

Hatékony iteratív fejlesztési módszertan a gyakorlatban a RUP fejlesztési módszertanra építve



Kérdő Attila, ügyvezető, INSERO Kft.

EOQ MNB, Informatikai Szakosztály, HTE,

ISACA2012. május 17.

- Miért is kellenek nekünk módszertanok
 - Lelkes ifjunc: saját kárán tanul
 - Serdülő kor: a saját tapasztalatok beépítése
 - Érett kor: amit mások kitaláltak, ne találjuk fel újra

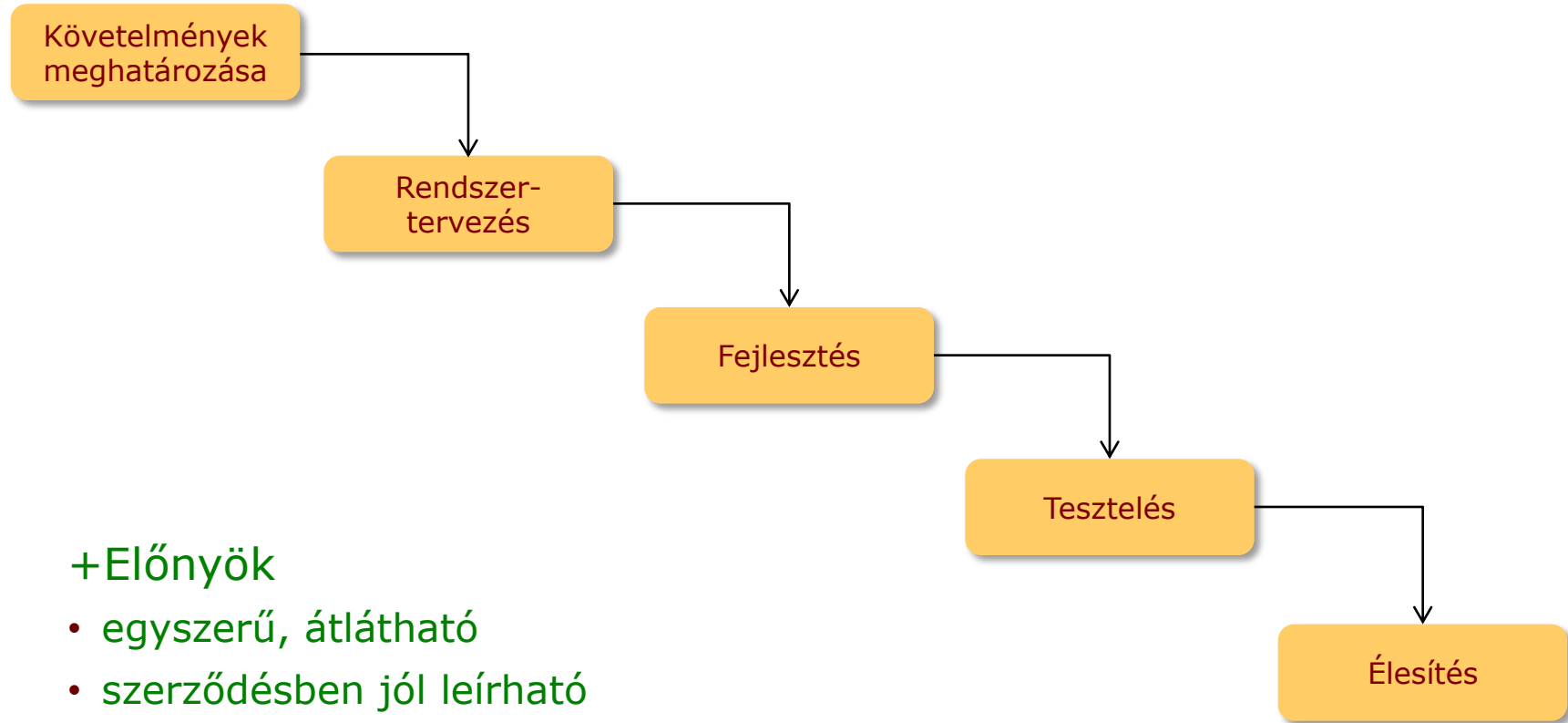
■ Módszertanok:

- A legjobban bevált gyakorlat összefoglalása
- Kapaszkodó a bonyolult fejlesztési projektek sikeres elvégzése
- Fontos a mögöttes elv megértése

