülhetett a Kész oszlopba és a Rendezés story pedig helyet kapott a tesztelői csapatnál.

| Teljesítendő | Fejlesztés alatt<br>WIP limit: 3 | Tesztelésre vár | Tesztelés alatt<br>WIP limit: 2 | Kész                   |
|--------------|----------------------------------|-----------------|---------------------------------|------------------------|
|              |                                  |                 | Regisztrációs pop-up            | FreeGame blokk         |
|              |                                  |                 | Rendezés                        | Lapozás                |
|              |                                  |                 |                                 | FreeGame játékkeret    |
|              |                                  |                 |                                 | SEO ellenőrzés - teszt |
|              |                                  |                 |                                 | "Mutasd mindet" gomb   |
|              |                                  |                 |                                 |                        |
|              |                                  |                 |                                 |                        |
|              |                                  |                 |                                 |                        |

14. táblázat: FreeGame fejlesztési folyamat 12. lépés

Végző lépésként a maradék kettő tesztelés alatt lévő story is sikeresnek bizonyult, így azok is átkerülnek a Kész oszlopba, amit a 15. táblázat szemléltet.

| Teljesítendő | Fejlesztés alatt<br>WIP limit: 3 | Tesztelésre vár | Tesztelés alatt<br>WIP limit: 2 | Kész                   |
|--------------|----------------------------------|-----------------|---------------------------------|------------------------|
|              |                                  |                 |                                 | FreeGame blokk         |
|              |                                  |                 |                                 | Lapozás                |
|              |                                  |                 |                                 | FreeGame játékkeret    |
|              |                                  |                 |                                 | SEO ellenőrzés - teszt |
|              |                                  |                 |                                 | "Mutasd mindet" gomb   |
|              |                                  |                 |                                 | Regisztrációs pop-up   |
|              |                                  |                 |                                 | Rendezés               |
|              |                                  |                 |                                 |                        |

15. táblázat: FreeGame fejlesztési folyamat 13. lépés

A 15. táblázat lépéseivel sikeresen zárulhat a FreeGame funkció fejlesztése és tesztelés teszt környezetben, így kiadható éles rendszerre, amit már valós felhasználók is látnak.

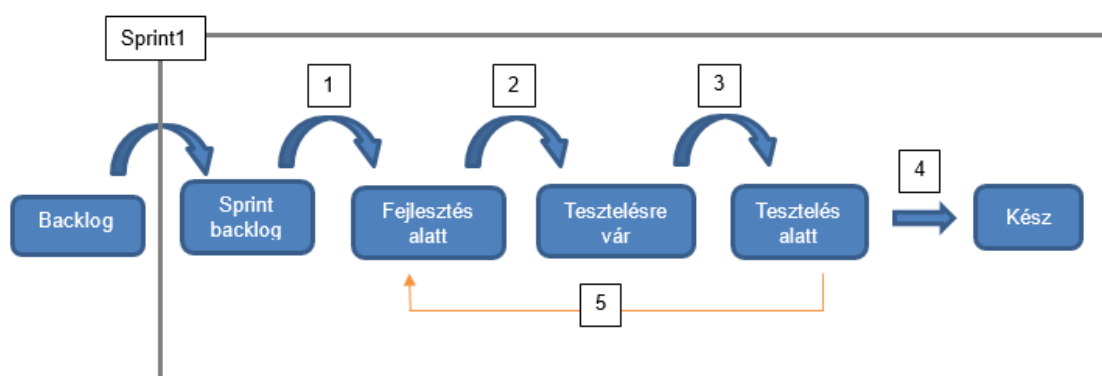
## 8. SCRUM

### 8.1. Fejlesztés SCRUM esetén

Ahogy a korábbiakban említésre került, a kanban mellett más módszerek is elterjedtek az informatika világában, melyekből a scrum az egyik listavezető. Előszeretettel használják ezt a módszert is, mivel egyes projektek esetén tökéletes választás lehet a csapat munkájának elősegítésében.

Míg a kanban esetén folyamatos fejlesztés történik, a scrumban iterációkban történik. Előre meghatározott feladatok listájára koncentrálnak, amit sprintekben rögzítenek. A sprintek jellemzően 2-4 hetesek és erre az időintervallumra előre kiválasztásra kerülnek azok a feladatok, amik teljesítése elvárt a sprint időtartama alatt. Az éles rendszerre történő kiadás a sprintek lezárásával kerül megvalósításra, így elvárt, hogy a sprint tételei időben teljesítésre kerüljenek, hogy még az adott csomagban megtörténhessen a kiadásuk.

