KORMÁNYZATI INFORMATIKAI EGYEZTETŐ TÁRCAKÖZI BIZOTTSÁG

24. SZÁMÚ

AJÁNLÁSA

a központi közigazgatási szervek szoftverfejlesztéseihez kapcsolódó minőségbiztosításra és minőségirányításra vonatkozóan

2005. december

Készült:

a Miniszterelnöki Hivatal Elektronikuskormányzat-központ megbízásából

Az Ajánlást a KIETB a 2005. december 1-jei ülésén fogadta el.

Tartalomjegyzék

ELŐSZÓ **5**

1. BEVEZETÉS **7**

2. ÖSSZEFOGLALÓ **9**

3. MINŐSÉGGEL KAPCSOLATOS FOGALMAK **11**

4. A MINŐSÉGBIZTOSÍTÁS CÉLJA, HELYE ÉS FELADATAI A FEJLESZTÉSBEN **12**

5. ELŐZETES KOCKÁZATFELMÉRÉS SZEMPONTJAI ÉS HATÁSA A MINŐSÉGBIZTOSÍTÁSRA **14**

6. AZ INFORMATIKAI RENDSZEREK FEJLESZTÉSÉNEK MINŐSÉGBIZTOSÍTÁSI SAJÁTOSSÁGAI **15**

6.1. EGYMÁSNAK ELLENTMONDÓ CÉLOK ÉS PRIORITÁSOK 15
6.2. TÖKÉLETLEN SPECIFIKÁCIÓ KEZELÉSE 15
6.3. A SPECIFIKÁCIÓ HELYESSÉGÉNEK MEGÍTÉLÉSE 17
6.4. AZ IGÉNYEK ÉS A TECHNOLÓGIA ADOTT SZINTJE KÖZTI ELLENTMONDÁSOK 17
6.5. TOVÁBBFEJLESZTHETŐSÉG, MINT MINŐSÉGI CÉL 17
6.6. RENDELKEZÉSRE ÁLLÁS, MINT MINŐSÉGI CÉL 17
6.7. ÜZEMELTETHETŐSÉG, MINT MINŐSÉGI CÉL 18
6.8. TESZTELÉS MINŐSÉGE ÉS A TERMÉK MINŐSÉGE 18
6.9. TERVEK, DOKUMENTÁCIÓK MINŐSÉGE, TESZTELÉSE 19

7. A MINŐSÉGBIZTOSÍTÁS ELŐZETES KÖVETELMÉNYEI **20**

8. A FOLYAMATOS MINŐSÉGIRÁNYÍTÁS SZERVEZETE ÉS FELADATAI **21**

8.1. A MINŐSÉGIRÁNYÍTÁS SZERVEZETE 21
8.2. A MINŐSÉGIRÁNYÍTÁSI RENDSZER ÉS A FEJLESZTÉSI PROJEKT MINŐSÉGIRÁNYÍTÁSA 22

9. A FEJLESZTÉSI PROJEKT MINŐSÉGBIZTOSÍTÁSI ELJÁRÁSAI **24**

9.1. PROJEKTKEZDEMÉNYEZÉS 25
9.1.1. Célok *25*
9.1.2. Áttekintés, termékek *25*
9.1.3. Igényfelvétel és az előzetes döntés ellenőrzése *27*
9.1.4. A megvalósíthatósági tanulmány ellenőrzése *27*
9.1.5. A projektindítás előkészítésére és a projekt indításáról történő döntésre vonatkozó ellenőrzés *29*
9.2. PROJEKTDEFINÍCIÓS SZAKASZ 30
9.2.1. Célok *30*
9.2.2. Áttekintés *30*

9.2.3. Termékek ... *31*
9.2.4. A Projektdefiníció ellenőrzése ... *32*
9.3. PROJEKTTERVEZÉS ... 33
9.3.1. Célok ... *33*
9.3.2. Áttekintés és termékek ... *33*
9.3.3. A menedzsmentterv ellenőrzése ... *34*
9.3.4. A projekt terv ellenőrzése ... *35*
9.4. SZOFTVERFEJLESZTÉSI FOLYAMATOK ... 36
9.4.1. Áttekintés ... *36*
9.4.2. Specifikációs folyamatok ... *39*
9.4.3. Rendszertervezési folyamatok ... *45*
9.4.4. Implementációs folyamatok ... *53*
9.4.5. Tesztelési folyamatok ... *60*
9.4.6. Kibocsátási folyamatok ... *69*
9.5. PROJEKTZÁRÁS ÉS ÉRTÉKELÉS ... 71
9.5.1. Áttekintés és célok ... *71*
9.5.2. Termékek ... *73*
9.5.3. A projektzárás és ellenőrzés minőségellenőrzése ... *73*
9.6. PROJEKTMENEDZSMENT ... 74
9.7. KOCKÁZATMENEDZSMENT ... 74
9.7.1. Áttekintés ... *74*
9.7.2. Célok ... *77*
9.7.3. Termékek ... *77*
9.7.4. A kockázatkezelési stratégia ellenőrzése ... *77*
9.7.5. A kockázatkezelés ellenőrzése ... *78*
9.8. KONFIGURÁCIÓMENEDZSMENT ... 79
9.8.1. Áttekintés ... *79*
9.8.2. Célok ... *81*
9.8.3. Termékek ... *81*
9.8.4. A konfigurációmenedzsment ellenőrzése ... *81*
9.9. VÁLTOZÁSMENEDZSMENT ... 84
9.9.1. Áttekintés ... *84*
9.9.2. Célok ... *84*
9.9.3. A változásmenedzsment folyamatának ellenőrzése ... *85*

10. A MINŐSÉGBIZTOSÍTÁS FOLYAMATÁNAK DOKUMENTUMAI ... 90

10.1. MINŐSÉGBIZTOSÍTÁSI TERV ... 90

11. VONATKOZÓ SZABVÁNYOK ... 93

12. KAPCSOLÓDÓ INFORMÁCIÓK ... 94

Előszó

A fejlesztések hatékonyságának emeléséhez, a nem kielégítően elkészített szoftvertermékek miatt később szükségessé váló pótlólagos költségek megszüntetése vagy minimális mértékűre csökkentése, illetve a rendszerek alkalmazása során jelentkező kockázatok felmérése és figyelembe vétele érdekében a fejlesztések végrehajtásával egyidejűleg a minőségirányítás rendszerének működtetése indokolt. A specifikumok miatt ugyanakkor nem célszerű és nem indokolt a különböző területek mindegyikét átfogó, egységes minőségbiztosítási követelményrendszer, ajánlás kiadása. Más minőségbiztosítási eljárások szükségesek egy késztermék vásárlása, egy üzemeltetési szerződés előkészítése, és más a szoftverek, alkalmazások fejlesztése során.