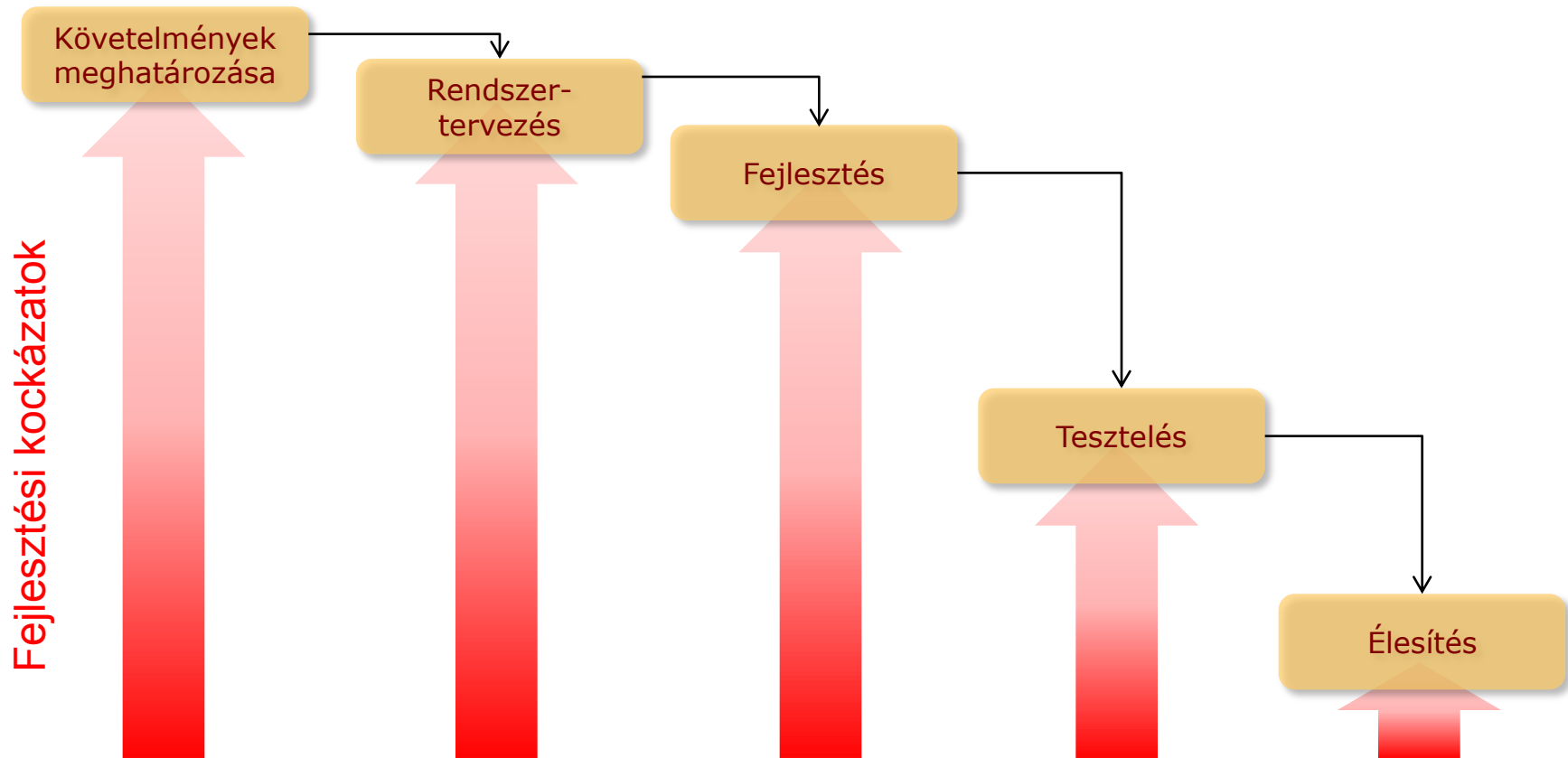
A módszertan nem szent és sérthetetlen!

A RUP modell háttere

Vízesésmodell



Vízesésmodell - hátrányok



– Hátrányok

- Nehezen tervezhető
- Magas kockázatokkal jár!

