

1.

Bevezetés a szoftvertechnológiába

Kérdések

- Mi a szoftvertechnológia (szoftvermérnökség)?
- Mik a szoftvertechnológiát érintő legfontosabb kérdések és válaszok?
- Etikai és szakmai kérdések: hogyan érintik ezek a szoftvermérnököket?

Tartalom

- Szoftvergyártás GYIK
- Szakmai és etikai felelősség

Szoftvertechnológia

- MINDEN fejlett ország gazdasága szoftverfüggő
- Egyre több rendszert szoftver irányít
- A fejlett országok nemzeti jövedelmük jelentős részét szoftverre költik
- A szoftvertechnológia a professzionális szoftvergyártás elméleti módszereivel és eszközeivel foglalkozik

A szoftver ára

- A számítógépes rendszerek árában általában dominál a szoftver költsége. Egy PC-n futó programok ára jóval magasabb, mint a hardver költsége.
- A szoftver fenntartása többbe kerül, mint a kifejlesztése. Hosszú élelciklusú rendszerek esetén a fenntartási költségek a fejlesztési költség többszörösére rúgnak.
- A szoftvertechnológia egyik célja a költséghatékony szoftverfejlesztés.

Szoftvermérnökség GYIK

- Mi a szoftver?
- Mi a szoftvermérnökség, ill. szoftvertechnológia?
- Mi a különbség a szoftvermérnökség és a számítástudomány között?
- Mi a különbség a szoftvermérnökség és a rendszermérnökség között?
- Mi a szoftvergyártás?
- Mi a szoftvergyártás modellje?



Szoftvermérnökség GYIK

- Mik a szoftvergyártás költségtenyezői?
- Milyen szoftvergyártási módszerek vannak?
- Mi a CASE (Computer-Aided Software Engineering)?
- Mik a jó szoftver ismérvei?
- Mik a szoftverkészítés legfőbb nehézségei?



Mi a szoftver?

- Számítógépes programok és a hozzá kapcsolódó dokumentációk (pl. követelmények, tervezési modellek és felhasználói kézikönyvek)
- Szoftver termékek készülhetnek egy meghatározott vevő, vagy egy széles piac számára:
 - Általános (generic)– felhasználók széles rétege számára fejlesztett és általuk használt szoftver. Pl. Excel, Word
 - Egyedi (custom) – egy megrendelő egyedi igényei szerint készült
- Új szoftver készülhet új programok írásával, általános szoftver konfigurálásával, vagy létező szoftver újrafelhasználásával

Mi a szoftvermérnökség, ill. szoftvertechnológia?

- A szoftvertechnológia olyan *mérnöki tudomány*, amely a szoftvergyártás *minden kérdésével* foglalkozik
- A szoftvermérnöknek munkája magas színvonalú végzéséhez *szisztematikus tervezési elveket*, valamint a megoldandó probléma, a fejlesztési kényszerek és felhasználható erőforrások függvényében megfelelő *eszközöket és technikákat* kell használnia.

Mi a különbség a szoftvermérnökség és a számítástudomány között?

- A számítástudomány az alapvető elméleti kérdésekkel, míg a szoftvermérnök a jól használható szoftver kifejlesztésének praktikus kérdéseivel foglalkozik
- A számítástudomány ma még nem képes a szoftvermérnökség megfelelő elméleti alátámasztására (ellenpélda: fizika és villamosmérnöki tudományok)

Mi a különbség a szoftvermérnök és a rendszermérnök között ?

- A rendszermérnök számítógép-alapú rendszerek tervezésének minden kérdésével foglalkozik, beleértve a hardver-, szoftver- és folyamatmérnöki teendőket. A szoftvermérnök ennek egy részfolyamataért felelős, pl. a szoftver infrastruktúra, alkalmazások, adatbázisok fejlesztése.
- A rendszermérnökök részt vesznek a rendszer specifikációjában, a rendszertervezésben, a rendszerintegrációban és a beüzemelésben.

Mi a szoftvergyártás?

- Olyan tevékenységek sorozata, amelyeknek célja a szoftver kifejlesztése és továbbfejlesztése (evolúciója).
- A szoftverfejlesztés általános lépései:
 - Specifikáció – mit kell a rendszernek tudnia és mik a fejlesztési kényszerek, kötöttségek
 - Fejlesztés – A szoftver rendszer megalkotása
 - Validáció – ellenőrzés: a szoftver azt csinálja, amit a megrendelő akar?
 - Evolúció – A szoftver változó igények szerinti továbbfejlesztése.