A 2316/2003. (XII. 10.) Korm. határozat 3. pontja a Miniszterelnöki Hivatalt vezető miniszter feladatköreként előírja, hogy ajánlás kibocsátása szükséges a minőségbiztosítás rendszeréről, amelyet a tapasztalatok alapján rendszeresen aktualizálni kell. A kormányhatározat ezen pontjának végrehajtása érdekében a Kormányzati Informatikai Egyeztető Tárcaközi Bizottság (KIETB) elkészíttette az informatikai rendszerek fejlesztésével, ezen belül a szoftverfejlesztéssel kapcsolatos ajánlást.

Az ajánlás címe a szoftverfejlesztéshez kapcsolódó minőségbiztosításról szól, de tartalmából is nyilvánvalóvá válik, hogy nem az egyszerű, kisértékű egyedi szoftverek, hanem a szoftverrendszerek, alkalmazások fejlesztésének minőségirányítással összefüggő feladatait foglalja össze és rendszerezi.

A központi közigazgatás intézményei – nagyon kevés kivételtől eltekintve – önállóan nem fejlesztenek szoftvereket, de a megrendelői pozíció érvényesítéséhez, az átvételre kerülő szoftvertermék minőségének megállapításához nem elegendő a késztermék tesztelése. A minőség garantált módon úgy biztosítható, ha a megrendelő a projekt kezdetétől fogva és folyamatosan figyelemmel kíséri a fejlesztés végrehajtását, és a szükséges esetekben beavatkozik.

Az ajánlás tartalmazza mindazon az eljárások, technikák, dokumentumok, illetve végrehajtandó feladatok ismertetését, amelyek a fejlesztés során a megfelelő minőségű végtermék létrehozását biztosítják.

A szoftverek (alkalmazások) fejlesztéséhez kapcsolódó minőségbiztosítás terén a központi közigazgatás intézményeinek nincsenek speciálisan eltérő feladatai, ezért az ajánlás máshol is alkalmazható, azonban a KIETB hatásköre a központi közigazgatás szervezeteire terjed ki, ezért szükségesnek tartottuk az érintett intézményi kör behatárolását.

Az ajánlás az informatikai (szoftver) rendszerek minőségbiztosításával összefüggő feladatokra koncentrál, ezzel összefüggésben a fejlesztés minden fázisát, és a rendszer kialakításához kapcsolódó kérdéseket érinti. Ebben az ajánlásban nem kívántunk foglalkozni a nem digitális adatok kezelésével foglalkozó információs rendszerekre vonatkozó minőségbiztosítási szempontok vizsgálatával, mert azok specifikussága meghaladta volna az ajánlás tervezett kereteit.

A KIETB az informatikai rendszerek üzemeltetésével (outsourcingbe adásával) kapcsolatos minőségbiztosítási ajánlását külön jelenteti meg.

1. Bevezetés

Jelen dokumentum célja, hogy szoftverfejlesztési projektek minőségbiztosítási tevékenységére vonatkozóan nyújtson olyan ajánlást, mely alapján a közigazgatási intézmények az egyes projektek minőségbiztosítását el tudják végezni, valamint segít kialakítani szoftverfejlesztési projektjeikre vonatkozó minőségirányítási rendszerüket.

Az ajánlás szükségességét az indokolja, hogy általában és a közszférában is a szoftverfejlesztési projektek rendkívül nagy hányada végződik sikertelenül, vagy részleges sikerrel. A legnagyobb problémák általában nem technológiai korlátokból fakadnak, hanem projektvezetési és minőségi hiányosságokból, az alkalmazások megbízhatatlanságából. Így a projekt sikerét nagyban befolyásolja az, hogy a fejlesztési folyamat során biztosított-e a minőségi kritériumok kialakítása és teljesítése. A szoftverfejlesztésben a fejlesztési technológiák javulása miatt egyre inkább csökken a funkcionális követelmények kielégítésének nehézsége, és a megbízható és pontos működés kerül előtérbe. Ezek kiemelik a minőségbiztosítási tevékenység fontosságát.

A fentiek okán a Miniszterelnöki Hivatal Elektronikuskormányzat-központ megbízást adott egy ajánlás elkészítésére.

Kiknek és milyen projektekre szól ez az ajánlás?

Az ajánlás azoknak a vezetőknek vagy minőségbiztosítással megbízott szakértőknek szól, akik közepes méretű fejlesztési projektekben vesznek részt, így érdekeltek a projekt sikerességében és a termékek hosszú távú felhasználhatóságában és gazdaságosságában. A minőségbiztosítási tevékenységek a fejlesztési folyamat minőségének kontrollálásában nyújtanak segítséget. Ezen kívül az ajánlás a szervezetek szoftverfejlesztésre vonatkozó minőségirányítási rendszer kialakításával megbízott vezetőknek is jó kiindulási alapot nyújt.

Hogyan érdemes használni ezt az ajánlást?

Ez az ajánlás a fejlesztési projektek minőségbiztosítási tevékenységeinek olyan halmazát vázolja fel, melyet minden projekt a követelményeket és a projekt jellegzetességeit figyelembe véve használhat *gyakorlati útmutatóként*. A könnyű felhasználást és az átláthatóságot segítendő a dokumentum *a projekt folyamatát követve épül fel* az összefoglalóban bemutatottak szerint. Az egyes projektek minőségbiztosítási tervének kialakításakor a projekthez és környezetéhez igazodva kell a megfelelő hangsúlyokat megtalálni.

Amennyiben a szervezetben kevés a projekt-minőségbiztosítási tapasztalat, érdemes egyszerűbb és letisztultabb rendszert kialakítani: kevesebb minőségbiztosítási tevékenységet elvégezni, azonban azokat alaposan és a kellő rendszerességgel, kontrollal és szakértelemmel.

Az ajánlás elkészítésekor az alábbi előfeltételezésekkel éltünk a projekttel és a projektet megvalósító szervezettel szemben:

- a fejlesztési feladatot projekt-szerűen valósítják meg, tehát projektek általános keretei érvényesek a fejlesztési folyamatra;
- a fejlesztés során van allokált erőforrás a minőségbiztosítási tevékenység elvégzésére;
- a minőségbiztosítási tevékenység tartalmi javaslatokat is ad a fejlesztéshez, azaz nem csupán pl. dokumentációs rendet felügyel, hanem a minőségbiztosító szakmai munkát is végez, javaslatokat is adhat;
- jelen ajánlás egyedi fejlesztések minőségbiztosítására készült elsősorban. Ezek természetesen tartalmazhatnak kész szoftver elemeket, de a célzott projektek alapvetően egyedi fejlesztést valósítanak meg. Nem rendszer bevezetési, testreszabási projektek támogatására született az ajánlás, de részelemei ott is felhasználhatóak.

