



Használati esetek Use Cases

UML és a minták alkalmazása

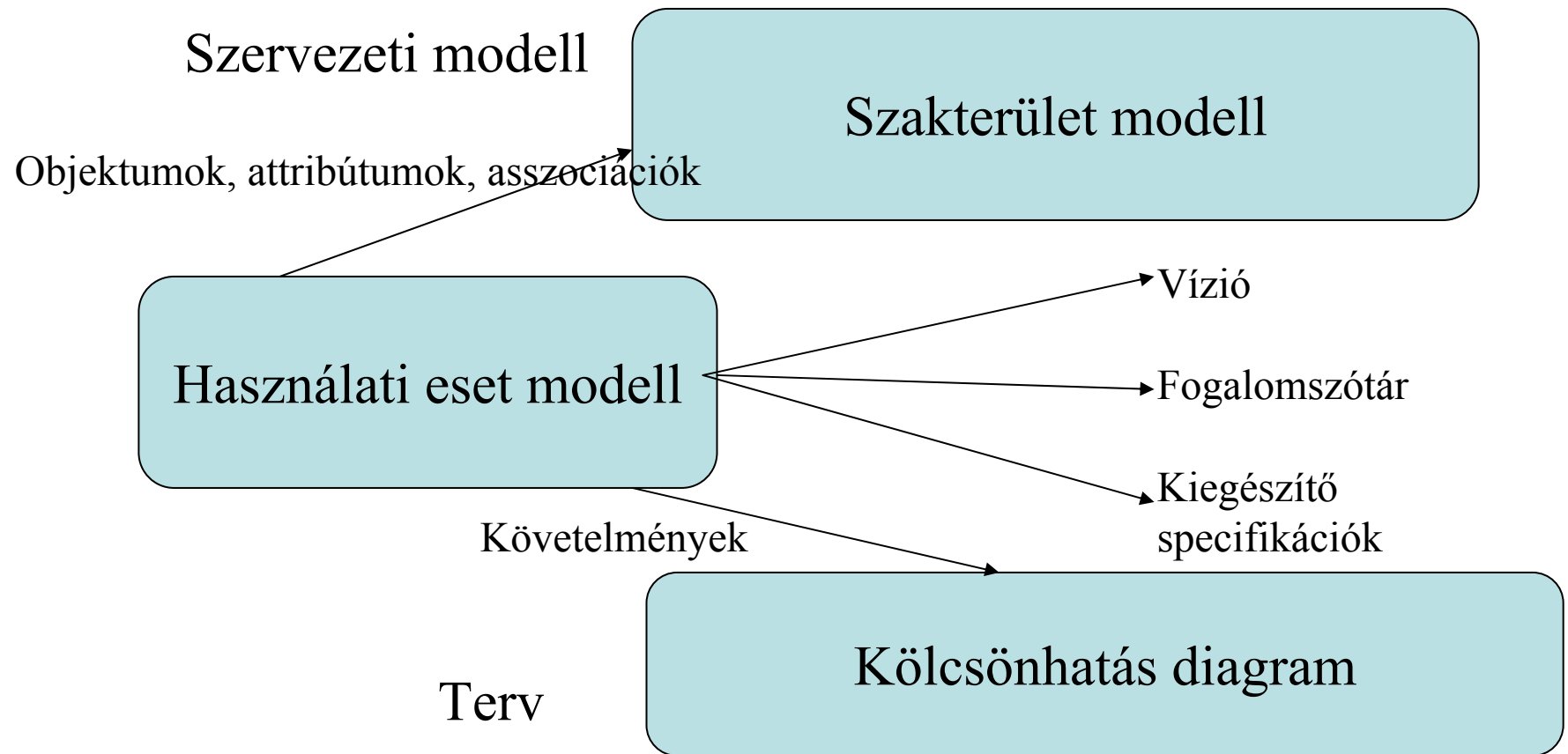
A probléma meghatározása

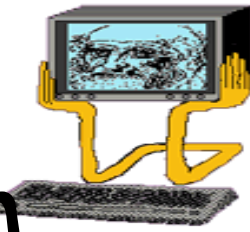


A legfontosabb kérdés:
Valóban ezt a rendszert kell elkészíteni?

("Is this the right system to make?")

Használati eset kapcsolatrendszer





Használati eset nem diagram

Használati esethez lehet diagram kapcsolva, a használati eset diagram megkönnyíti a szervezeti folyamatok megvitatását a szakterület szakértőjével

A használati eset azonban elsősorban szöveg. A szöveg fontos. A diagram opcionális.

A célok hangsúlyozása



- A célok megkeresése - a feladatok és eljárások vizsgálata helyett - nagy mértékben javítja a követelmények lényegét tartalmazó információk történő koncentrációját és azok összegyűjtését — a szándékot a feladatok és eljárások mögött.
- A követelményelemzésnek az a felfogása, hogy a feladatok felismerésére kell koncentrálni, sokkal inkább a létező rendszer reprodukálására vezet, a helyett, hogy a komolyan hiányos, hibás rendszert próbálná helyettesíteni

Miért pont használati eset kell ?



Egyszerű és ismerős történet mesélés megkönnyíti azt, hogy - elsősorban a megrendelő - hozzájáruljon a célok megfogalmazásához, és a célok felülvizsgálatához.

A használati eset leírást a lehető legegyszerűbb formában kell tartani.

A célokra és a felhasználó nézőpontjára helyezik a hangsúlyt.

A használati esetek írásában kezdők túl komolyan veszik.

Szereplők vagy használati eset először?