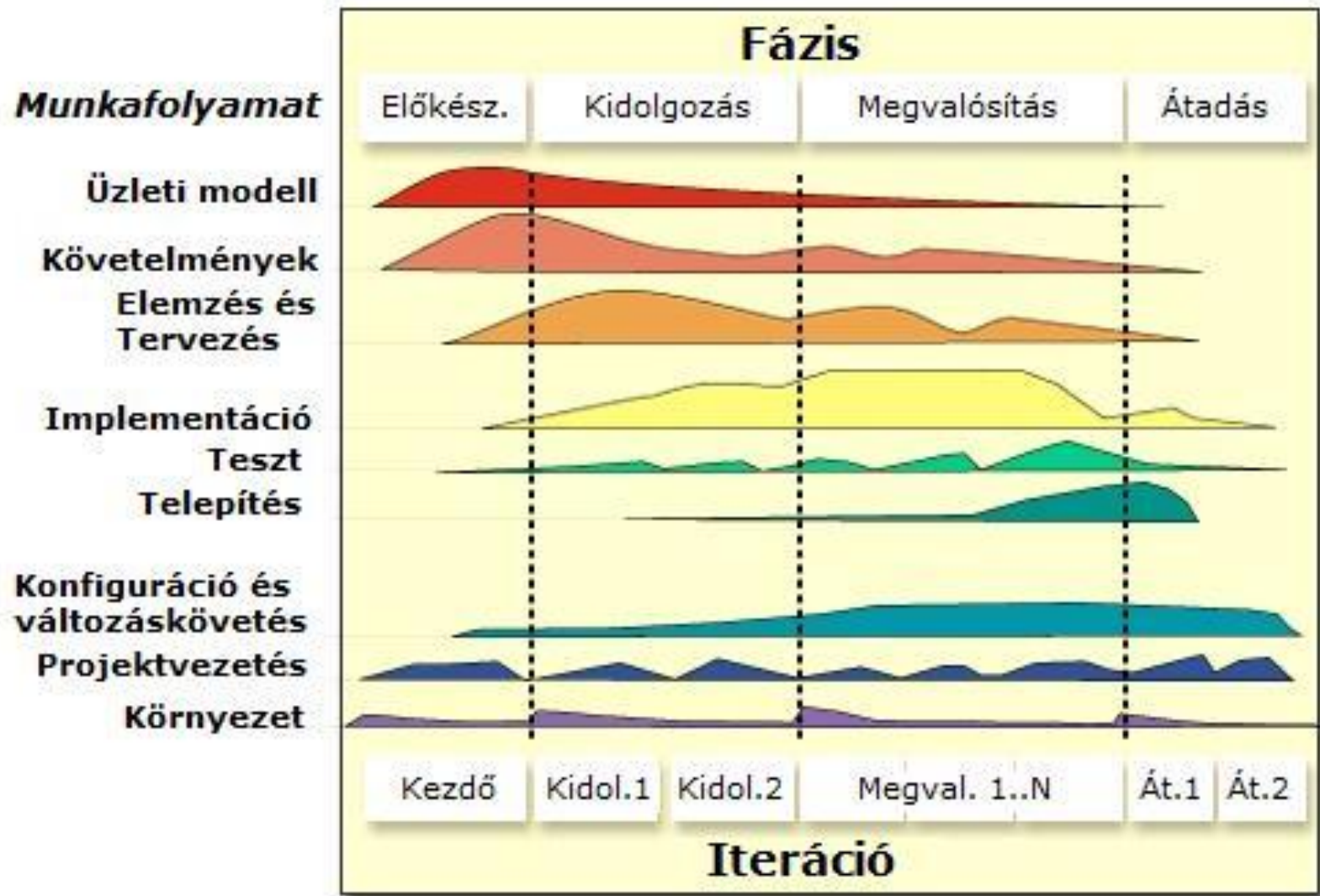
Kockázat:

Nem azt kapja a felhasználó, amit szeretne!

Iteratív megközelítés kell:

A felhasználó akkor tudja megmondani, hogy mire van szüksége, ha látja, hogyan működik a szoftver

RUP modell



Rational Unified Process (RUP)

- Egységes, széles körben elfogadott
- Teljes fejlesztési ciklust támogatja
- Fejlesztőknek és megrendelőknek is segít
- Rugalmasan alakítható
 - fejlesztési folyamatokhoz
 - rendszerhez