2. Összefoglaló

Az ajánlás a minőségirányítás célja, a projektben, valamint a szervezetben betöltött szerepe, a fejlesztési projektekben megmutatkozó sajátosságok, az előzetes követelmények tisztázása után a szoftverfejlesztési projektek minőségbiztosítási tevékenységeit mutatja be.

A fejlesztési projektek minőségbiztosítási eljárásai a *projekt teljes folyamatában* megjelennek, tehát a minőségbiztosítási feladatokat a fejlesztési folyamatoktól elkülönülten, de velük egy időben kell végezni. Ennek megfelelően a dokumentum felépítése a fejlesztési projektek folyamatához igazodik:

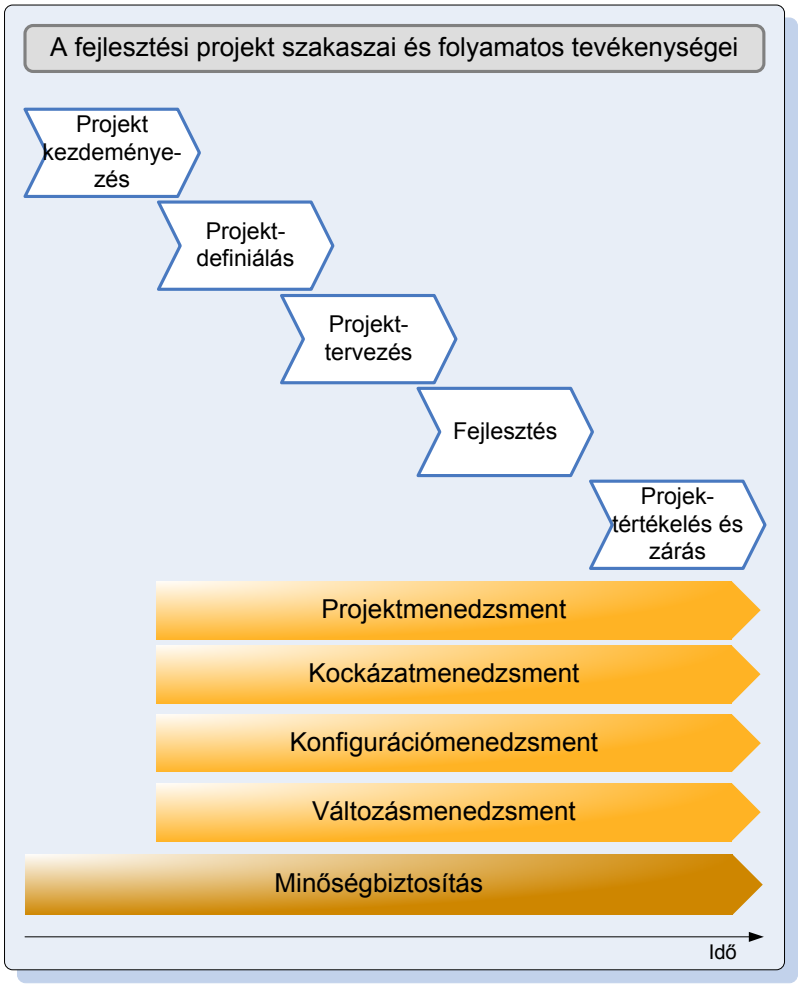
- bemutatja a projektek során *rendszeresen*, szakaszoktól függetlenül végzendő minőségbiztosítási tevékenységeket,
- végigtekinti a *projekt szakaszokat és a hozzájuk kapcsolódó* minőségbiztosítási tevékenységeket.

A rendszeresen végzendő minőségbiztosítási tevékenységek jelentik

- egyrészt, hogy a *minőségbiztosító a projekt szervezet része* és a projekt *munkában rendszeresen részt vesz*; tagja az egyeztető megbeszéléseknek, részt vesz a szakaszok előkészítésében és tervezésében, szúrópróba-szerűen és rendszeresen vizsgálja a köztes termékeket, és szakmai javaslatokat tesz;
- másrészt, hogy a *szakaszokon átívelő tevékenységekben* (projektmenedzsment, konfigurációmenedzsment, kockázatmenedzsment, változásmenedzsment) is ellát *ellenőrzési* feladatokat, és ezekben is szakmai javaslatokat tesz.

Az egyes projekt szakaszok minőségbiztosításának bemutatása során *végigtekintjük röviden a szakaszokban végzett tevékenységeket* és az elkészítendő termékeket, majd kifejtjük a hozzájuk kapcsolódó minőségellenőrzések célját, az ellenőrzést végző(k) javasolt tudáskészletét, az ellenőrzéshez szükséges bemeneteket (pl. dokumentumok), valamint az *ellenőrzés feladatait, szempontjait*.

A projekt szakaszokat és az azokon átívelő folyamatos tevékenységeket az alábbiak szerint vettük sorba a tanulmányban:



Ábra 2-1. A szoftver fejlesztési projekt szakaszai és folyamatos tevékenységei

A minőségbiztosítási feladatok elvégzése – a fentiekből következően – általában tapasztalt, szenior szakértőt kíván.

Az alkalmazott minőségbiztosítási módszereknek és az ellenőrzéseknek arányosnak kell lenni a fejlesztett, illetve beszerzett informatikai rendszer értékével, illetve kritikusságával, valamint a felmerülő kockázatokkal.

3. Minőséggel kapcsolatos fogalmak

Mivel ebben az ajánlásban olyan tevékenységeket tárgyalunk, melyek a projektekben végzett tevékenységek, valamint a termékek minőségének javítását célozzák, mindenképpen tisztázni szükséges a minőséghez kapcsolódó legfontosabb fogalmakat.