Mivel a használati eset részeit külön-külön kell megérteni, ezért a részeit külön-külön kell ábrázolni is. Azonban a használati eset készítésénél a készítőnek ide-oda kell váltania a részek között. A szöveges résznek a használati eset lényeges elemeit már tartalmaznia kell, amikor a figyelmet a rendszer szereplői felé fordítjuk.

Általában, mind a rendszer szereplőit (aktor) mind a használati eseteket az elemzés korai szakaszában fel kell ismerni, majd részletesebben kell vizsgálni, hogy vajon más használati esetek léteznek-e, vagy még több rendszer szereplő létezik-e?

A használati esetek felismerése



A rendszer használatának egyedi, specifikus módját olyan formában ragadjuk meg úgy, hogy a szereplő és a rendszer közötti párbeszédként írjuk le.

A használati eseteket a következőkre lehet használni:

A rendszer követelmények megragadására

A végfelhasználókkal és a szakterület szakértőjével történő kommunikációra

A rendszer tesztelésére, bevizsgálására

A használati eset megfogalmazása



Minden használati esetre kell egy dokumentumot készíteni

Világosan meg kell határozni, hogy mire szolgál a használati eset
A legjelentősebb („fő”) sikeres forgatókönyv, eset lefutás meghatározása („Happy Path”).

Az alternatív forgatókönyveket is meg kell határozni

A „kezdeményezés-válasz” (Stimulus: Response) metaforát kell használni

Mindegyik használati eset leírás megfogalmazásának, specifikációjának tesztelhetőnek kell lennie.

A szereplők szemszögéből és a szereplők szótárával kell leírni a használati esetet



www.usecases.org sablon

Név
Elsődleges szereplő
Terjedelem (Scope)
Szint
Érintett és érdekelt felek
Előfeltétel (Minimális garancia)
Utófeltétel (siker feltétel
/garancia)
A „fő” sikeres lefutás
forgatókönyve
Kiegészítések

This is the basic format
used in the text and in
*Alistair Cockburn's Writing
Effective Use Cases*
(Addison Wesley, 2000,
ISBN 0201702258).

További lehetséges elemek



Néhány további kiegészítés (A siker feltétel után)

A használati esetet kezdeményező dolog („trigger“)

Különleges/speciális követelmények

Technológiai és adat variációk

Bekövetkezés/ végrehajtás gyakorisága

Nyitott kérdések

A bevezető rész elemei



Csak olyan elemeket érdemes ide tenni, amelyek érthetőek a „fő” sikeres forgatókönyv elolvasása nélkül is. Ide tartoznak:

Megnevezés(*azonosítás céljára*)

Elsődleges szereplő

Érintett és érdekelt felek listája

Előfeltétel

Sikeresség feltétele (Utófeltétel)

Használati esete elnevezése



A végfelhasználó szempontjából egy teljes folyamatnak kell lennie.

Általában TÁRGY (főnév) - IGE (névszói alak), „Pizza vásárlás”

Elegendően részletes legyen ahhoz, hogy az adott esetre vonatkozzon.

A szereplő szempontjából és nem a leendő rendszer szempontjából írja le.

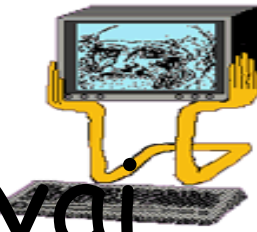
Útmutatás



A használati eseteknek adott kifejező nevek fontosak. Mivel a projekt korai szakaszában jönnek létre, irányt szabnak az egész projektnek.

A használati esetek készítésekor tipikus hiba: a nem kifejező nevek adása, az eljárások ábrázolása a teljes felhasználói folyamatok helyett, és a rendszer nevének és határainak elhagyása.

A használati esetek elnevezésének arany szabályai



Mindegyik használati esetnek olyan nevet kell adni, amelynek jelzi, hogy milyen értéket vagy célt ért el a szereplő akkor, amikor érintkezésbe lépett a rendszerrel.

Néhány kérdés megfogalmazás, amely segít a szabályok betartásában.

Miért kezdeményezné a szereplő ezt az érintkezést a rendszerrel?

Milyen cél elérés van a szereplő fejében akkor, amikor ezeket a lépéseket megteszi?

Milyen érték jött létre és mely szereplő számára?

(Dr. Use Case (Leslee Probasco) in the Rational Edge, March, 2001)

Példák a használati esetek elnevezésére



Kiváló - Hangverseny jegy vásárlása

Nagyon jó - Hangverseny jegyek vásárlása

Jó - Jegyek vásárlása (nem elegendően részletes)

Közepes - Jegy vásárlandó (passzív)

Gyenge- Jegy rendelés (a rendszernézet, nem felhasználói)

Elfogadhatatlan - Fizetés a jegyért (eljárás, nem folyamat)

CRUD mátrix - LOAT



Példák rossz használati eset elnevezésekre a LOAT -ra támaszkodva (mindegyik procedurális és nem fed fel semmit a szereplő szándékaiból)

L (C) - szereplő Létrehoz (Creates) adatot

O (R) - szereplő Olvas(Read/ Retrieves) adatot

A (U) - szereplő Aktualizál (Updates) adatot

T (D) - szereplő Töröl (Deletes) adatot

Egyes szám vagy többes szám



Ezt általában a (szöveg) környezet határozza meg.

Előnyben kell részesíteni a legegyszerűbb formát, minél ismertebb, elterjedtebb annál jobb

A hangverseny jegy vásárlási példánál, a legtöbb ember több jegyet vásárol, de egy lényeges rész csak egy jegyet

Egy szupermarketben: Áruk vásárlása (a legjobb)

Egy automatánál: Áru vásárlása.