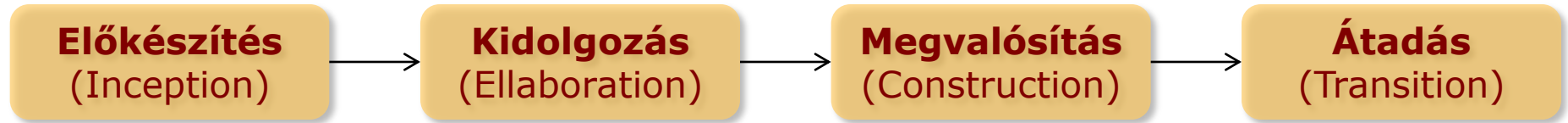
RUP rövid – fontosabb jellemzők

- Használati eset vezérlés (use case driven)
 - használati esetek meghatározása → ütemezés
- Architektúra központú
 - a munkafolyamatokban nagy hangsúlyt kap a megfelelő architektúra kialakítása
- Iteratív és inkrementális
 - fejlesztést ismétlődő, iteratív ciklusokban
 - egy iteráció a rendszert újabb funkcionalitással bővíti
- Teljes életciklust támogató megoldás
- Csapat munka támogatása

RUP - munkafolyamatok

- Üzleti modellezés (business modeling)
 - kezelendő entitások, használati esetek
- Követelmények (requirements)
 - rendszerrel kapcsolatos követelmények (funkcionális, nem funkcionális) – felhasználói szempontból
- Elemzés (analysis)
 - követelményekből kiindulva logikai rendszerterv
- Tervezés (design)
 - teljes körű tervezés programozáshoz
- Implementáció (implementation)
 - komponensek implementálása (forráskód, futtatható programok...)
- Teszt (test)
 - tesztek tervezése, végrehajtása, kiértékelése

RUP – fázisai



■ Előkészítés (Inception)

- Eredeti ötlet kidolgozása, mely alapján a fejlesztés tervezhető lesz
- Meghatározzuk az architektúrát és az alapvető felhasználást

■ Kidolgozás (Elaboration)

- Használati esetek kidolgozása
- Stabil architektúra kialakítása

■ Megvalósítás (Construction)