- Minőségirányítási rendszer:
  a minőségirányítás megvalósításának az eszköze. Ez tartalmazza “a minőség megvalósításában alkalmazott szervezeti struktúrát, felelősségeket, eljárásokat, folyamatokat és erőforrásokat” (ISO 8402-1986). Ezen keresztül képes a vezetés a minőségügyi politikát és a minőségügyi célokat megvalósítani a szervezet fő-, irányítási és támogató folyamatainak szabályozásával.
- Minőségtervezés:
  a minőségirányításnak azon része, mely a minőségcélok, valamint a szükséges működési folyamatok és a velük kapcsolatos erőforrások meghatározására összpontosít a minőségpolitika megvalósulása érdekében.
- Minőségbiztosítás:
  a minőségbiztosítási tevékenység alatt azokat a rendszeres és strukturált minőségellenőrzéseket értjük, melyeket a minőségirányítási rendszer eljárásai és folyamatai meghatároznak.
- Minőségellenőrzés:
  a minőségellenőrzések során a minőségi kritériumoknak való megfelelést ellenőrzése történik meg. Az ellenőrzések kialakítása figyelembe veheti az iparági legjobb gyakorlatokat és az adott szervezet irányelveit, előírásait is.
- Minőségpolitika:
  egy szervezetnek a minőségre vonatkozó, a felső vezetés által megfogalmazott és kinyilvánított általános szándékai és irányvonala.
- Projekt:
  a projekt olyan feladat végrehajtása, amely
  - egy egyszeri, jól körülhatárolt cél érdekében zajlik,
  - több szervezeti funkcionális területet érint (komplexitás), rögzített erőforráskeretekkel, és
  - kötött határidővel rendelkezik.

  Minden egyes feladatjellemző rendkívül fontos, azaz kizárólag akkor beszélhetünk projektről, ha a fenti feltételek mindegyike teljesül.

Megjegyzés: a projekt egyes definíciói az erőforráskorlátokat nem tartalmazzák, csak az egyedi cél és az időbeli behatároltság kerül megemlítésre.

4. A minőségbiztosítás célja, helye és feladatai a fejlesztésben

A szoftverfejlesztési projektekben a minőségbiztosítás célja, hogy a fejlesztési tevékenység folyamatainak, így a termékeinek minőségét is javítsa. A javulás az egyes tevékenységekre vonatkozó kritériumok kidolgozásán, azok teljesülésének ellenőrzésén, valamint a folyamatos szakértői részvételen keresztül valósulhat meg. Az ellenőrzések a projekt során a minőségirányítási rendszer által rendelkezésre bocsátott szervezet és erőforrások, valamint a kidolgozott eljárások és folyamatok segítségével kontrolláltan történnek.

A folyamatos minőségbiztosítás célja tehát, hogy javítsa a minőséget, minél korábban jelezze, amennyiben a minőségi céloktól eltérést tapasztal, valamint meghatározza azokat a lépéseket, melyekkel a hibák kiküszöbölhetőek.

A fejlesztési folyamat minőségének javulásától általánosan elvárt konkrét eredmények az alábbiak:

- kevesebb szoftverhiba (bug),
- gördülékeny fejlesztés, kevesebb erőforrás-szükséglet a hibajavításokra,
- stabilabb implementáció, működés, és továbbfejleszthetőség,
- kevesebb felhasználói támogatást igénylő bejelentés a működés során.

A minőségbiztosítás helyének és feladatainak meghatározásánál figyelembe kell venni az alábbi megállapításokat.

- Fontos, hogy a minőségbiztosítást a projekt legelejétől végezzék, az a teljes fejlesztési folyamatra kiterjedjen (a kezdeményezéstől a zárásig), mivel minél később történnek az ellenőrzések, annál több erőforrást igényel a felfedezett hibák javítása, és annál nehezebb észrevenni azokat, így a termék minősége várhatóan rosszabb lesz. A minőségbiztosítás megelőzi az apró elmaradások, problémák felhalmozódását, amelyek általában csak a tesztelés kezdetén kerülnek felszínre.
- A minőségbiztosítási tevékenység módszereit, feladatait a fejlesztés méretének, jellegének és súlyának, valamint az erre a feladatra rendelkezésre álló erőforrások mennyiségének figyelembevételével kell kialakítani minden projektben. A minőségbiztosítási feladatoknak arányosnak kell lenniük a fejlesztett/beszerzett informatikai rendszer értékével, illetve kritikusságával és a felmerülő kockázatokkal (ld. később).
- A felhasználói területek bevonása a minőségellenőrzés folyamatába és a tesztelésbe nagymértékben javítja a kiadott szoftverek minőségét és a felhasználói megelégedést.
- A minőségbiztosítási csapat legyen a projekt szervezet része, illeszkedjen be a projektmunkába, ezzel biztosítva a minőségügyi követelmények teljesülését.

- Fejlesztő csapat oktatása révén kell tudatosítani a minőségbiztosítás hasznosságát, hogyan járul hozzá a munkájuk sikeréhez. Ezzel érhető el a minőségbiztosítók megfelelő gyakorlati bevonása a projektekbe.

Általános szabályként elmondható, hogy a projektek számára rendelkezésre álló erőforrások 5-10 százalékát érdemes a minőségbiztosításra fordítani, de ez a körülményektől függően jelentősen módosulhat.

Fontos tisztázni, hogy a *minőségbiztosítási tevékenység* a fejlesztési folyamatban *két módon* helyezkedik el:

- egyrészt az *egyes (rész)szakaszok végén* a szakaszok által adott *termékek minőségbiztosítása* történik meg, a később kifejtett módon,
- másrészt a *tevékenységek folyamatos minőségbiztosítása* szükséges mind az egyes szakaszokhoz kapcsolódó tevékenységek esetében, mind pedig a szakaszokon átívelő tevékenységeknél (projektmenedzsment, kockázatmenedzsment, változáskezelés).

A fentiekből következően a minőségbiztosító a projektszervezet része és a projektmunkában rendszeresen részt vesz. Tagja az egyeztető megbeszéléseknek, részt vesz a szakaszok előkészítésében és tervezésében, szúrópróbaszerűen vizsgálja a köztes termékeket, és szakmai javaslatokat tesz. Munkamódszere magában foglalhatja a projektdokumentumok vizsgálatát, vagy éppen a személyes interjúkat, informális megbeszéléseket. A feladat általában tapasztalt, szenior szakértőt kíván. A szakértő lehet a szervezet munkatársa, figyelembe véve az összeférhetetlenségeket, vagy külső szakértő, tanácsadó, amennyiben az független a szállítótól.

5. Előzetes kockázatfelmérés szempontjai és hatása a minőségbiztosításra

A minőségbiztosítási tevékenység módszereinek kialakításában meghatározó az, hogy a projekt a különböző kockázati szempontok szerint milyen jellemzőkkel rendelkezik. A projekt kockázatainak felmérésekor az alábbi szempontokat érdemes figyelembe venni:

- a projekt összértéke, az allokált erőforrások mennyisége,
- a projekt hossza,
- a projekt bonyolultsága,
- a használt technológiák műszaki újdonsága,
- a résztvevők száma és tapasztalata,
- a rendszer kritikussága,
- a szállítók jellemzői és kapcsolatuk a szervezettel,
- a felhasználás tárgya és a felhasználók személye, száma.