A szereplők felismerése



Addig nem tudjuk megérteni azt, hogy a rendszer, hogyan működik, amíg meg nem tudjuk azt, hogy ki fogja használni

Közvetlen felhasználók (végfelhasználók)

Felhasználók, akik felelősek a működtetésért és karbantartásért/napra készen tartásért

A rendszer által használt külső rendszerek

A rendszerrel érintkezésbe lépő külső rendszerek

Szereplők meghatározása



A szereplők a rendszeren kívül vannak

A szereplők nem - determinisztikusak viselkednek

Mik lépnek érintkezésbe a rendszerrel?

A szereplők valójában ugyanannak az egyénnek különböző szerepkörei lehetnek, amelyek érintkezésbe lépnek a rendszerrel

Szereplők más rendszerek is lehetnek

Nem tételezhető fel, hogy a Szereplő = Egy egyénnel

Szereplők



Elsődleges szereplő

Olyan célja van , amelyet a rendszernek kell teljesítenie

Támogató szereplő

A rendszernek nyújt szolgáltatásokat

Színen kívüli szereplő

Érdekelt/érintett a rendszer viselkedésében, de nem járul hozzá

A diagramokban, az elsődleges szereplőket balra, a többi jobboldalra rajzoljuk In diagrams,

A szereplők leírása



A szereplőket nem kell elemezni, vagy részletesen leírni, hacsak ezt az alkalmazási terület meg nem kívánja.

A szereplők leírásának sablonja

Név

Leírás/meghatározás

Példa egy ATM alkalmazásra:

Ügyfél: Folyószámla tulajdonosa aki kezeli a folyószámlát, pénzbetét elhelyezéssel illetve felvétellel

Szereplők azonosítása



Elsődleges szereplő

A hangsúly az a használati eset elsődleges szereplőjén van.

Érdekelt/érintett

A többi szereplőt mint érintett felet soroljuk fel.

A kulcsfontosságú szereplők érdekeltségi területeit le kell írni

Használati esetek készítése



- Határozzuk meg a szereplőket, akik érintkezésbe lépnek a rendszerrel
- Elemezzük a szereplők szükségleteit és dokumentáljuk
- Az elkülönülő igényekre hozzunk létre egy-egy használati esetet
- Az elemzés során terjesszük ki a használati esetet kölcsönhatás diagrammal

Előfeltételek



Bármilyen olyan feltétel, amelynek igaznak kell lennie mielőtt a forgatókönyv végrehajtását megkezdenék, előfeltétel.

Előfeltételek fennállását feltételezik, a használati eseten belül nem vizsgálják.

A nyilvánvaló előfeltételeket célszerű kihagyni (tápfeszültség bekapcsolva). Csak azokat az elemeket kell dokumentálni, amelyek megértése kell a használati esethez

Utófeltételek, sikeres lefutás



Utófeltételek, a sikeres lefutás feltételei azt érzékeltetik, hogy milyen állapotoknak kell igazaknak lenniük, ha a használati eset sikeresen befejeződött. A sikeres „fő” forgatókönyv és néhány alternatív lefutás eredménye is ide tartozik. Például, ha a sikeres lefutás készpénzes eladás, a hitelre történő értékesítés is azonban sikernek tekinthető.

Az érintett feleknek egyet kell érteniük a sikerességet jelentő utófeltételben.

Forgatókönyv



A „fő” sikeres forgatókönyv a rendszer elvárt, elsődleges használata, problémák, rendkívüli események, kivételes eljárások nélkül.

Az alternatív *forgatókönyveket* vagy *kiegészítéseket*, arra használják, hogy más szokásos lefutásokat dokumentáljanak, beleértve a hiba kezelést és a kivételes eljárásokat.

A sikeres lefutás dokumentálása



A sikeres forgatókönyv (vagy alapeset) a használati eset megértését nyújtja

Mindegyik lépés tartalmazza azokat a tevékenységeket, bemeneti adatokat, amelyekre a szereplő és a rendszer válaszol.

Ha van legalább három vagy több elem, akkor készíts listát.

Az egyes lépéseket lásd el címkével a későbbi konfiguráció kezelési feladatok és a követelmények nyomon követhetősége érdekében.

Használj jelen időt

Ne feledd, a felhasználói felület tervezője fogja ezt a leírást használni.

Kiegészítések (alternatív lefutások)



A használati eset kiegészítések vagy alternatív lefutások, lehetővé teszik azt, hogy le lehessen írni

- A tranzakciók különböző kezelését
- A hiba feldolgozást

A használati eset szekciókra, fejezetekre bontása az alternatív lefutások leírásának kényelmes módja

A szekciók, fejezetek olyan használati eset részek, amelyeket a fősodoron kívül hajtanak végre.

A kiegészítések két része



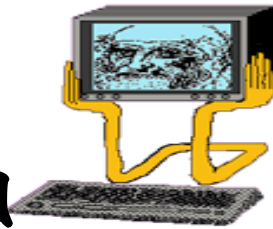
Feltételek

Az alternatív lefutás okát írják le mint feltételt, és amelyet a felhasználó észlelni tud.

Kezelés

A feldolgozási folyamat ugyanolyan leírása mint a főforgatókönyvé, az eredeti szekcióval összhangban álló számozási eljárás alkalmazása.

A kiegészítések dokumentálása



A sikeres lefutás leírásával azonos formátum használata
Azokat a tevékenységeket kell dokumentálni, amelyek
eltérnek a sikeres lefutástól

Beleértve a hiba feltételeket

Az egyes alternatívákat meg kell számozni, és a
feltételekkel kell kezdeni:

3A. Feltétel: HA [*szereplő*] végrehajt [*tevékenységet*] a rendszer ...