A scrumban ez alapján kettő backlog létezik, az egyik a korábban is ismertetett backlog, amibe folyamatosan bekerülnek új és új feladatok (“product backlog”); a másik pedig az úgynevezett “sprint backlog”, aminek elemeivel a sprint elindításra kerül. A sprinten belüli feladatok mozgása hasonlóképpen történik ezután, mint a kanbannál, kivéve, hogy az oszlopok nincsenek limitálva és a sprinten belül már nem kerülhet be új teljesítendő tétel, tehát ez egy rögzített lista.



14. ábra: Scrum sprint

Egy sprint alatt megadott mennyiségű feladatot tudunk teljesíteni. Tekintve, hogy a feladatok különböző komplexitással rendelkeznek, az egy sprintben teljesíthető mennyiséget nem a feladatok számában, hanem azok komplexitásának számában mérjük, így a scrum-nál

jelentős szerepet kap a becslés. A becslési értékeket “story point”-nak nevezzük, mérhető többféle formában, például /óra, /nap, /komplexitás. Az, hogy a csapat egy sprint alatt mennyit tud teljesíteni (“velocity”), általában tapasztalat alapján alakul ki, mint a WIP limit is.

A PULL stratégia a scrum esetén is jelen van, mind a sprint backlog összeállításánál, mind a további folyamatokban. Az adott sprinten belüli folyamatok hasonlóan működnek, mint a kanbannál, viszont itt fix rögzített backlog van és csak annak elemeit kell teljesíteni adott időn belül.

### **Esetpélda5**

Scrum esetén a FreeGame funkció lefejlesztésénél először becslésre kerülnek a backlogban lévő storyk.

A becslésre a Fibonacci-számokat használtam, minél nagyobb az érték, annál magasabb komplexitású a részfeladat. A becsült érték tartalmazza a fejlesztői és tesztelői munkát is.

| Story                            | Story point (becslés) | Prioritás |
|----------------------------------|-----------------------|-----------|
| <b>1: FreeGame blokk</b>         | 3                     | magas     |
| <b>2: FreeGame játékkeret</b>    | 5                     | magas     |
| <b>3: “Mutasd mindet” gomb</b>   | 2                     | közepes   |
| <b>4: Lapozás</b>                | 3                     | közepes   |
| <b>5: SEO ellenőrzés – teszt</b> | 1                     | közepes   |
| <b>6: Rendezés</b>               | 2                     | alacsony  |

*16. táblázat: FreeGame - becslési értékek*

A csapat kettő hetes sprintekben dolgozik és a velocity-je 14 story point, általában ennyit tud teljesíteni egy sprint alatt. A sprint1-be így 14 story point-nak megfelelő feladat kerülhet be. A becslés szerint a funkció lefejlesztése összesen 16 story point, így egy sprintbe nem

fér be az összes elem. Mivel a sprintek végén van kiadás – “release” – az éles rendszerre, így nem optimális, hogy egyben nem tud kikerülni éles rendszerre. A prioritást és a story point-okat figyelembe véve a Rendezés story marad ki a sprint1-ből és kerül át a sprint2-be. Mivel a funkciót 3 hét múlva szeretnék a felhasználóknak rendelkezésre bocsátani, ezért a következő opciók jöhetnek szóba:

- rendezés nélkül kerül kiadásra a funkció a sprint1 végén és a rendezés a sprint2-vel kerül kiadásra (azaz 4 hét múlva)
- egyben kerül ki a funkció a sprint2 végén

Az első opció kerül kiválasztásra, mivel a rendezés fontossága elenyésző ahhoz képest, hogy 2 hét alatt hány új potenciális felhasználót hozna az új funkció.

A Sprint1 részei:

- 1: FreeGame blokk
- 2: FreeGame játékkeret
- 3: “Mutasd mindet” gomb
- 4: Lapozás
- 5: SEO ellenőrzés – teszt

A sprint1 kiindulási pontja a 15. ábrán látható. A kettő hetes sprint alatt az a cél, hogy a sprint1 elemei a Teljesítendő oszlopból a Kész oszlopba kerüljenek.

| Sprint1                |                  |                 |                 |      |  |
|------------------------|------------------|-----------------|-----------------|------|--|
| Teljesítendő           | Fejlesztés alatt | Tesztelésre vár | Tesztelés alatt | Kész |  |
| FreeGame blokk         |                  |                 |                 |      |  |
| Illetékes: -           |                  |                 |                 |      |  |
| Prioritás: magas       |                  |                 |                 |      |  |
| FreeGame játékkeret    |                  |                 |                 |      |  |
| Illetékes: -           |                  |                 |                 |      |  |
| Prioritás: magas       |                  |                 |                 |      |  |
| Lapozás                |                  |                 |                 |      |  |
| Illetékes: -           |                  |                 |                 |      |  |
| Prioritás: közepes     |                  |                 |                 |      |  |
| "Mutasd mindet" gomb   |                  |                 |                 |      |  |
| Illetékes: -           |                  |                 |                 |      |  |
| Prioritás: közepes     |                  |                 |                 |      |  |
| SEO ellenőrzés - teszt |                  |                 |                 |      |  |
| Illetékes: -           |                  |                 |                 |      |  |
| Prioritás: közepes     |                  |                 |                 |      |  |

15. ábra: Scrum tábla

Az időközben bejövő új story (Regisztrációs pop-up) nem kerül be a sprint1 rögzített listájába, hanem a product backlog-ba.