A fenti tényezőket a projekt kezdése előtt azonosítani kell (egyéb tényezők figyelembe vétele természetesen szükséges lehet) és ki kell őket értékelni.

Ezek fényében kell meghatározni a minőségbiztosításra szánt erőforrásokat és annak módszereit, elvárt alaposságát – a körülményeknek megfelelően súlyozva a tényezőket. Minél nagyobb kockázatot hordoz a projekt, annál több erőforrást érdemes allokálni a minőségbiztosítási tevékenységre – ami az alkalmazható módszerekre is hatással van.

6. Az informatikai rendszerek fejlesztésének minőségbiztosítási sajátosságai

Az informatikai rendszerek fejlesztése számos olyan sajátossággal rendelkezik, amelyeket figyelembe kell venni a minőségbiztosítás kapcsán is.

6.1. Egymásnak ellentmondó célok és prioritások

A fejlesztési projektek egyes céljai ellentmondásba kerülhetnek egymással. A célok ellentmondása veszélyezteti a projekt sikerét, azaz kockázatot jelent, amelyet kezelni kell. A minőségbiztosításnak vizsgálnia kell azt is, hogy az ellentmondó célok közt prioritások kerüljenek felállításra, és a kompromisszumos megoldás a prioritások figyelembevételével kerüljön kialakításra.

- A projekt határideje, kiterjedése és az erőforrások kényes egyensúlyban vannak. Döntésünktől függően ezek közül az egyiket fixnek vehetjük, a másikat paraméternek tekinthetjük, a harmadik ezek eredményeként adódik. Ezt figyelembe kell venni a célok kialakításánál, a célokat súlyozni kell egymáshoz képest.

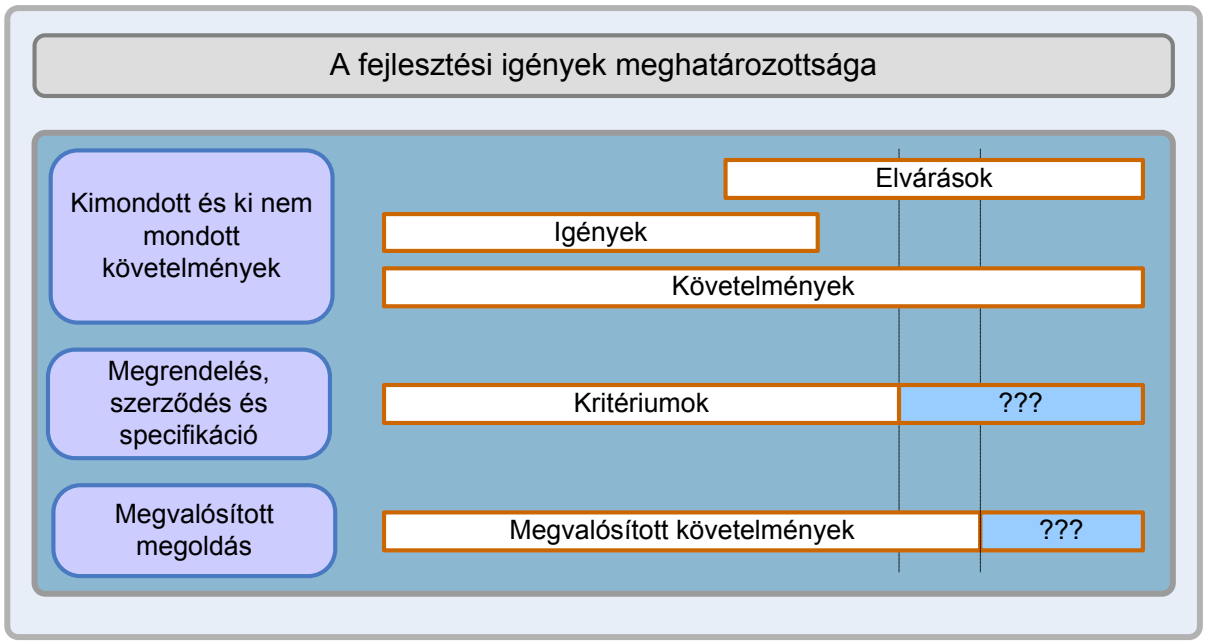
Megjegyzés: Ezt az ökölszabályt még annyiban érdemes árnyalni, hogy az időbeli ráfordításnak vannak sajátos jellemzői is: bizonyos tevékenységek időszükséglete több erőforrással sem csökkenthető egy adott szint alá. Ennek klasszikus szemléltető példája, hogy ha 1 nő 9 hónap alatt szül meg egy gyereket, attól még 9 nő 1 hónap alatt nem végzi el ugyanezt. Sok szellemi tevékenységnek is vannak ilyesfajta korlátai: ha egy szakértő egy rendszer-architektúra kialakítását 1 hónap alatt végzi el, akkor 2 szakértővel nem feltétlenül lesz kész fele idő alatt; sőt a kommunikációs igények és a koncepcionális disszonanciák miatt akár több időt is igényelhet a több résztvevő.

Megjegyzés: A határidő – feladat – erőforrások hármasa mellett értelmezhető a határidő – minőség – erőforrás hármasa is, vagyis nem csak az igaz, hogy nagyobb méretű feladat megoldásához több idő és/vagy több erőforrás kell, hanem az is, hogy azonos méretű feladat igazoltan (tehát tesztekkel alátámasztottan) jobb minőségben történő megoldása is több időt és/vagy erőforrást igényel.

- A fejlesztés kritériumai, (mint biztonság és rugalmasság, sebesség vagy erőforrásigény) a technológia adott szintje és adott költségkeret mellett szintén egymásnak ellentmondhatnak, ezért ezek közt is egyensúlyt kell teremteni.

6.2. Tökéletlen specifikáció kezelése

A minőségbiztosítás mindig a célok teljesüléséhez képest tud állást foglalni a minőségről. A fejlesztési célok egy része állandó (alapcélok, projektcélok), más része a projekt során kerül megfogalmazásra vagy pontosításra.