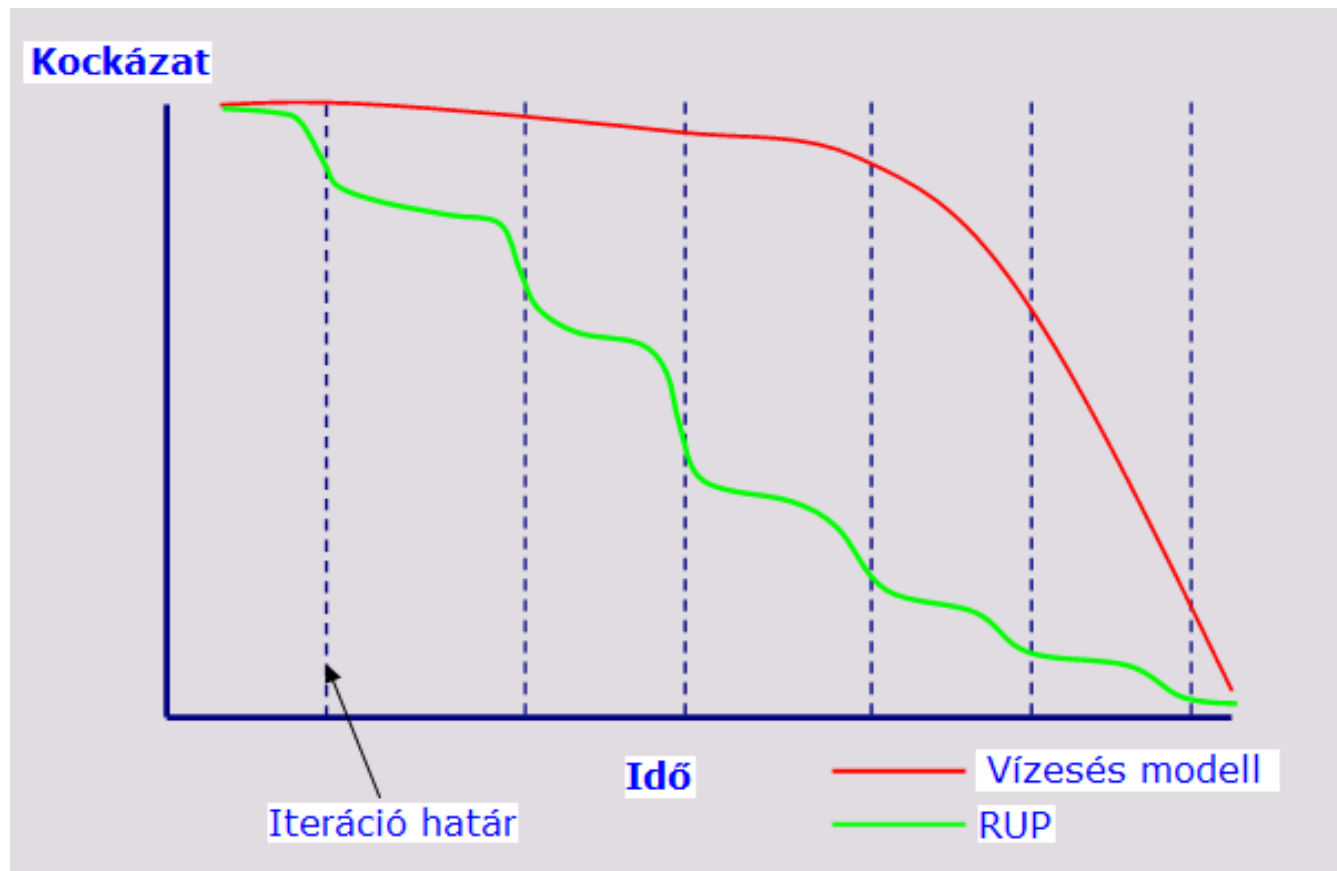
- Teljes rendszer kialakítása

■ Átadás (transition)

- A rendszer átadása a felhasználóknak tesztelésre

... és mitől hatékonyabb mint a vízesés modell?

RUP – kockázatok kezelése



Hogyan működik a gyakorlatban?

Esettanulmány:
Fejlesztési projekt RUP megközelítési móddal
(INSERO adaptáció)

Esettanulmány – fejlesztési projekt

- Eset
 - banki háttérű fejlesztés, befektetési rendszer bevezetése
- Fejlesztési projekt scope
 - „SHARE 2000” rendszer testreszabása egyedi igények alapján
- Háttér
 - Ilyen referenciával nem rendelkezik a szállító
 - Az üzleti igények nem tisztázottak, pontosítani kell
 - A program testre szabható, de nem ismertek a fejlesztési igények

Esettanulmány – fejlesztési projekt

A tervezett fejlesztési szakaszok (fázisok)

- 0. szakasz: Üzleti ötlet kidolgozása
 - Cél: alapvető üzleti célok és lefedendő üzleti folyamatok meghatározása
- 1. szakasz: Alapstruktúra kialakítása
 - Cél: az architektúra illetve az alpműködés kialakítása
- 2. Szakasz: Rendszer kialakítása
 - Cél: rendszer kialakítása
- 3. szakasz: Rendszer finomhangolása
 - Cél: a teljes és végső funkcionalitás kialakítása