Ha következő lépések megegyeznek a sikeres
lefutáséival, akkor az egyeztessük össze azokat és
címkézzük meg egyformán

Azokat a lépéseket, amelyeket nem emelünk át az
alternatív lefutás leírásába - feltételezzük -, nem
hajtják végre

Különleges követelmények



Ha egy nem-funkcionális követelmény közvetlenül befolyásolja a használati esetet -minőségi tulajdonság, peremfeltétel stb. -, akkor azt mint egy különleges követelményt kell rögzíteni.

A technológia és az adatok változatainak listája



Gyakran vannak műszaki különbségek abban, hogy hogyan végeznek el bizonyos dolgokat, mégis lényegében ugyanazt csinálják. Ezeket a variációkat lehet rögzíteni, a technológia és az adatok változatainak listájában.

Például, ha egy kártya-olvasó nem tudja el olvasni a mágnes csíkot a bankkártyán, akkor a pénztárosnak a billentyűzetről kell tudnia bevinni az adatokat

A használati esetek típusai



A legáltalánosabban alkalmazott használati esetek, a „Magas szintű használati eset” (High Level Use Cases), „ Kibontott lényegi használati eset” (Expanded Essential Use Cases), és a tervezésben „ Kibontott valóságos használati eset”. A definíciók a következők dián.

Továbbá, a használati eset diagrammokat a követelmények megragadása során az érintett felekkel folytatott megbeszéléseken alkalmazzák.

A használati esetek kidolgozása



Magas szintű használati eset (rövid, tömör)

Megnevezés, Szereplők, Cél, Áttekintő leírás

Kibontott használati eset (teljesen kidolgozott)

A rendszer eseményekkel és a válaszokkal bővített

Lényegi használati eset (fekete doboz)

A technológia választásból származó követelmények
következményeinek kihagyása

Valóságos használati eset (fehér doboz)

A technológia választásból származó követelmények
következményeinek beillesztése

Megjegyzés a nevezéktanhoz



A használati esetek alkalmazása széles körben elterjedt a szoftveriparban és a megnevezések gyakran változtak. Ezért hasznos ha a különböző, alternatív megnevezésekkel tisztában van mindenki (kibontott kontra teljesen kidolgozott, magas szintű kontra tömör) (expanded vs. fully dressed) (high level vs. brief).

A technológia gyorsan változik, és ezért szükséges napra kész ismeretekkel rendelkezni.

A döntések elhalasztása



A lényegi használati eseteket addig alkalmazzuk, amíg csak lehetséges és csak a (szoftver)modulok tervezése során térjünk át a valóságos használati esetek alkalmazására, így időt hagyunk a probléma megértésére mielőtt a valóságos használati eseteket elkészítjük. A valóságos használati esetek alkalmazása gyakran a létező technológia megerősítéséhez vezet akkor is, amikor jobb technológia is rendelkezésre áll.

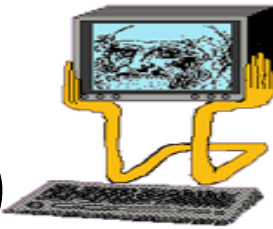
Technológia



A lényegi (fekete doboz) - ahol kihagyjuk a technológiát -, és a valóságos (fehér doboz)- ahol figyelembe vesszük a technológiát -, használati eset megkülönböztetése alapvetően lényeges.

Például, egy ATM esetében, a lényegi használati eset csak az azonosításra és az ellenőrzésre utal, míg a valóságos használati eset utalhat a billentyűzetre és a kártya olvasóra.

Kibontott lényegi használati eset (Teljesen kidolgozott használati eset)



Cél:

A rendszertervezőnek és a megrendelőnek lehetővé kell tenni, hogy vizuálisan megjelenítve láthassa a „szereplő tevékenységét” és a „rendszer válaszait”. A megrendelő meg fogja érteni azt, hogy a felhasználó hogyan fogja használni, és a tervező pedig mindegyik funkcióra meg fogja tudni írni a pszeudo kódot. Továbbá, e dokumentum segítségével a lehetséges felhasználói hiba lehetőségeket fel lehet tárni, amelyeket a végleges rendszernek le kell tudnia kezelni.

Definíciók:

Mi ez: egy elemzési dokumentum, amely részleteiben leírja azokat a funkciókat, amelyeket „Magas szintű használati eset” -ben feltártak.

Mi **nem**: Kibontott lényegi használati eset nem grafikus diagram. Nem tartalmaz pálcika emberkéket, a rendszert érzékeltető téglalapokat, vagy egyéb más ikonografikus szimbólumokat, amelyek azonban a „Magas szintű használati eset” -ben megjelenhetnek, és esetleg valamelyikükhöz hozzá is kapcsolódhat

Kibontott lényegi használati eset



Hogyan készítsünk egyet:

1. lépés: Nevezd meg a használati esetet (rendszer funkció, pl. vidd be a számolótáblához az adatokat).
2. lépés: Az érintett szereplőket azonosítsuk.
3. lépés: A Megrendelő számára érthető nyelven írjuk le, hogy mire szolgál, mire szánják a használati esetet.
4. lépés: Az adott használati esethez vagy más olyan használati esetekhez tartozó feltételezéseket és korlátozásokat is fel kell tárni, amelyek az adott használati eset kibontásával keletkeznek, vagy amelyekre a jelenlegi használati esete támaszkodik.
5. lépés: Írjuk le az ideális tevékenység sorozatot, két oszlopban, az egyik oszlop fejléce „Szereplők tevékenysége”, a másik „Rendszer válaszok”. Számozzunk be minden lépést. Ennek a használati esetnek ez a sikeres lefutása.
6. lépés: Tárjuk fel a lehetséges végfelhasználói hibákat, és készítsünk az összes lekezelésére egy alternatív lefutást.