Ábra 6-1. A fejlesztési igények meghatározottsága projekt során

A fejlesztési projekt (akár külső, akár belső fejlesztésről beszélünk) feladata egy adott *kritérium*rendszernek megfelelő megoldás elkészítése. A kritériumrendszer tartalmazza a megrendelő szervezet kimondott *igény*eit és a ki nem mondott *elvárás*ok (pl.: törvényi megfelelés, szabványoknak való megfelelés, belső előírásoknak való megfelelés, megszokásból adódó feltételezések) egy részét, de nem tér ki az összes követelményre. Ha a fejlesztést külső szállító végzi, akkor a kritériumrendszer általában az ajánlatkérés során nagyrészt rögzített, esetleg a projektnek része a specifikáció pontosítása.

A projekt előrehaladásával a ki nem mondott elvárások újabb része explicit módon is megfogalmazásra kerül, illetve az alacsonyabb szintű részcélokat a tervezés és a megvalósítás is módosítja.

Emiatt a céloknak, a specifikációnak való megfelelőséget az aktuális célokhoz, specifikációhoz viszonyítva kell megítélni.

Ezeknek a szerződéses megállapodáshoz való viszonya, és azzal való összhangja külön vizsgálandó.

Megjegyzés: azon módszertanokban, melyek elfogadják, hogy a már leírt specifikációkon változtassanak a valós igények kielégítése érdekében, létezik az ún. „élő dokumentum” fogalma. Ennek jelentése, hogy bizonyos dokumentumokat folyamatos iterációval az igényekhez igazítanak a projekt folyamán. Ez semmi esetre sem azt jelenti, hogy bármikor „büntetlenül” módosítani lehetne a követelményeket, vagy a terveket, hanem arra vonatkozik, hogy a dokumentum első változatát viszonyítási pontnak („baseline”) tekintik. Ezt csak a változáskezelési folyamatok betartásával, és a hatások gondos mérlegelése mellett lehet módosítani.

6.3. A specifikáció helyességének megítélése

A fejlesztés tulajdonképpen a követelmények egyre formalizáltabb megfogalmazását jelenti a specifikáción, terveken (logikai, fizikai) és magán az alkalmazás kódján keresztül is. (A kód is egy specifikáció, mely nagyon formalizált és részletes.)

A specifikáció a tervezés előrehaladtával egyre formálisabb leírást jelent, amit a megrendelő / felhasználó egyre kevésbé ért. Ez nehezíti a specifikáció helyességének ellenőrzését, és szakértő bevonását igényli.

A tervezési dokumentációk ellenőrzése annál egyszerűbb, minél inkább a megrendelő által is ismert formalizmust (jelölésrendszert, módszertant) használ.

A specifikáció helyességének ellenőrzését segítheti a prototípus alapú fejlesztés és a kísérleti bevezetés (pilot projekt).

6.4. Az igények és a technológia adott szintje közti ellentmondások

Az igények maradéktalan teljesítése a fejlesztések esetében technológiai korlátokba ütközhet. (Teljesítmény elvárások, használhatósági elvárások.) Annak eldöntése, hogy az adott elvárások a technológia adott szintje mellett teljesíthetőek-e, speciális vizsgálatokat, teszteket igényelhetnek, amelyek elkészítéséhez szakértő bevonására van szükség.

6.5. Továbbfejleszthetőség, mint minőségi cél

A fejlesztések gyakran egy távlati átfogóbb elképzelés egy lépcsőjeként kerülnek megvalósításra. Ilyenkor előre ismert, hogy a rendszer funkcionalitása a későbbiekben bővülni és/vagy változni fog. Ekkor cél, hogy a fejlesztések úgy készüljenek, hogy a későbbi továbbfejleszthetőséget is támogassák (akár a pillanatnyi többletköltségek árán is).

Ennek a minőségi célnak a megítélése a rendszer architektúrájának, a kód és a dokumentációk minőségének ellenőrzését igényli, amihez szakértők bevonása szükséges.

Ezen kívül szükséges definiálni – ha megközelítőleg is – azt a távlati referenciapontot, melyhez az adott projekt, mint egy lépcsőfok, közelebb visz. Ezért fontos, hogy a megrendelőnek legyen távlati képe (víziója), illetve nagyvonalú fejlesztésütemezési elképzelése (roadmap) az adott témában.

6.6. Rendelkezésre állás, mint minőségi cél

Az alkalmazástól elvárt rendelkezésre állás olyan tervezési szempont, melyhez figyelembe kell venni az alkalmazás karbantartási igényét, és az alkalmazás architektúrát kiszolgáló infrastruktúra (hardver, szoftver és szolgáltatások) sajátosságait és ezek összefüggéseit egyaránt.

A rendelkezésre állási céloknál fontos tudatosítani, hogy a nagy rendelkezésre állás legfontosabb megvalósítási módja a redundancia beépítése. Ez viszont a költségek jelentős emelkedésével jár.

A rendszert rendelkezésre állási szempontból egy összefüggő láncolatnak kell tekinteni, ahol a leggyengébb láncszem elve érvényesül: amennyiben a legkisebb rendelkezésre állással rendelkező elem kiesik, az egész rendszer működésképtelenné válik.

Ezért az egyes elemekkel szemben tehát úgy érdemes kialakítani a rendelkezésre állási követelményeket, hogy összességében teljesítsék a rendszer egészével szemben fennálló elvárásokat.

Ebből következően nem érdemes egy alkalmazással szemben olyan magas rendelkezésre állási kritériumokat megfogalmazni – és nagy ráfordításokkal megvalósítani –, melyeket például az alatta levő infrastruktúra, vagy a kapcsolódó rendszerek nem teljesítenek.

A rendelkezésre állás növelésének másik lehetséges módja ugyanazon az infrastruktúrán a rendszer megbízhatóságának kezelése. Ez a stabilabb kód készítésével, kiteszteltség növelésével, öndiagnosztikai és/vagy önjavító mechanizmusok alkalmazásával lehetséges.

6.7. Üzemeltethetőség, mint minőségi cél

Az alkalmazások használatának (tulajdonlásának) összes költségei közül az üzemeltetéshez kapcsolódó költségek tipikusan meghaladják a beszerzéshez kapcsolódó költségeket, ezért az üzemeltethetőség fontos tervezési szempont.

Az üzemeltethetőségre vonatkozóan mérhető kritériumokat kell meghatározni, és a minőségbiztosítás ezek teljesülését ellenőrzi.

6.8. Tesztelés minősége és a termék minősége

A szoftverfejlesztés kapcsán teljes hibamentességről és így teljes kiteszteltségről nem tudunk beszélni. (Pl. a http://www.kaner.com/coverage.htm oldalon több mint 100 tesztelési szempont található, nem teljes lista, amely alapján a teljes kiteszteltség értelmezhető lenne. Ezen tesztek maradéktalan elvégzésének költsége és végrehajtási ideje miatt sem lehetséges, még kis rendszerek esetében sem.)