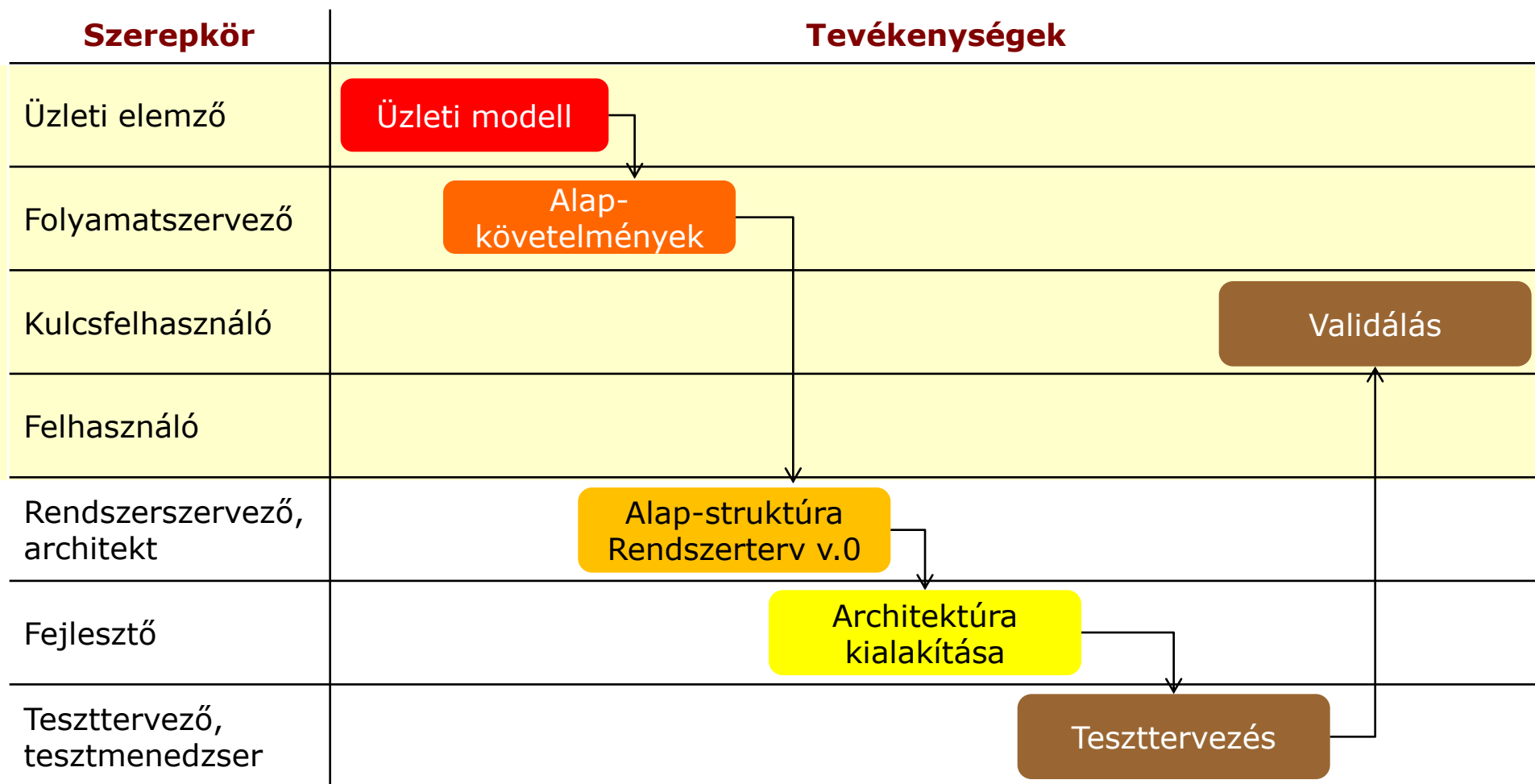
Szerepkörök

	Szerepkör	Tevékenységek
Üzlet, megrendelő	Üzleti elemző	Üzleti modell megalkotása Használati mód meghatározása
	Folyamatszervező	Követelmények összegyűjtése, rendszerezése
	Kulcsfelhasználó	Követelmények megfogalmazása Validálás, tesztelés
	Felhasználó	Tesztelés
IT, Szállító	Rendszerszervező, architekt	Architektúra kialakítása Rendszerterv elkészítése
	Fejlesztő	Fejlesztés
	Teszttervező, tesztmenedzser	Teszttervek, tesztforgatókönyvek elkészítése; Tesztelési tevékenységek felügyelete, eredmények kiértékelése

1. Szakasz – alapstruktúra kialakítása

- Cél
 - A felhasználók megismertetése a rendszer logikájával
 - Az alapstruktúra meghatározása