Megjegyzés



Általában a használati esetek egy vagy két oldal hosszúak.

Utófeltételek



Utófeltételek (vagy siker garanciák) olyan állítások, amelyek a használati eset sikeres lefutása esetén mindig igaznak kell lennie. A nyilvánvaló megfogalmazásokat kerüljük, de a nem nyilvánvalókat világosan kell dokumentálni. Ez a használati eset leírások legfontosabb része.

Feltételek és elágazások



Ragaszkodjunk a sikeres lefutáshoz, az alap esethez, a forgatókönyv fő részéhez, és a feltételekhez tartozó részeket és az elágazásokat a kiegészítésekben vagy alternatív lefutásokra hagyjuk.

Felbontás



Ha a használati esetet megismétlődik,
akkor nem kell a tartalmát megismételni.

Például, egy ébresztő óra ugyanazt a
megjelenítő felületet mutatja akkor,
amikor a pontos időt állítjuk be és akkor
is, amikor az ébresztőt.

Válasszuk le „Az idő megjelenítése”
használati esetet és hivatkozzunk
mindkét használati esetben rá

Használati esetek kiegészítése



A felhasználók az egyszerűséget értékelik, ezért a legtöbb használati esetből kihagyják az alternatív lefutásokat.

Ez úgy érhető el, hogy az eredeti használati esetet változatlanul hagyjuk, és kiegészítésekkel bővítjük ki.



Figyelmeztetés

Használati eseteket nem szabad arra használni, hogy funkció leírásokat utánozzák, az egymás után következő iterációkban.

A használati esetek közötti „felhasznál” kapcsolatot csak akkor szabad használni, ha ugyanazt a forgatókönyv többször is előfordul.

A (rendszer) sajátosságok („features”) listája



A régebbi, hagyományos módszerek a követelmények részletezését több oldalon keresztül tartalmazták. Azonban a részletek gyakran nem voltak láthatók a szövegekörnyezetben.

Jelenleg az a (szakmai) filozófia, hogy a használati esetek formájában egy magasabb szintű megközelítést használjanak, a lista helyett.

Magas szintű rendszer sajátosságok listája, akkor elfogadható, ha a rendszer tömör összefoglalását tudja adni

Használati eset nem O-O



Használati eset nem O-O módszer. Sok strukturált fejlesztési környezetben is használják

Az egyesített folyamat modell bátorítja a használati eset alapú rendszerfejlesztést.

Használati esetek szintjei



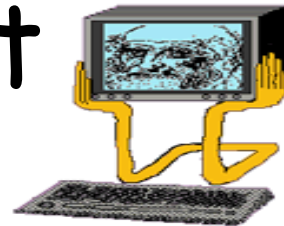
Felhasználói célok szintje

Egy teljes folyamat a felhasználó szemszögéből, durván megfeleltethető egy elemi szervezeti folyamatnak („ügyviteli”, „üzleti”).

Részfunkció szint

Azokat a lépéseket részletezi, amelyek egy felhasználói cél megvalósításához szükségesek.

Használati eset által vezérelt fejlesztés



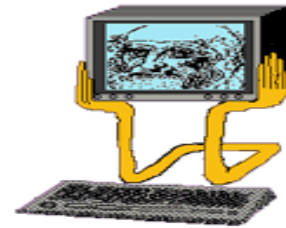
A követelményeket elsődlegesen a használati eset modellben rögzítjük.

A fejlesztési projektbeli iterációkat bizonyos használati esetek köré szervezzük.

Használati esetek megvalósítása/realizálása vezérli a tervezést.

A használati esetek befolyásolják azt a módot, ahogy felhasználói kézikönyveket szerkesztik össze.

A használati esetek mindig rosszak!



Az írott dokumentumok gyakran azt az illúziót keltik, hogy helyesek és szakmai tekintély áll mögöttük, azonban ez csak valóban illúzió.

Használati esetek a probléma terület egy elsődleges megértését mutatják, amelyet a felhasználók és fejlesztők meg tudnak vitatni és utána egyetértést kialakítani.

Azonban folyamatos visszajelzésre van szükség a felhasználók/megrendelő oldaláról azért, hogy a fejlesztés során folyamatosan korrigálni lehessen a hiányzó információkat és a téves információkat mielőtt veszélyeztetnék a rendszer (program) funkcionalitását.