Másik probléma, hogy a tesztelés teljessége és a maradék hibák mennyisége közt nincs ok-okozati, csak tapasztalati / statisztikai összefüggés. Ez azt jelenti, hogy a tesztelés minősége nem vonja maga után tervezhető módon a termék minőségét. Hogy milyen erős a kettő közti kapcsolat, azt csak utólag tudjuk megállapítani. Előzetesen csak iparági statisztikák / benchmark-ok segítségével becsülhető.

Az iparági benchmark-ok típushibák és teszttípusok, illetve ezek fedettsége közti összefüggésekre adnak jóslást. Éppen ezért a tesztelés minősége sok olyan paraméter megválasztásától függ, amely szakértői mérlegelést igényel:

- Tesztcélok helyes megválasztása

- Tesztelésre szánt idő
- Alkalmazott tesztek típusai
- A tesztelés teljessége (fedettség) teszttípusonként

6.9. Tervek, dokumentációk minősége, tesztelése

Érdekes módon a bejelentett szoftverhibák jelentős része nem a kódból, hanem másból származik. (specifikáció, tervek, hibajavításokkal bekerült új hibák, felhasználói dokumentációk hibái).

A dokumentációk ellenőrzésére, tesztelésére még kevésbé léteznek módszerek, és nagy szerepe van a szakértőnek. A tervek, specifikációk tesztelhetősége nagymértékben függ attól, hogy az alkalmazott módszertan maga mennyire strukturált, támogatja-e az ellenőrizhetőséget.

7. A minőségbiztosítás előzetes követelményei

Ahhoz, hogy egy projektben a minőségbiztosítás megvalósulhasson, és valóban elérje a kívánt eredményeket, bizonyos előfeltételeknek teljesülniük kell. Ezek az alábbiak:

- A projekt és a projektet megvalósító szervezet erős elkötelezettséggel rendelkezik a minőség iránt. Az ajánlás alkalmazása és a minőségbiztosítási dokumentumok elkészítése nem helyettesíti a vezetőség és a projekt munkatársak minőség iránti elkötelezettségét.
- A minőségbiztosítással foglalkozók rendelkeznek a megfelelő *hatókörrel* ahhoz, hogy a szükséges információhoz hozzáférjenek, és az erőforrásokat mozgósítani tudják.
- A minőségbiztosítással foglalkozók esetében *nem áll fenn összeférhetetlenség* a projekt szereplőivel, vagy magával a projekt céljával kapcsolatban. Így a munkatársak *nem állnak függelmi viszonyban* a szállítóval, a projekt menedzserrel, a projektet lebonyolító szervezeti egységgel. Javasolt, hogy a minőségbiztosító ne abból a szervezeti egységből kerüljön ki, aki a fejlesztést megrendelte, ez biztosítja a független külső megfigyelő és javaslattevő szerepet.
- A minőségbiztosítók olyan *szakmai tapasztalattal rendelkeznek*, melyek lehetővé teszik, hogy a formai ellenőrzéseken, a tevékenységek teljesülésének ellenőrzésén túl szakmai, módszertani javaslatokat is tudjanak tenni.

8. A folyamatos minőségirányítás szervezete és feladatai

A fejezet témája a minőségirányítási eljárások működéséhez szükséges szervezeti háttér és feladatok körvonalazása, valamint a fejlesztési folyamat minőségirányításának beillesztése a szervezet általános minőségirányítási rendszerébe.

8.1. A minőségirányítás szervezete

A minőségirányítás szervezetének felépítésére nincs általános legjobb gyakorlat, mivel azt minden intézménynek a saját jellegzetességei szerint kell kialakítania. A *minőségirányítási szervezet felépítését befolyásolja* többek között

- az intézmény tevékenységi köre,
- általános minőségirányítási rendszer működése (van-e auditált rendszer)
- a fejlesztések mérete és gyakorisága,
- a fejlesztések legtöbbször külső-e vagy belső fejlesztések,
- a szervezetben hosszútávon meglévő és szükséges szoftverfejlesztési szaktudás,
- a rendelkezésre álló létszámkeret.

A minőségirányítási szervezet így lehet önálló szervezeti egység, mely a különböző projektekhez rendel erőforrásokat, de lehet olyan kisebb egység is, mely feladatait szükség szerint látja el az erőforrásokkal rugalmasan gazdálkodva mátrix szervezetben és/vagy külsős munkatársakat is alkalmazva.

A *minőségirányítási szervezet*, illetve a minőségirányítással foglalkozó *vezető feladatait* az alábbiakban lehet összefoglalni:

- Olyan csapat összeállítása, mely tagjai megfelelő képességekkel rendelkeznek a fejlesztési projekt folyamatának és termékeinek minőségbiztosítására, és ismerik a szervezet fejlesztési szabványait.
- Belső minőségbiztosítási szabványok kidolgozása.
- Annak ellenőrzése, hogy a fejlesztés szabályozásához szükséges dokumentumok (szabályzatok, irányelvek, stratégiák) előírás szerint elkészülnek-e és rendelkezésre állnak-e.
- Általános szakmai támogatás nyújtása az intézményben minőségbiztosítási témákban.
- Részvétel a minőségellenőrzési eljárások, folyamatok és tesztek kidolgozásában, esetenként azok végrehajtásában.
- Minőség mérését biztosító metrikák meghatározása, az eredmények gyűjtése és kezelése.

- Részvétel a különböző irányító (steering comittee), audit bizottságokban.
- A menedzsment minőségtörekvéseinek képviselete, a szükséges lépések kidolgozása és megvalósítása.
- Általános feladat: a fejlesztők, felhasználók oktatása a minőségirányítási rendszerről és annak szükségszerűségéről.

A projektben dolgozó minőségbiztosító a szponzornak riportol, tehát a projekt szervezeti ábrát tekintve egy szinten helyezkedi el projektmenedzserrel. Ez lényeges elem, mely a minőségbiztosító függetlenségét és objektivitását segíti.

8.2. A minőségirányítási rendszer és a fejlesztési projekt minőségirányítása

Az intézményben működő, általános működésre vonatkozó minőségirányítási rendszernek összhangban kell lennie a szoftverfejlesztésre kidolgozott minőségirányítással, a projektben működő folyamatokkal.