A sprint1 befejezése után megtörténhet a sprint2 megtervezése az éppen aktuális backlog lista alapján és a következő sprint már azokkal a kiválasztott elemekkel indul el.

## 8.2. Scrum és kanban

A korábbi esetpéldák szemléltetésével a scrum és kanban között a következő folyamat szintű különbségek figyelhetők meg:

|                                 | KANBAN                                                 | SCRUM                                                              |
|---------------------------------|--------------------------------------------------------|--------------------------------------------------------------------|
| <i>Munka folyamata</i>          | a teljes backlog rendelkezésre áll                     | a fókusz csak az adott sprint elemein van                          |
| <i>Változások</i>               | rugalmasan kezeli                                      | fix, nem rugalmas                                                  |
| <i>Mérőszámok</i>               | átfutási idő, ciklusidő (8. fejezet), wip              | “velocity”                                                         |
| <i>Vizualitás</i>               | kanban tábla (12. és 13. ábra)                         | scrum tábla (15. ábra)                                             |
| <i>Korlátozás</i>               | WIP limit (esetpéldában a fejlesztés: 3, tesztelés: 2) | előre meghatározott sprint elemek (sprint1)                        |
| <i>Tervezés</i>                 | nem szükséges                                          | szükséges (például a feladatok becslésére, sprintek betervezésére) |
| <i>Verzió kiadása (release)</i> | Nincs fix dátum                                        | Fix dátum, sprinthez kötött                                        |

17. táblázat: kanban vs scrum

A 8. táblázatban látható, hogy mindkét módszer más szemlélettel közelíti meg a szoftverfejlesztés folyamatát.

## 9. Javaslatok a folyamaton belüli fejlesztésekre

### 9.1. CFD elemzés

Folyamatos tökéletesítésre törekszünk a kanban során, így bizonyos időközönként javasolt felülvizsgálni a folyamatokat. CFD-vel könnyen elemezhető és azonosítható, hogy a folyamaton belül milyen területek fejlesztendőek. A CFD (cumulative flow diagram) rövidítés a kumulatív folyamatábrát jelenti.

Az elemzés segít megvizsgálni a következő mérőszámokat:

- **Ciklusidő:** megmutatja, hogy egy tétel feldolgozásához mennyi idő szükséges. Tulajdonképpen az az időtartam, ami eltelik onnantól kezdve, hogy elkezdenek dolgozni a feladaton odáig, hogy befejezik, tehát a fejlesztéstől a kész oszlopig. Előfordul, hogy ciklusidőt mérnek csak egy státuszra vonatkozóan is. A ciklusidő optimális hosszának meghatározása nagyban függ a feladat jellegétől. Amennyiben hosszú a ciklusidő a feladat komplexitásához képest, érdemes megvizsgálni a lehetséges okokat.
- **Átfutási idő:** Az átfutási idő tulajdonképpen a ciklusidő és a válaszüő egyben, tehát onnantól hogy bekerül az elem a backlogba odáig, hogy kész lesz.
- **WIP:** folyamatban lévő elemek
- **Backlog:** teljesítendő tételek