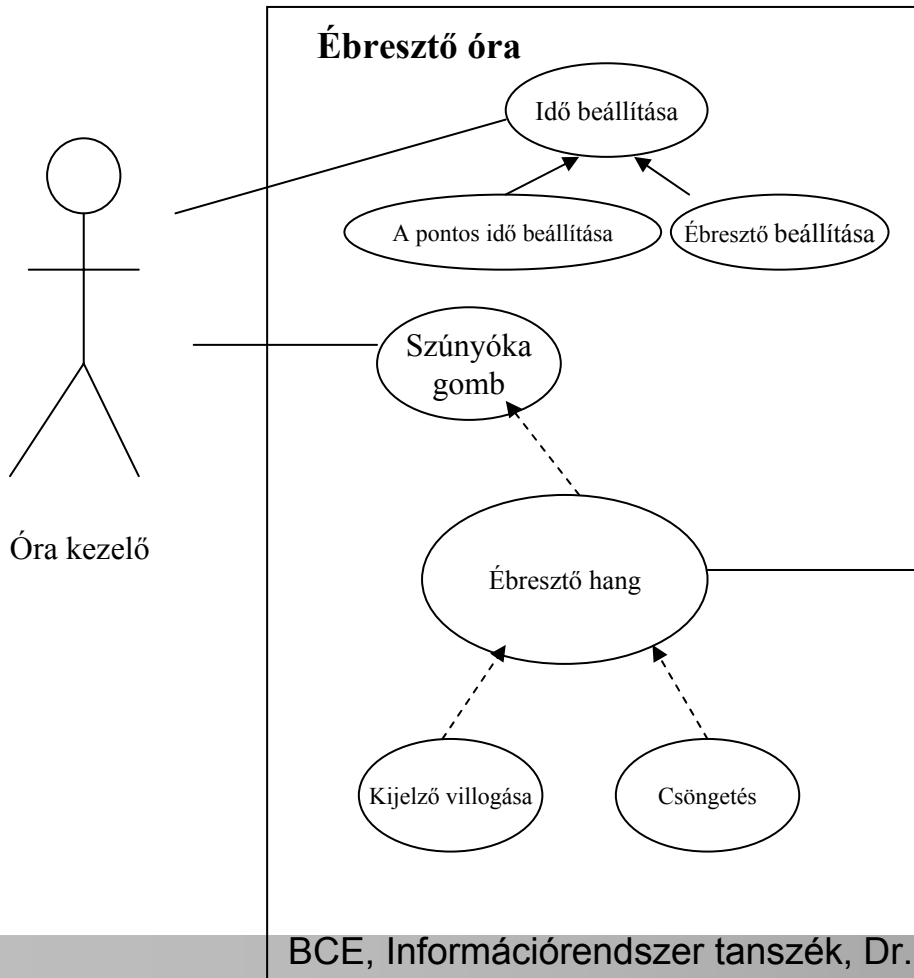


A használati esetek diagram (rajzolási) technikája

A használati eset a szöveg maga!

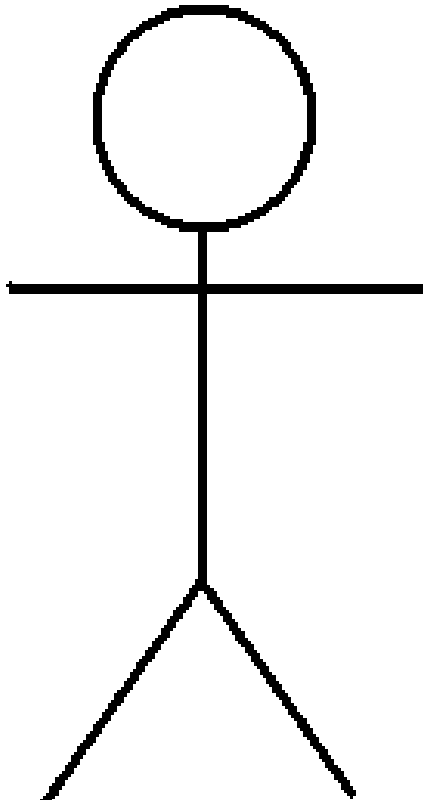
A diagramok csak kiegészítik a
szöveget vagy segítenek a
követelmények felderítésében.

Áttekintő leírás



A használati eset diagram azonosítja a tranzakciókat a rendszer szereplők és a rendszer között mint egyedi használati eseteket.

Szereplő



Tanuló

A szereplő a rendszer egy idealizált felhasználója.

Szereplők lehetnek a felhasználók, folyamatok, és más rendszerek

Sok felhasználót össze lehet fogni egy szereplőben, egy közös szerepkörben.

Egy felhasználó különböző szereplőként jelenhet meg, a különböző szerepkörei miatt.

A szereplőt a szerepköre alapján nevezik el.

Nem ember szereplők



Raktár

Rendszer

Szereplők lehetnek a felhasználók, folyamatok, és más rendszerek

A nem ember szereplőket, más módon kell ábrázolni, gyakran egy téglalappal

A nem ember szereplők általában nem elsődleges szereplők, és ezért általában az ábra jobb oldalán jelennek meg.

Használati eset



A használati eset egy a rendszer által nyújtott kívülről látható funkcionalitás koherens egységbe rendezése és az üzenetek egy sorozatával lehet leírni.

A további viselkedési minták a szülő-gyermek , kiterjeszt és a tartalmaz használati esetek közti kapcsolatokkal írható le. Olyan nevet kell adni, amit a felhasználók megértenek.

Követelmény diagram készítése

Rendszer



Használati eset modellező

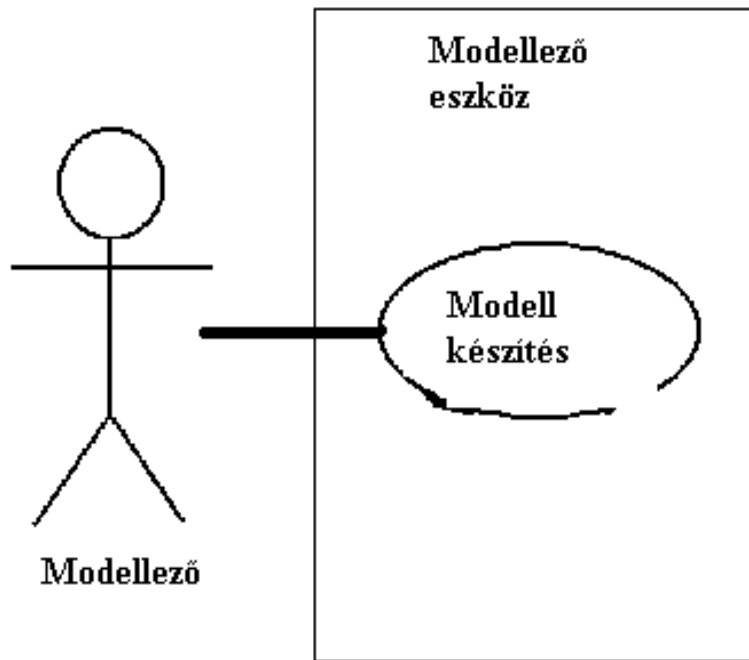
A rendszert a téglalap ábrázolja, amelyet a rendszer nevével címkéztek meg.