1. Szakasz – alapstruktúra kialakítása

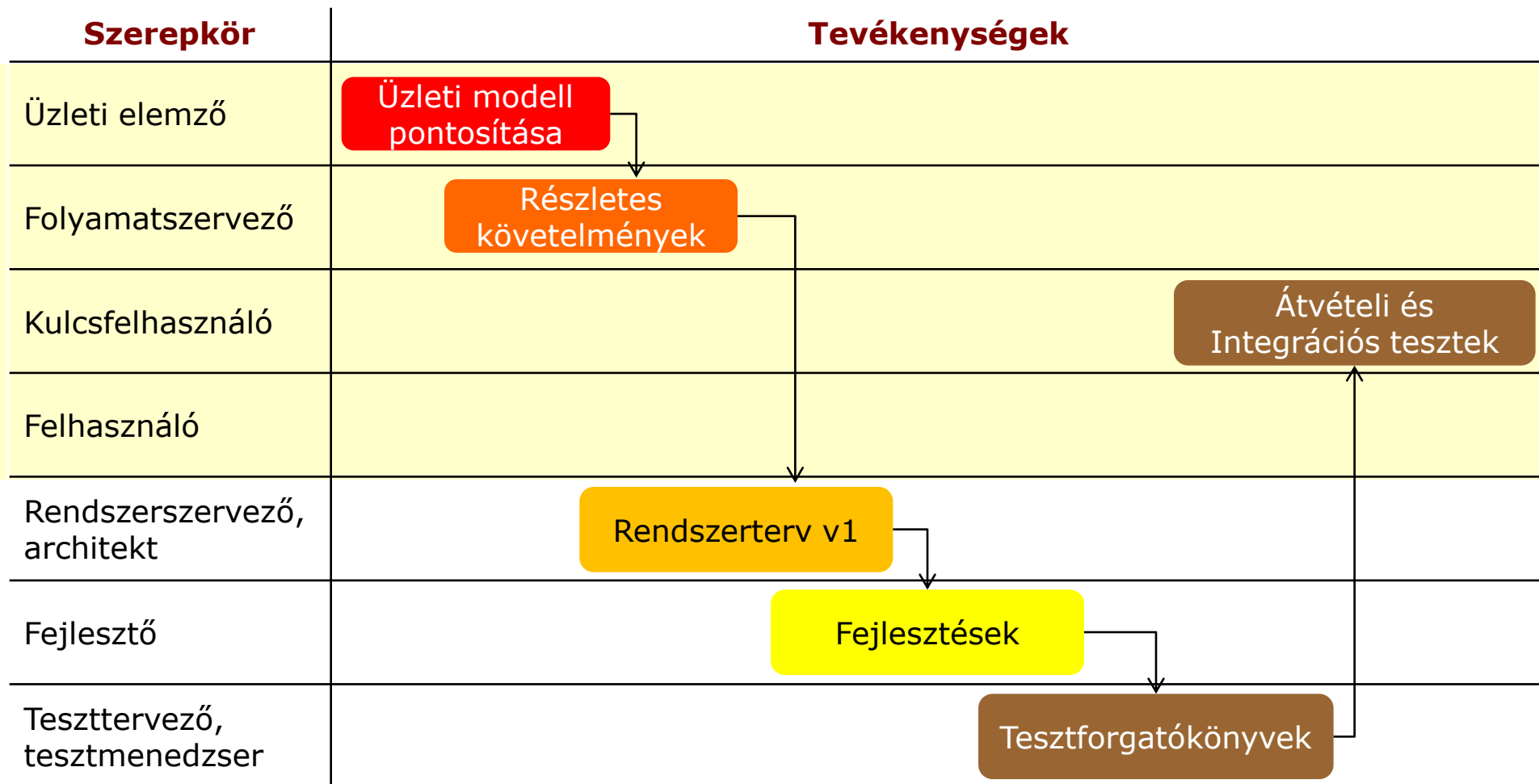


2. Szakasz – Rendszer kialakítása

- Cél

- Részletes követelmények meghatározása
- Teljes rendszer kialakítása

1. Szakasz – Rendszer kialakítása

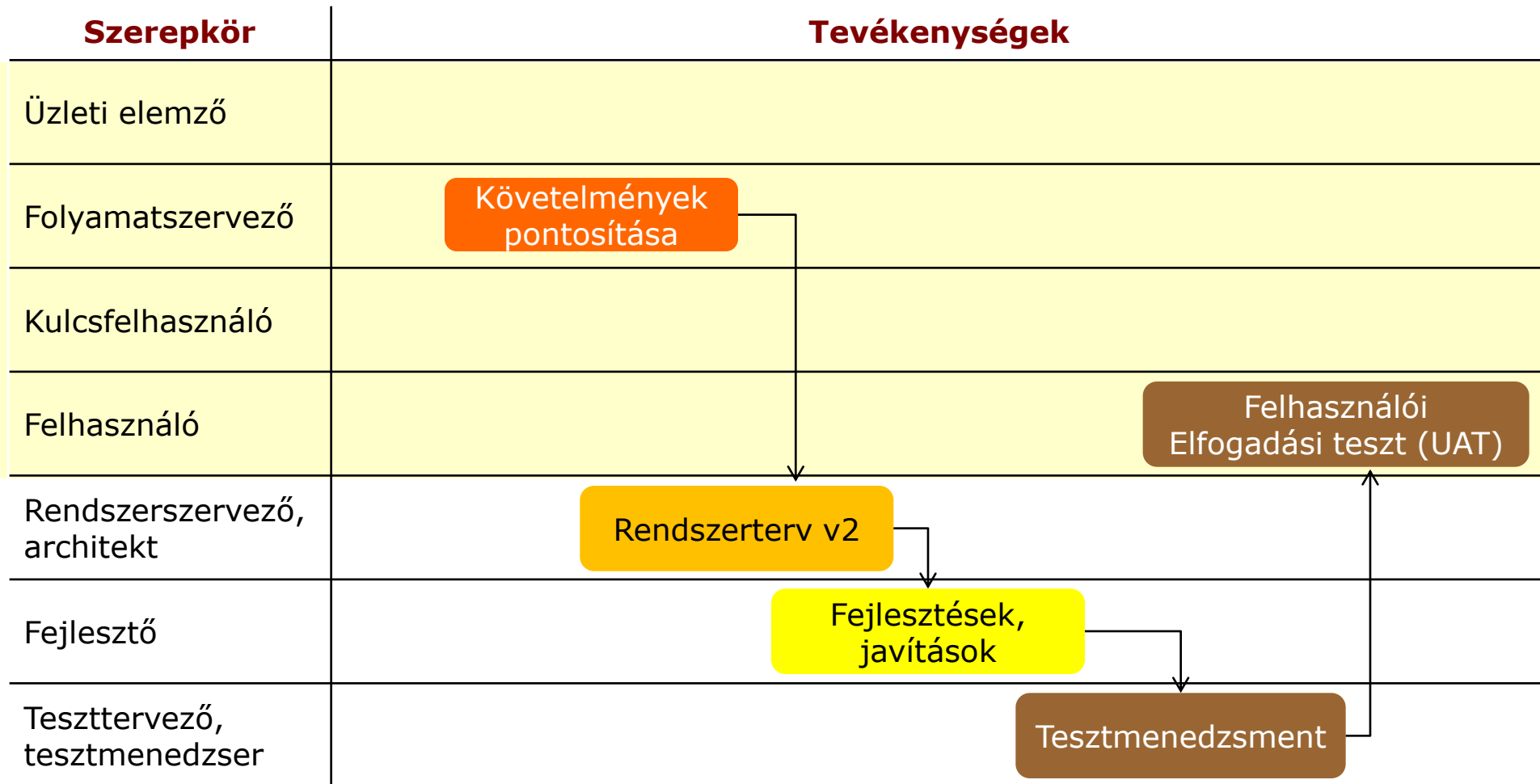


3. Szakasz – Rendszer finomhangolása

- Cél

- Kulcsfelhasználói tesztek alapján finomhangolás
- A végleges rendszer kialakítása

3. Szakasz – Finomhangolás



Tevékenységek összefoglalása

	1. Szakasz	2. Szakasz	3. szakasz
Üzleti modell	Üzleti modell meghatározása	Üzleti modell pontosítása	
Követelmények	Alapfolyamatok meghatározása	Követelmények pontosítása	Követelmények kiegészítése
Elemzés és tervezés	Alapstruktúra meghatározása	Rendszerterv elkészítése	Rendszerterv pontosítása
Fejlesztés	Alapstruktúra	Rendszer kialakítása	Finomhangolás, hibajavítás
Szállítás	Architektúra	Rendszer v1.0	Teljes rendszer v2.0
Tesztelés	Teszttervezés Validálás	Tesztforgatókönyvek Fejlesztői teszt, Átvételi és integrációs tesztek	Felhasználói elfogadási teszt (UAT)

RUP - Elért előnyök

- Jelentősen csökkentett fejlesztési kockázatok
- Alaposabb tervezése
- A felhasználók hamarabb megértik a működési logikát
- Nagyobb az esély, hogy az igényeknek megfelelő szoftver készül