Mik a szoftvergyártás modelljei?

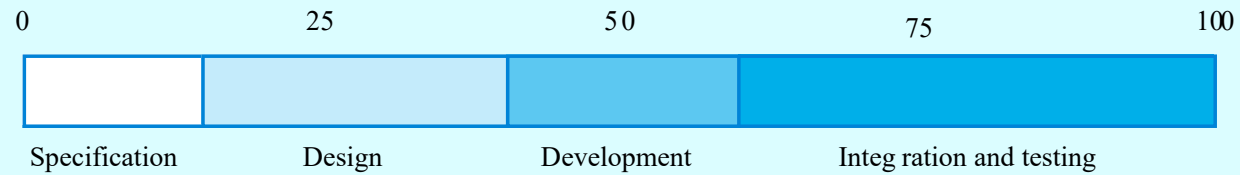
- A szoftvergyártás *egyszerűsített* reprezentációja, egy specifikus *nézőpontból*.
- Nézőpontok (példák):
 - Munkafolyam (workflow) nézőpont – aktivitások sorozata;
 - Adatfolyam (data-flow) nézőpont – az információterjedés folyamata;
 - Szerep/akció (role/action) nézőpont – ki mit csinál?
- Általános modellek
 - Vízesés (Waterfall);
 - Iteratív fejlesztés;
 - Komponens alapú szoftverfejlesztés.

Mik a szoftvergyártás költségtényezői?

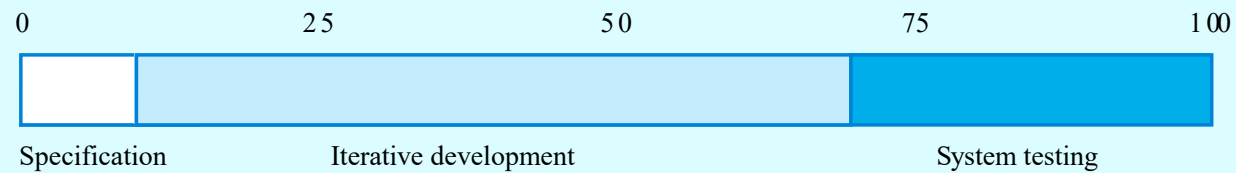
- A költségek kb. 60%-a fejlesztési, 40%-a tesztelési költség.
- Egyedi szoftverek esetén az evolúciós költségek gyakran meghaladják a fejlesztési költséget.
- A költség függ a fejlesztett rendszer típusától és az elvárásoktól (pl. a rendszer megbízhatósága és teljesítménye).
- A költségek eloszlása függ az alkalmazott fejlesztési modelltől.