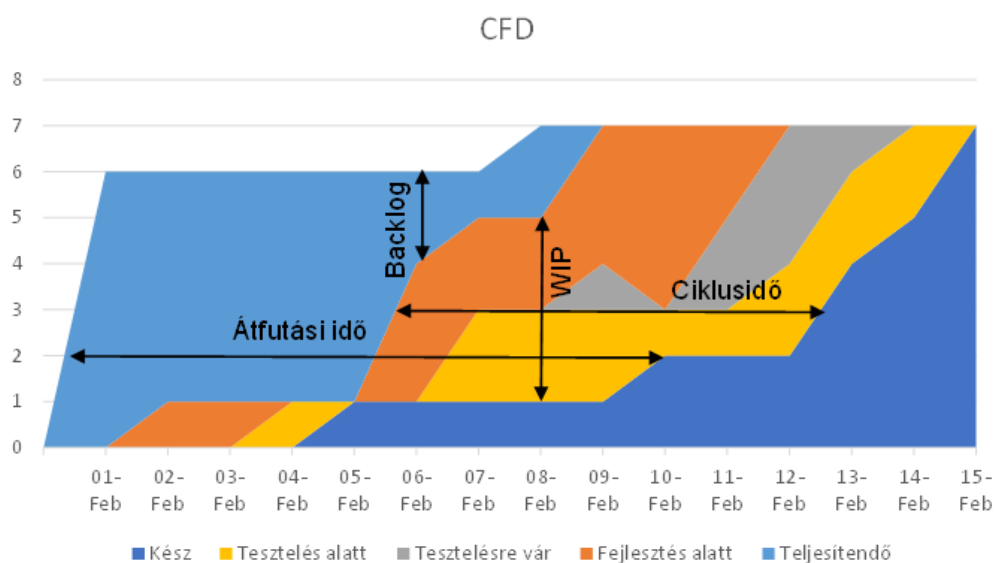
#### Esetpélda6

A FreeGame funkció fejlesztési folyamatának CFD elemzéséhez szükséges adatokat a 18. táblázat tartalmazza. A táblázatban szereplő adatok azt mutatják, hogy adott időpontban hány tétel volt az egyes oszlopokban.

| Dátum  | Teljesítendő | Fejlesztés alatt | Tesztelésre vár | Tesztelés alatt | Kész |
|--------|--------------|------------------|-----------------|-----------------|------|
| 01-Feb | 6            | 0                | 0               | 0               | 0    |
| 02-Feb | 5            | 1                | 0               | 0               | 0    |
| 03-Feb | 5            | 1                | 0               | 0               | 0    |
| 04-Feb | 5            | 0                | 0               | 1               | 0    |
| 05-Feb | 5            | 0                | 0               | 0               | 1    |
| 06-Feb | 2            | 3                | 0               | 0               | 1    |
| 07-Feb | 1            | 2                | 0               | 2               | 1    |
| 08-Feb | 2            | 2                | 0               | 2               | 1    |
| 09-Feb | 0            | 3                | 1               | 2               | 1    |
| 10-Feb | 0            | 4                | 0               | 1               | 2    |
| 11-Feb | 0            | 2                | 2               | 1               | 2    |
| 12-Feb | 0            | 0                | 3               | 2               | 2    |
| 13-Feb | 0            | 0                | 1               | 2               | 4    |
| 14-Feb | 0            | 0                | 0               | 2               | 5    |
| 15-Feb | 0            | 0                | 0               | 0               | 7    |

18. táblázat: FreeGame fejlesztési adatok CFD elemzéshez

A 18. táblázatban összegyűjtött adatokból szerkesztett kumulatív folyamatábra a 16. ábrán látható. A CFD ábra a FreeGame funkció fejlesztésénél a story-k áramlását mutatja a kezdeti ponttól a befejezésig.



16. ábra: Kumulatív folyamatábra (CFD)

A CFD ábra 15 nap alakulását prezentálja. A fejlesztés időtartama alatt a következő megállapítások olvashatóak le:

- **Teljesítendő:** a vizsgált időtartam 1. napján 6 tétel volt a backlogban. A február 8-ai adat alapján megállapítható, hogy a backlogba új tétel kerültek be, mielőtt még

teljesítve lett volna az összes ezt megelőzően hozzáadott tétel. → kis mértékben nem okoz problémát, de ha szignifikánsan nagyobb mértékben nő a backlog, mint ahogy a Kész oszlop elemeinek száma, akkor problémát jelenthet. Ebben az időszakban a fejlesztés alatt összesen 1 tétellel nőtt a backlog, ami rendben van a folyamat szempontjából, mivel a kanban rugalmas ezen a téren. A fejlesztés végére a Teljesítendő oszlop és a Kész oszlop ugyanabban a vonalban zárul (a 7-es szinten), így leolvasható, hogy minden teljesítendő feladat megvalósításra került (7-ből 7).