- Jobban tartható fejlesztési határidők
- Nincs kritikus hiba az éles rendszerben



JOBB MINŐSÉGŰ ALKALMAZÁS

Mi a helyzet Magyarországon?

- Sikeres informatikai projektek aránya ~30-35%
- Vízesés modell előnyben !
- Válság újabb kihívásai
 - Üzleti igények hatékonyabb támogatása
 - Nem engedhető meg a fejlesztések csúszása
 - Szűkösek a költségkeretek → nincsenek tartalékpénzek a változási igényekre

Tipikus problémák

- Nincs iteráció
- Az üzlet úgy határoz meg követelményeket, hogy nem ismeri a rendszert
- Nem elég alapos a követelmények felmérése
- A rendszerterv nem egyértelmű
- A tesztelés során derül ki a tényleges működési mód
- A tesztelést lerövidítik a határidő tartásának érdekében
- Éles indulás kritikus hibákkal

Öszefoglalás – mire figyeljünk

- Minél hamarabb **ismertessük meg a felhasználót** a rendszer **működési logikájával!**
- Ne mulasszuk el a **validációt**: a rendszer **alpműködésének elfogadása** a fejlesztések korai szakaszában!
- Hagyjunk **lehetőséget a finomhangolásra!**
- Hagyjuk **elég időt a tervezésre és elemzésre!**
- A **tesztelési tevékenységet** kezdjük meg már **az előkészítő szakaszban!**



Kapcsolat

Kérdő Attila

ügyvezető, INSERO

+36 30 566 2688

kerdo.attila@insero.hu

www.insero.hu