A költségek eloszlása

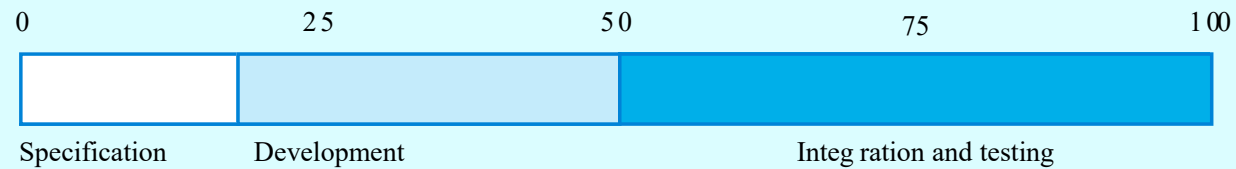
Waterfall model



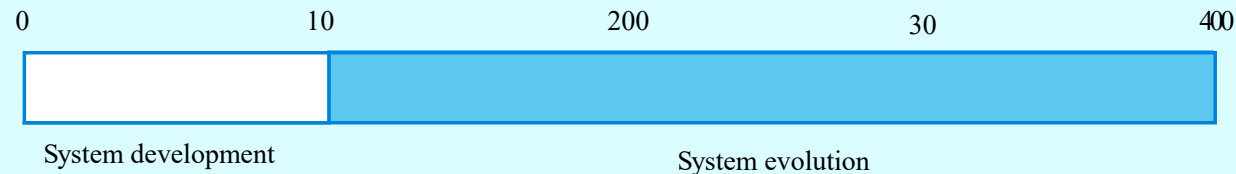
Iterative development



Component-based software engineering

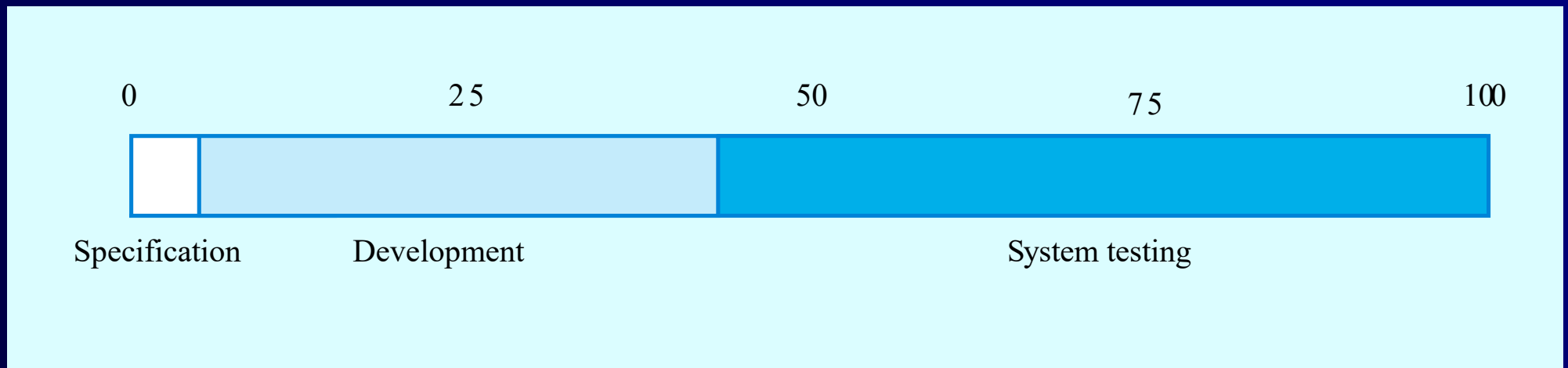


Development and evolution costs for long-lifetime systems



Egyedi SW

A rendszerfejlesztési költségek eloszlása



Általános SW

Mik a szoftverfejlesztési módszertanok?

- Olyan strukturált szoftverfejlesztési módszerek, amelyek tartalmazznak rendszermodellező eszközöket, jelölési konvenciót, szabályokat és tervezési ajánlásokat, valamint fejlesztési útmutatót.
- Modell leírások
 - A létrehozandó grafikus modellek leírása;
- Szabályok
 - A rendszermodellekre vonatkozó kényszerek;
- Ajánlások
 - A helyes tervezési megoldásokra vonatkozó tanácsok;
- Fejlesztési útmutató
 - A modellfejlesztés során végrehajtandó tevékenységek sorozata

Mi az a CASE (Computer-Aided Software Engineering)?

- Olyan szoftver rendszerek, amelyek a szoftverfejlesztési folyamatot automatikus eszközökkel támogatják
- A CASE rendszerek rendszerint egy metodikát támogatnak.
- Upper-CASE
 - A fejlesztés korai fázisait támogató eszközök (követelmény-analízis, tervezés);
- Lower-CASE
 - A fejlesztés későbbi fázisait támogató eszközök (programozás, hibakeresés, tesztelés) .

Mik a jó szoftver ismérvei?

- A felhasználó által megkívánt funkcionalitást és teljesítményt szolgáltatja, jól karbantartható, megbízható, hatékony és befogadható.
- Karbantarthatóság
 - A szoftvernek a változó igényekkel lépést kell tartania;
- Megbízhatóság
 - A szoftvernek megbízhatóan kell működnie;
- Hatékonyság
 - Ne bántson pazarlóan a rendszererőforrásokkal;
- Befogadhatóság/használhatóság
 - A célfelhasználók által befogadhatónak kell lennie: érthető, könnyen használható, más rendszerekkel kompatibilis.

Mik a szoftverkészítés legfőbb kihívásai?

- XXI. század: heterogenitás, határidők, bizalom
- Heterogenitás
 - Szoftverkészítést heterogén platformokra és végrehajtási környezetekre.
- Határidők
 - Gyorsabb fejlesztés és átadás.
- Bizalom
 - Felhasználók bizalmát megnyerni képes fejlesztési technológia.

Szakmai és etikai felelősség

- A szoftvermérnöknek nem csak technikai tudásának alkalmazásáért kell felelősséget vállalnia.
- Egy szakmailag megbecsült szoftvermérnök viselkedése a becsület és etika vezérelveit kell kövesse.
- Az etikus viselkedés több, mint a törvény betűjének betartása.