- **Fejlesztés alatt:** a CFD alapján a fejlesztési csapat nem mindig töltötte ki az oszlophoz rendelt 3-es limitet. Február 1-5 között csak egy tétel volt fejlesztés alatt, viszont ennek oka, hogy az előfeltétele volt a többi fejlesztés megkezdésének. A többi időszakban a csapat törekedett arra, hogy amíg a backlog és a limit is engedi minél több feladattal haladjanak. Február 7-én a backlogban nem volt fejlesztési munkálat, így azért nem került be semmi a fejlesztési csapathoz (a SEO story nem igényelt fejlesztést). Február 8-án megfigyelhető, hogy a Regisztrációs pop-up bekerült a Teljesítendő tételekhez, amin másnap el is kezdett dolgozni a csapat. Továbbá, február 10-én a csapat átlépte a WIP limitet, mivel a tesztelésen megbukott egy story. A túllépett WIP limit miatt ebben az időszakban belassulhatott a fejlesztés. → megoldás lehet, ha ilyen esetben továbbra is csak 3 tételen dolgoznak a 4-ből (prioritás alapján) és a 4.-el csak akkor kezdenek el foglalkozni, ha szabadul kapacitás
- **Tesztelésre vár:** az ábra alapján többször is előfordult, hogy ez az oszlop üresen állt. Tesztelés esetén természetesnek számít, hogy a folyamat elején kevesebb a funkció-tesztelés, mivel a fejlesztés még nincs olyan fázisban. A tesztelői csapat ezekben az időszakokban tudott foglalkozni a tesztelési munkálatok előkészítésével, például teszt esetek/teszt dokumentációk írásával, teszt adatok előkészítésével. A fejlesztési folyamat későbbi szakaszaiban megfigyelhető, hogy a tesztelésre váró tételek nem álltak sokáig várakozó állapotban és a tesztelői csapat hamar elkezdett dolgozni ezeken az elemeken.
- **Tesztelés:** a vizsgált időszak alatt a tesztelés folyamata az elvártak szerint működött. Amikor volt tesztelésre váró feladat, akkor hamar átkerült tesztelés alá. Megfigyelhető, hogy február 12-től már csak tesztelői feladatok voltak, ez ebben az esetben elfogadható és logikus.
- **Kész:** a kész tételek listája folyamatosan nő, nincs lefele ágazó rész, tehát nincs olyan

feladat, ami visszamozgatásra került ebből a státuszból, az oszlop tételeinek száma folyamatosan növekszik.

### Mérőszámok elemzése:

#### Ciklusidő

A WIP limit bevezetésével sikerült elérni, hogy ne legyen túl sok tétel egyszerre folyamatban, ami által a WIP és a ciklusidő is csökken. A ciklusidő a 16. ábrán is látható. A ciklusidő az ábrán három státuszt ölel fel, a fejlesztés alatt, tesztelésre váró és tesztelés alatt lévő oszlopokat. Az egyes oszlopok elemzése a korábbiakban került ismertetésre. A ciklusidő viszonylag elfogadható mértékű, pár kirívóbb esetet leszámítva, ezekben az esetekben a fejlesztői vagy tesztelői munkálatok hosszabbak. A tesztelésre váró oszlopban töltött átlagos ideje egy tételnek elenyésző, ez segít a ciklusidő alacsony szinten tartásában. A fejlesztői és tesztelői munkálatok idejének csökkentésében szükséges megvizsgálni az egyes eseteket. Amennyiben a fókuszvesztés a kiváltó ok, úgy a WIP limitet érdemes csökkenteni. Tesztelésnél az is előfordulhat, hogy az előkészületek nem megfelelőek, például a Lapozás és "Mutasd mindet" gomb tesztelése egyszerű és rövid feladatnak tűnik, mégis több napot vett igénybe, így itt blokkolta a munkálatokat valami. Blokkoló lehetett például, hogy az egyes kategóriák nem voltak előre feltöltve játékokkal, ami mindkettő story teszteléséhez szükséges. A játékok feltöltésének átfutási ideje van, így azok csak másnap kerülnek láthatóvá az oldalon.

A ciklusidő csökkentésének érdekében a továbbiakban érdemes bevezetni egy extra jelölést, ami a feladatok egyes státuszokban töltött idejére hívja fel a figyelmet a kanban táblán. Például a 17. ábrán a Február 9-ei napon a Lapozás és "Mutasd mindet" gomb 3 napja vannak tesztelés alatt, ez a pontokat tartalmazó hozzáadott sorban olvasható. Ez a lépés segít a ciklusidő adott státuszra vonatkozó időtartamát korrigálni oly módon, hogy felhívja a figyelmet.

| Tesztelés alatt             |
|-----------------------------|
| <b>Lapozás</b>              |
| Illetékes: XY               |
| Prioritás: közepes          |
| ...                         |
| <b>"Mutasd mindet" gomb</b> |
| Illetékes: ZV               |
| Prioritás: közepes          |
| ...                         |

17. ábra: Státuszokban töltött idő jelölése

### Átfutási idő

A ciklusidő nagyban befolyásolja az átfutási időt, ennek csökkentésére vonatkozó javaslatok az előző pontban találhatóak. A válaszügy viszonylag magasnak számít a fejlesztés első szakaszában. Ennek oka főleg az, hogy a FreeGame blokk befejezésétől függött a többi feladat, illetve a WIP limit. Mivel a projekt teljesült a meghatározott időszáron belül, így a jelenlegi csapatfelállással nem növelném a WIP limitet, mivel ez a fókuszuvesztés miatt nem garantálná, hogy rövidebb lenne az átfutási idő. Továbbá látható, hogy ahol új story került be ott rövid válaszügy történt, tehát a rendelkezésre álló adatok alapján, a csapat gyorsan reagál, ahol nincs egyéb korlátozó tényező.

