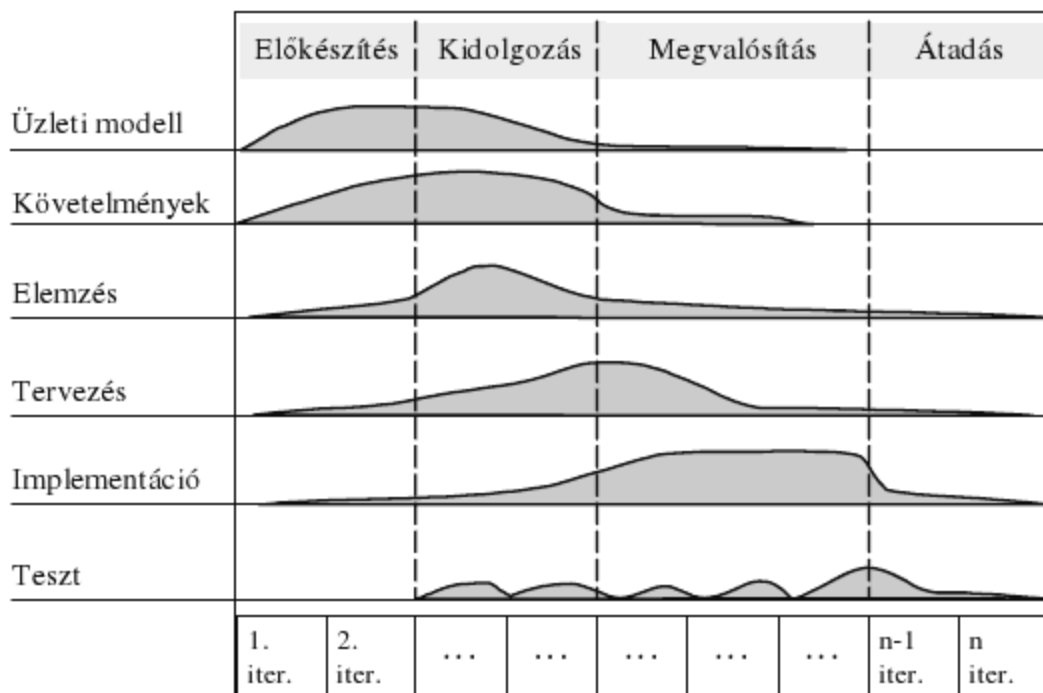


7.tétel: A RUP elemzési fázisa. Entitások, vezérlő és határosztályok. Tranziens és perzisztens osztályok



Elemzés-tervezés (Analysis & design): cél a követelményelemzés során meghatározott elvárásoknak megfelelő, robusztus rendszer tervezése.

Az előzőekben összegyűjtött követelményeket kielégítő rendszer tervezése, a rendszerterv illesztése az implementációs környezethez és a hatékonysági elvárásokhoz. Az elemzés folyamán már a *fejlesztők* nyelvezetét használjuk és a rendszer alapvető *belső* felépítését határozzuk meg.

Célja:

- a követelményeknek egy pontosabb leírását adja
- a fejlesztők számára, az ő nyelvezetükön készül, így nagyobb formalitást és egzaktságot biztosít
- a követelményekből következő alapstruktúrát rajzolja meg
 - kiindulópontja az ezt követő munkafolyamatnak, a tervezésnek

Elemzési modell

Az elemzés eredménye az *elemzési modell*, amelyben a követelményeknél megjelenő fogalmak osztályokként szerepelnek. Az elemzési modell azonban a rendszernek csak az absztrakt képe, így nem tartalmazza a nem-funkcionális követelmények megoldását, sem a tervezés során kiválasztandó programozási nyelvvel, platformmal, operációs rendszerrel, felhasznált keretrendszerrel, stb. kapcsolatos döntéseket. Az elemzési modellt osztálydiagramokkal megadott objektum-modell köré szervezzük. Az osztálydiagramok főszereplői az *osztályok*, melyeket a Unified Process az elemzés során három fő csoportba sorol, melyek külön sztereotípiákkal és ahhoz rendelt külön ikonokkal jelöl.

□ **Határosztályok** (boundary): szerepe az aktorok és a rendszer közötti kommunikáció biztosítása, általában felhasználói felületekként, esetleg külső rendszer interfészeként jelennek meg.

□ **Entitásosztályok** (entity): valóságban is megjelenő fogalmakat reprezentálnak, melyeket a rendszer tárol, együttesen a rendszer alap-adatszerkezetét adják meg.

□ **Vezérlő osztályok** (control): egy adott használati esethez kapcsolódó vezérlést, tevékenységsorozatokat, tranzakciókat valósítanak meg, melyek nem kapcsolhatók más entitásosztályhoz

Az elemzési modell osztálydiagramjain a három típusnak megfelelő osztályok, azok attribútumai és a hozzájuk közvetlenül kapcsolható műveletek, valamint az osztályok közötti kapcsolatok és egyéb (pl. általánosítás-pontosítás) viszonyok szerepelnek. A modell hierarchikus tagolására szolgálnak az *elemzési csomagok*. Egy csomagon belül elhelyezkedő osztályoknak és belső csomagoknak egymáshoz szorosan kell kapcsolódni, a külső elemekhez történő kapcsolódást pedig a minimumra kell csökkenteni.

Munkafolyamat

Architektúrális elemzés

Az architektúrális elemzés során a modellt kisebb, könnyebben kezelhető részekre bontjuk fel. Azonosítjuk a *nyilvánvaló entitás osztályokat*, a több modell-elemre vonatkozó, közös *speciális követelményeket*

Használati eset elemzés

Azonosítjuk az elemzési osztályokat, melyeket határ-, entitás- vagy vezérlő-osztályokként adunk meg, elkészítjük az elemzési objektum interakciók leírását összegyűjtjük a használati esettel kapcsolatos speciális követelményeket, így a nem-funkcionális követelményeket, amelyeket figyelembe kell venni a tervezés és implementáció során.

Osztály elemzése

Először azonosítjuk az osztály felelősségeit, melyet úgy kaphatunk meg, hogy sorra vesszük az összes szerepet, amelyben az a különböző használati esetekben megjelenik. Ezután azonosítjuk az osztály attribútumait, amelyek az osztály egy-egy jellemzőjét reprezentálják. Az attribútumok után azonosítsuk az osztályok közötti asszociációs és aggregációs viszonyokat, ezután meghatározzuk az általánosításokat, végül minden egyes osztályra összegyűjtjük és megadjuk a speciális követelményeket.

Az entity osztályok általában prezisztensek (képesek az állapotukat tárolni valamilyen háttértáron). Prezisztensnek jelölt osztálynak lehetnek tranziens(átmeneti, nem kerül tárolásra) példányai is.

Csomag elemzése

Sorra vesszük és elemezzük a csomagokat a függőségeket figyelembe véve úgy, hogy a csomagok csak a szorosan összekapcsolódó, funkcionálisan összefüggő osztályokat tartalmazzák, és a lehető legkisebbre korlátozzuk a csomagok közötti kapcsolódásokat.