A szakmai felelősség kérdései

- Titoktartás
 - Az alkalmazó és az ügyfelek adatait bizalmasan kell kezelni, függetlenül attól, hogy erre formális titokvédelmi szerződés kötelez-e vagy sem.
- Felkészültség
 - Nem szabad saját szakértelmünket és tapasztalatunkat felülbecsülni. Tilos olyan munkát elvállalni, amihez felkészültségünk nem elegendő.

A szakmai felelősség kérdései

- Szellemi tulajdonok
 - Tisztában kell lennünk a törvény szellemi jogokra vonatkozó részleteivel (pl. szabadalom, copyright, stb.) Úgy kell eljárunk, hogy az a munkaadó és az ügyfelek szellemi tulajdonát védje.
- Technikai visszaélés
 - Nem szabad technikai ismereteinket számítógépes visszaélésre felhasználni. A visszaélések skálája széles: pl. munkaadó gépén játék, vagy akár vírusok terjesztése.

Az ACM/IEEE etikai kódexe

- A fenti szakmai szervezetek együttműködésével jött létre az etikai kódex.
- A szervezetek tagjai csatlakozásukkal elfogadják a kódexet.
- A Kódex nyolc alapelve a szakemberek (mérnökök, oktatók, vezetők, diákok, stb.) viselkedését és szakmai döntéseit szabályozza.

Etikai kódex - bevezető

- A szoftvermérnökök kötelessége, hogy szakmájukat (szoftveranalízis, -specifikáció, -tervezés, -fejlesztés és -karbantartás) hasznosan és elismerésre méltó módon végezzék.
- Az egészségügyi, biztonsági rendszabályokkal, valamint a közjó érdekével összhangban a következő nyolc alapelv betartását vállalják:

Etikai kódex - alapelvek

1. KÖZÉRDEK

- A szoftvermérnöknek mindenkor a köz érdekének megfelelően kell cselekedniük.

2. ÜGYFÉL ÉS ALKALMAZÓ

- A szoftvermérnöknek a megrendelő és az alkalmazó érdekében kell eljárnia, a közérdek figyelembevételével.

3. TERMÉK

- A szoftvermérnöknek biztosítania kell, hogy termékei a lehető legmagasabb szakmai színvonalat ériék el.

Etikai kódex - alapelvek

4. ÍTÉLŐKÉPESSÉG

- A szoftvermérnökök szakmai ítéleteit önállóan és függetlenül kell meghoznia.

5. MENEDZSMENT

- A menedzserek és egyéb vezetők kötelessége az etikus szoftverfejlesztés és -karbantartás biztosítása.

6. SZAKMA

- A szoftvermérnöknek a szakma jó hírét a köz érdekével összhangban öregbítene kell.

Etikai kódex - alapelvek

7. MUNKATÁRSAK

- A szoftvermérnöknek támogatnia kell munkatársait.

8. ÖNFEJLESZTÉS

- A szoftvermérnöknek folyamatosan fejlesztenie kell szakmai tudását. Mindenkor etikus szakmai viselkedést kell tanúsítania és azt másoktól is elvárnia.

Etikai dilemmák

- A vezetés elveivel való egyet nem értés.
- Az alkalmazó nem etikusan jár el. Pl.: egy biztonságkritikus rendszert megfelelő tesztelés nélkül bocsát ki.
- Katonai vagy nukleáris rendszerek fejlesztésében való részvétel.

Összefoglalás

- A szoftvermérnöknek a szoftver gyártásának összes aspektusával foglalkoznia kell.
- A szoftver termék része a program és a hozzá tartozó dokumentáció. Főbb ismérvei: karbantarthatóság, megbízhatóság, hatékonyság és jól használhatóág.
- A szoftvergyártás a szoftver létrehozásának lépéseiből áll. Főbb elemei: specifikáció, fejlesztés, validáció és evolúció.
- Különböző metodikák léteznek szoftverek fejlesztésének támogatására. Ezek tartalmazznak javaslatokat a fejlesztés menetére, az alkalmazott jelölésrendszerre, szabályokat a rendszer leírására, valamint tervezési ajánlásokat.

Összefoglalás

- A CASE rendszerek a szoftvergyártás rutintevékenységeinek támogatására szolgálnak. Pl.: diagram szerkesztés és tervezés, konzisztencia ellenőrzés, programtesztek nyilvántartása, stb.
- A szoftvermérnök felelős szakmájáért és a társadalomért is. Nem csak technikai feladatai vannak!
- Szakmai szervezetek etikai kódexben szabályozzák tagjaik viselkedési normáit.