### WIP

A WIP tételek száma korlátozva vannak a WIP limitek által. A funkció fejlesztése során a Fejlesztés és Tesztelés alatt lévő tételekre volt meghatározott limit. A tesztelésre váró feladatok oszlopára is be lehet vezetni korlátozást, viszont látható, hogy a story-k nem sokat várakoztak ebben az oszlopban, így nem érdemes limitálni. A megfigyelt időszak alatt látható, hogy maximum három tétel szerepelt itt, amit rövid időn belül át is tudott venni a tesztelési csapat. A későbbiekben is érdemes figyelmet fordítani az oszlop statisztikájára, hogy igényel-e limit bevezetést. Ez a lépés azt eredményezné, hogy amennyiben a folyamat megakad a két tesztelési oszlop limitje miatt, úgy más erőforrást kell mozgosítani a folyamat haladása érdekében, például egy fejlesztő ebben az esetben besegíthetene a tesztelésbe. Ezenfelül, ajánlott a meghatározott WIP limitek felülvizsgálata bizonyos időközönként, hiszen a túl alacsony WIP limit ahhoz vezet, hogy nincs kihasználva minden kapacitás, a túl magas WIP korlát pedig fókuszuvesztéshez és hosszabb átfutási időhöz vezethet.

### Backlog

A backlog alakulása a korábbiakban került elemzésre.

## **9.2. További javaslatok az esetpélda folyamatának optimalizálására**

Külső függőségek esetén előfordulhat, hogy egy feladat egy adott státuszban ragad. Annak érdekében, hogy a csapatot ne akadályozza további feladatok teljesítésében, amíg más csapatra várnak, megoldásként szolgálhat, ha egy új oszlop kerül bevezetésre a kanban táblán. A blokkolt tételek így átkerülnek ideiglenesen ebbe az oszlopba és a csapatok dolgozhatnak más feladatokon, amíg a külső függőség miatti blokk megszűnik. Amikor újra

tudunk dolgozni a korábban blokkolt feladaton, egyszerűen visszatesszük a megfelelő státuszba és onnan folytatjuk, ahol abbahagytuk.

Természetesen nem optimális blokkolt tételek felhalmozódása, így érdemes folyamatosan felülvizsgálni, hogy még mindig blokkolt-e az adott elem – amennyiben igen, megsürgetni a folyamatot, ha lehetséges-.

## 10. Összefoglalás

A kanban informatikai és logisztikai felhasználása sok hasonlóságot rejt magában, ahogy az 1. táblázatban is szemléltetésre került. A esetpéldákon keresztül megfigyelhető, hogy a vizuális megvalósítás kártyák és kanban tábla formájában történnek. A kártyák nagyrészt pull rendszer alapján mozognak a státuszok és oszlopok között. Az egyes státuszokban lévő mennyiségek szabályozására a WIP limit ad lehetőséget.

Továbbá, a 7. fejezet példáján keresztül is megfigyelhető, hogy a kanban rugalmasan kezeli a változásokat, így olyan projektek esetén is megfelelő választás lehet, ahol változásokra számítunk és nem tudunk előre tervezni pár héttel. A kanban nagyfokú flexibilátása lehetővé teszi, hogy bármikor új kártyákat adjunk hozzá a backlog-hoz vagy akár módosításra és törlésre is lehetőséget ad. A példákön látható, hogy a kártyákhoz kötődő módosítások nem akasztják meg és nem befolyásolják a későbbi folyamatokat, a munka zökkenőmentesen tud továbbhaladni.

A 6. fejezetben ismertetett scrum olyan csapatok munkáját tudja elősegíteni jobban, akiknél a munka előre tervezhető legalább pár héttel. Az esetpélda scrum-mal is megvalósítható lett volna, viszont kompromisszumokkal a kiadás dátumát és jellegét tekintve. A scrum a sprinten belül nem rugalmas, így egy nem tervezhető projektnél fennakadások jelentkeznének. Van lehetőség a két rendszer vegyítésének alkalmazására is, ezzel egy olyan hibrid rendszert tudunk alkalmazni, ami esetén a scrum és kanban elemeit kombinálva a tulajdonságok az aktuális projektre vannak szabva (scrumban).