A szereplők a rendszeren kívül vannak.

A használati esetek a rendszeren belül vannak.

A téglalap mutatja a rendszer kiterjedését és a határát.

Asszociációs kapcsolat



Az asszociáció a kommunikációs útvonalat fejezi ki a szereplő és használati eset között.

Egy tömör vonallal jelöljük. A végén nincs nyílhegy, és általában balról jobbra olvasandó.

Itt a modellező és modell készítés közti asszociációt mutatja az ábra

Használati esetek kapcsolatai



A következő kapcsolatok léteznek a használati esetek között:

Asszociáció

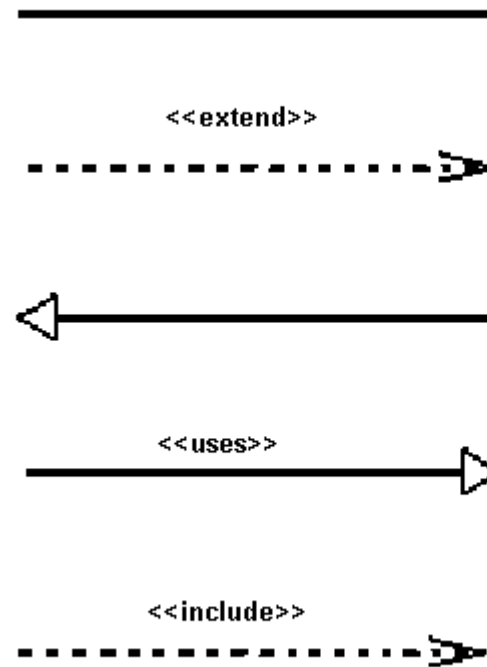
Kiterjeszt (extend)

Általánosítás
(Generalization)

Használ (Uses)

Tartalmaz (Include)

A legtöbb használati eset csak az asszociációt használja. A többi kapcsolatot csak nagyon visszafogottan szabad alkalmazni.



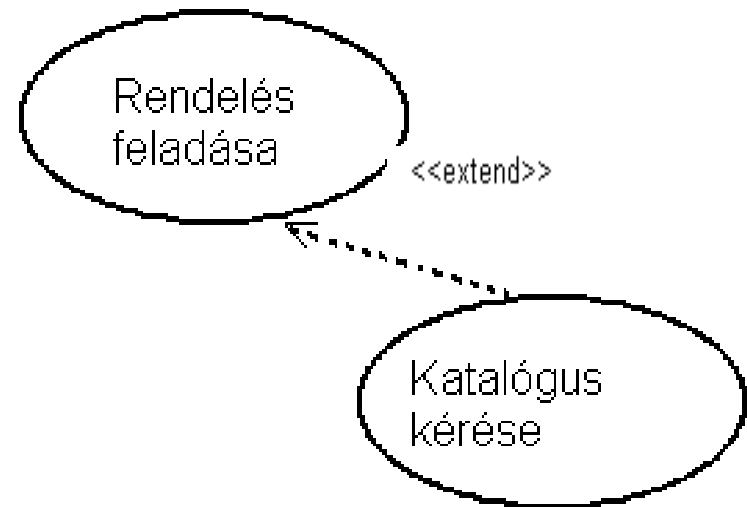
Kiterjeszt kapcsolat



A kiterjeszt („Extend”) további viselkedési mintákat helyez el egy olyan használati esetben, amelyik arról nem tud.

Szaggatott vonallal és nyíllal jelöljük és kap egy címkét `<<extend>>`

Ebben az esetben, a megrendelő kérhet egy katalógust miközben felad egy megrendelést.



Használati eset általánosítás

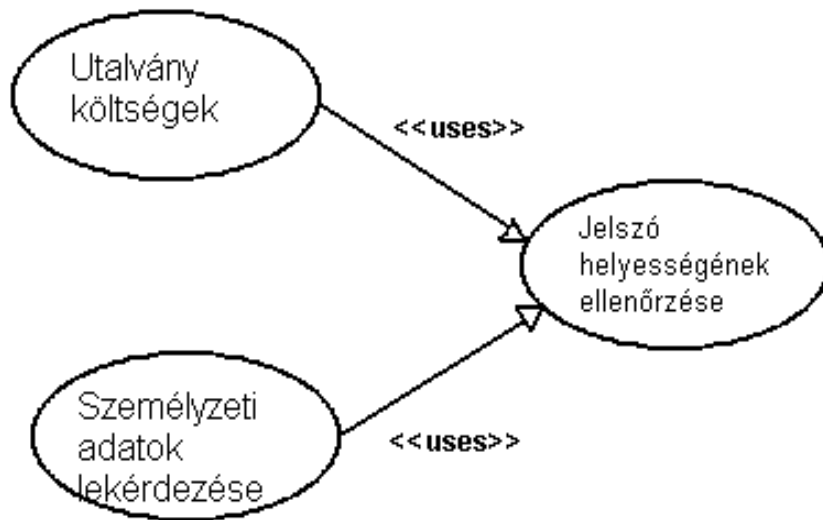


Az általánosítás egy általános és egy sokkal specifikusabb használati eset közti olyan kapcsolat, amelynek révén a specifikus megörökli és ugyanakkor kiterjeszti az általános sajátosságait.