A 8. fejezet elemzéséből egyértelműen látható, hogy a kanban a folyamatok optimalizálásával a rendszer tökéletesítésére ad lehetőséget. A mérőszámok segítségével könnyebben kimutathatóak a fejlesztendő területek hiányosságai vagy növekedési lehetőségei. A monitorozás és ehhez köthető szükséges változtatások jelentősen növelni tudják a hatékonyságot. A kanban használatánál szükséges és előnyös a rendszer folyamatos fejlesztése. Az elemzések által számtalan olyan pont merült fel, ahol tovább finomítható a rendszer, például a ciklusidő csökkentésével, a WIP limitek felülvizsgálatával, további erőforrás bevonásával a fejlesztői csapatba, blokkolt elemek külön kezelésével stb.

A kanban folyamatok fejlesztése lehetővé teszi, hogy közelebb jussunk a csapat és a projektcélok megvalósításához legmegfelelőbb verzióhoz.

A diplomadolgozatom írása során számtalan új és hasznos ismerettel lettem gazdagabb. Az esetpéldák segítségével és azok elemzésével mélyebb tudásra tudtam szert tenni a témában, amelyeket a későbbiekben fel tudok használni a szakmám során. Ezáltal a saját munkámat és a projektcsapat munkáját is segíteni tudom a projektcélok elérésének érdekében. Összeségében úgy gondolom, hogy a diplomamunkámban elemzett módszerek működésének megismerése hasznos lehet a szoftverfejlesztés területén dolgozóknak és felhívja a figyelmet a módszer kiválasztásának fontosságára.

# Irodalomjegyzék

## Szakirodalom

- [1] Cimorelli, S. (2013). *Kanban for the supply chain: Fundamental Practices for Manufacturing Management*. CRC Press.
- [2] Demeter, K. (2016). *Termelés, szolgáltatás, logisztika*. Budapest: Wolters Kluwer Kft.
- [3] Gelei, A. (2016). *Logisztikai döntések*. Budapest: Akadémiai Kiadó.
- [4] Greene J., Stellman A. (2014). *Learning Agile: Understanding scrum, xp, leand and kanban*. O'Reilly Media.
- [5] Blahó A.-Czakó E.-Poór J. (2021). *Nemzetközi menedzsment*. Budapest: Akadémiai Kiadó.
- [6] Kovács, Z. (2017). *A termelő és szolgáltató rendszerek fejlesztésének főbb irányai*. Budapest: Akadémia kiadó.
- [7] Monden, Y. (2012). *Toyota Production System: An Integrated Approach to Just-In-Time*. CRC Press.
- [8] ProjectManagementInstitute. (2018). *Agilis gyakorlati útmutató*. Budapest: Akadémiai Kiadó.
- [9] Sammons, A. (2020). *Agile Project Management with kanban revealed: The Secret to get out of stress & Overwhelming work to finally become productive*. M&M Limitless Online Inc.
- [10] Skarrin, M. (2015). *Real-World Kanban - Do Less, Accomplish More with Lean Thinking*. The Pragmatic Bookshelf.

## Cikk

- [11] Koncz, A. (2017). A lean szemlélet eszköztára = Tools of Lean Thinking. GRADUS, 4 (1). ISSN 2064-8014. pp. 229-241. .

## Internetes hivatkozások

- [12] GuideForSoftwareTesting. (2014). *GuideForSoftwareTesting*. Forrás: What is Software Development Life Cycle (SDLC)?:  
<http://guideforsoftwaretesting.blogspot.com/2014/10/what-is-software-development-life-cycle.html>  
Letöltés ideje: 2023.04.17.
- [13] GyartasTrend. (2009). *GyartasTrend*. Forrás: Just-in-time a középkorból:  
<http://gyartastrend.hu/autoipar/cikk/just-in-time-a-koezepkorbol>  
Letöltés ideje: 2023.04.17.
- [14] Kosztolányi, J. (2016). *Kaizenpro*. Forrás: Jidoka, andon, poka-yoke:  
<https://kaizenpro.hu/jidoka-andon-poka-yoke>  
Letöltés ideje: 2023.05.02.

- [15] Kosztolányi, J. (2016). *KaizenPro*. Forrás: Kanban: <https://kaizenpro.hu/kanban/>,  
Letöltés ideje: 2023.04.17.
- [16] LeanSzotar. (2012-2021). *Lean Szótár*. Forrás: Just-in-time (JIT):  
<https://leanszotar.hu/page.php?89>  
Letöltés ideje: 2023.04.17.
- [17] LogAp. (2021). *Logap*. Forrás: Kanban features: what it is, how it works and main  
benefits: <https://logap.com.br/en/blog/characteristics-of-kanban/>  
Letöltés ideje: 2023.04.17.
- [18] PromanConsulting. (2020). *Proman Consulting*. Forrás: Kanban: jelentése,  
előnyök/hátrányok, kanban vs. scrum: [https://promanconsulting.hu/kanban-  
rendszer/](https://promanconsulting.hu/kanban-rendszer/)  
Letöltés ideje: 2023.05.04.
- [19] Transpack. (2017). *Transpack*. Forrás: „Just In Time” – a logisztika csúcsra járatása a  
termelésben: [https://transpack.hu/2017/10/19/just-in-time-a-logisztika-csucsra-  
jaratasa-a-termelesben/](https://transpack.hu/2017/10/19/just-in-time-a-logisztika-csucsra-jaratasa-a-termelesben/)  
Letöltés ideje: 2023.04.17.

# Függelék

## 1. sz. függelék – Hallgatói és konzulensi nyilatkozat



Szent István Campus, Gödöllő  
Cím: 2100 Gödöllő, Péter Károly utca 1.  
Tel.: +36-28/522-000  
Honlap: <https://godollo.uni-mate.hu>

### Hallgatói és konzulensi nyilatkozat

#### NYILATKOZAT

Alulírott TÓTH KATA, a Magyar Agrár- és Élettudományi Egyetem,

GÖDÖLLŐI (Szent István) Campus,  
ELLÁTHATÁSI ÁLLÁS - MENEDZSMENT MSc szak nappali/levelező\* tagozat végzős hallgatója nyilatkozom, hogy a dolgozat saját munkám, melynek elkészítése során a felhasznált irodalmat korrekt módon, a jogi és etikai szabályok betartásával kezeltem. Hozzájárulok ahhoz, hogy Záródolgozatom/Szakdolgozatom/Diplomadolgozatom egyoldalas összefoglalója felkerüljön az Egyetem honlapjára és hogy a digitális verzióban (pdf formátumban) leadott dolgozatom elérhető legyen a témát vezető Tanszéken/Intézetben, illetve az Egyetem központi nyilvántartásában, a jogi és etikai szabályok teljes körű betartása mellett.

A dolgozat állam- vagy szolgálati titkot tartalmaz: igen nem\*

Budapest,

Kelt: 2023 év 05 hó 03 nap

Tóth Kata  
Hallgató

#### NYILATKOZAT

A dolgozat készítőjének konzulense nyilatkozom arról, hogy a Záródolgozatot/Szakdolgozatot/Diplomadolgozatot áttekintettem, a hallgatót az irodalmi források korrekt kezelésének követelményeiről, jogi és etikai szabályairól tájékoztattam.

A Záródolgozatot/Szakdolgozatot/Diplomadolgozatot záróvizsgán történő védelemre javaslom / nem javaslom\*.

A dolgozat állam- vagy szolgálati titkot tartalmaz: igen nem\*

Kelt: Gödöllő, 2023 év Május hó 03 nap

[Signature]  
Belső konzulens

\*Kérjük a megfelelőt aláhúzni!

## A DIPLOMADOLGOZAT TARTALMI KIVONATA

**Dolgozat címe:** Kanban bevezetése és működésének elemzése a szoftverfejlesztés területén

**A dolgozatot készítő hallgató neve:** Tóth Kata

**Szak, képzési szint és tagozat megnevezése:** Ellátásilánc-menedzsment MSC, levelező tagozat

**Tanszék/Intézet (ahol a dolgozat készült) megnevezése:**

Agrár- és Élelmiszergazdasági Intézet/Agrárlogisztika, Kereskedelem és Marketing Tanszék

*Belső témavezető:* Dr. Gyenge Balázs

*Külső témavezető:* -

### **Összefoglaló:**

A dolgozat témája a kanban módszer bevezetése és annak elemzése a szoftverfejlesztés világában. A szakirodalmi áttekintés segít a kanban logisztikai területen történő felhasználását megismerni, valamint a módszer informatikában elfoglalt helyét meghatározni.

A saját vizsgálat során a kanban gyakorlati alkalmazása esetpéldákon keresztül lett elemezve. Egyik célja a kanban sajátosságainak és hatékony működésének tanulmányozása, aminek részét képezi a limitációk vizsgálata is.

Továbbá, a kanban összehasonlításra került a scrum módszerrel, ahol a folyamatszintű hasonlóságok és különbségek kerültek megvizsgálásra.

A dolgozat igyekszik meghatározni az esetpélda folyamatának fejlesztendő területeit, amire a CFD elemzés is segítséget ad. Az elemzés által kiértékelésre került az esetpélda fejlesztési folyamata. Ennek eredményeképpen, a dolgozat végén javaslatok és következtetések is megfogalmazódtak a folyamatok tökéletesítésére vonatkozóan.