Tömör vonallal jelölik és zárt nyílhegygel.

Itt, a fizetési folyamatot módosítják aszerint, hogy készpénzes vagy kártyás fizetés történik.

Használ kapcsolat



Amikor a használati eset egy másik folyamatot használ akkor köztük fennálló kapcsolatot a „használ” kapcsolattal lehet érzékeltetni.

Ezt a kapcsolatot egy tömör vonallal és zárt nyílhegygel jelöljük, valamint kap egy címkét a <<uses>> kulcsszót.

Itt, a különböző rendszer folyamatok, a bejelentkezés használati esetét használják.

Tartalmaz kapcsolat



Tartalmaz kapcsolat az alap használati esethez illeszt be egy további viselkedési mintát.

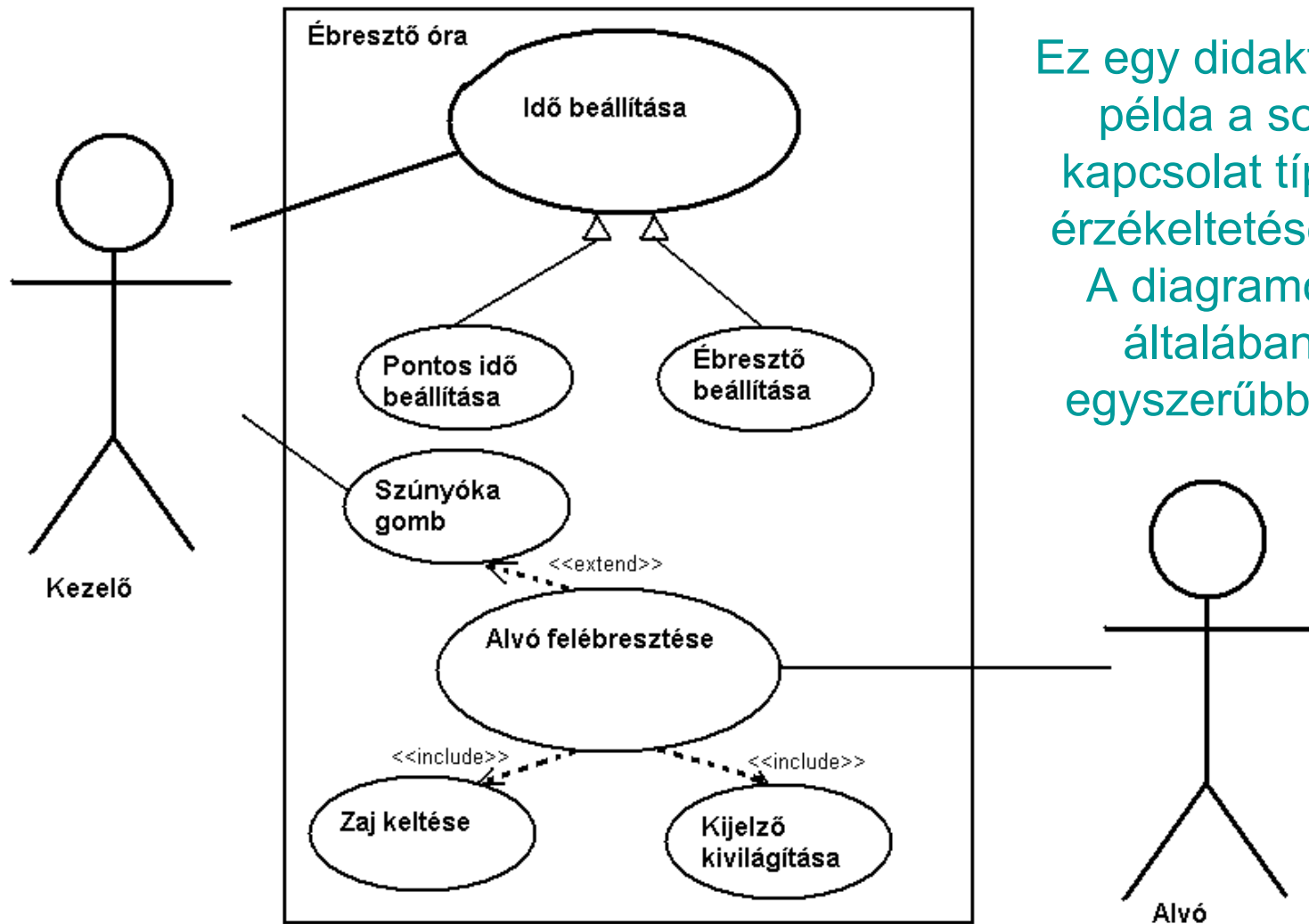
Ezt a kapcsolatot szaggatott vonallal és nyitott fejű nyílhegygel jelöljük, címkével látjuk el, az

`<<include>>` kulcsszóval

Egy tipikus viselkedési mintázatot mutató használati esete látható, angolszászok fizetésnapon.



Használati eset példa: Ébresztő óra



Ez egy didaktikus példa a sok kapcsolat típus érzékeltetésére. A diagramok általában egyszerűbbek.