Az általános működést szabályozó minőségirányítási rendszerrel szembeni követelmények:

- a rendszer tegye lehetővé, hogy a fejlesztési projektek esetében arra optimális minőségirányítást alkalmazzanak, legyen meg a kellő rugalmasság ehhez,
- a rendszer ne tegyen kötelezővé olyan dokumentációs rendet a fejlesztési projektre, mely csak az intézmény más jellegű tevékenységeinél hasznos elsősorban, a projektben nem feltétlenül szükséges és hátráltatja a munkát,
- legyen meghatározva, milyen bemenetekkel szolgálnak az egyes fejlesztési projektek az intézmény általános működésére vonatkozó minőségi elvárások követéséhez (pl. input a minőségcél teljesülésének méréséhez),
- az emberi erőforrásokkal való gazdálkodás módja és előírásai összhangban legyenek a fejlesztési projektekhez szükséges, rugalmas, esetleg szervezeti egységeken átívelő erőforrás gazdálkodással és erőforrás allokációkkal. Javasolt, hogy a projektszerű működéshez szükséges motivációs rendszer beilleszthető legyen a kialakított folyamatokba (pl. munkatársak értékelése).
- A tervezési, beszerzési folyamatok legyenek összhangban a szoftverfejlesztés sajátosságaival, az általános előírások ne akadályozzák az itt érvényes specialitások érvényesülését.

A fejlesztési projektekre kidolgozott minőségirányítási előírásokkal szembeni követelmény, hogy

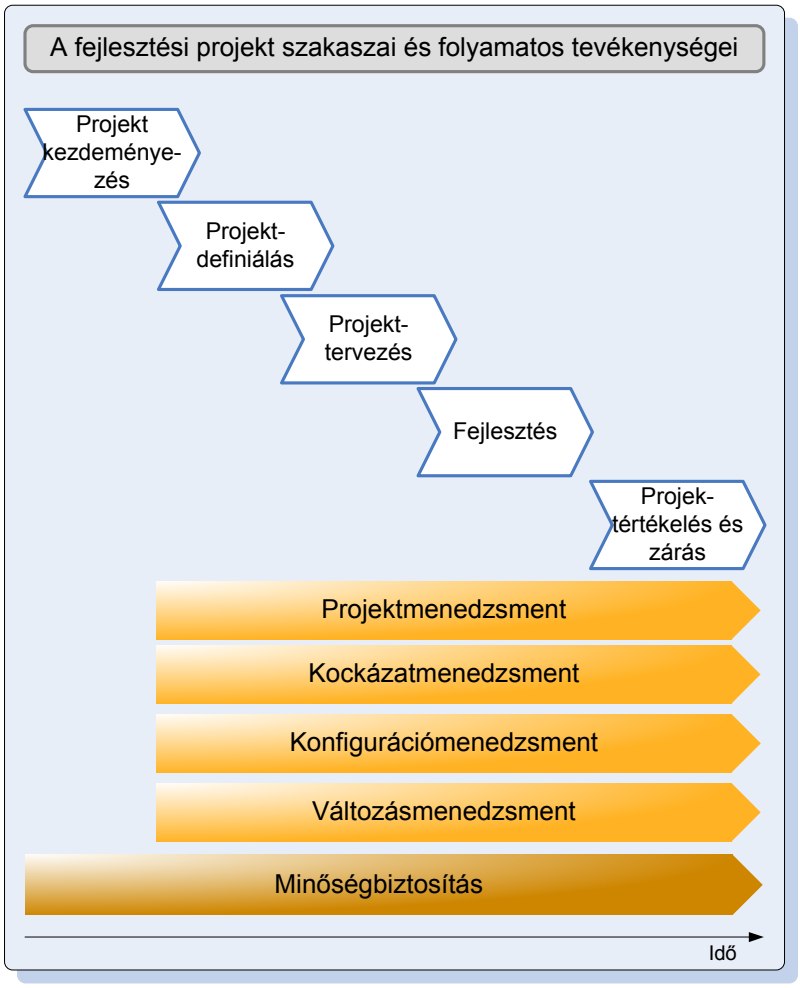
- a fejlesztési projektekben elkészített, vezetőknek szóló döntést előkészítő dokumentumokra vonatkozó előírások legyenek összhangban az általános vezetői döntési folyamatokhoz kialakított előírásokkal és folyamatokkal (pl. költségvetés jóváhagyása, munkakörnyezet és infrastruktúra biztosítása és allokálása),

- a minőségbiztosítási dokumentumok képezzenek olyan kimenetet a projekt során, mely a fentebb leírt intézményi minőségi elvárások követéséhez bemenetként szolgálhat,
- ne legyen ellentmondás az általános minőségirányítási rendszer és a projektre kidolgozott rendszer között, azonban a fejlesztési projekt specialitásait vegye figyelembe.

9. A fejlesztési projekt minőségbiztosítási eljárásai

A fejlesztési projektek minőségbiztosítási eljárásai a projekt teljes folyamatában megjelennek, tehát a minőségbiztosítási feladatokat a fejlesztési folyamatoktól elkülönülten, de velük egy időben kell végezni.

Jelen fejezet a fejlesztési projekt főbb szakaszaira és a szakaszokon átívelő tevékenységekre vonatkozóan vázolja fel a minőségellenőrzési feladatokat és azok szempontjait. Ezek a szakaszok és tevékenységek az alábbiak:



Ábra 9-1. A szoftver fejlesztési projekt szakaszai és folyamatos tevékenységei

Természetesen – ahogy a fenti ábra mutatja – a projekt szakaszai a gyakorlatban időbeli átfedéssel következnek egymás után.

A minőségellenőrzési folyamatokban leírt tevékenységek között átfedések lehetnek (pl. egyes projektmenedzsment elemek megjelenhetnek a fejlesztés minőségellenőrzési feladatai között), de nem változtat azon a követelményen, hogy ezeket a tevékenységeket el kell végezni.

A minőségellenőrzés a fejlesztési folyamatoktól kap inputokat, és eredményei visszacsatolást adnak, adott esetben változásokat indítanak el a fejlesztési folyamatban (pl. egy minőségellenőrzés következtében az informatikai rendszer követelményeit kell módosítani).

Az egyes szakaszok végén megvalósuló minőségbiztosítási tevékenység eredménye egy-egy *minőségbiztosítási jelentés*. Ennek minimális tartalmi követelményei az alábbiak:

- A jelentés jelölő adatai az általános konfigurációmenedzsment szabályainak megfelelően
- A vizsgálat célja, terjedelme és mélysége
- A vizsgálat során alkalmazott módszerek, a vizsgálat szempontjai
- A megállapítások hátterének leírása (ez biztosítja, hogy azok ellenőrizhetőek és megismételhetőek legyenek)
- A vizsgálathoz felhasznált információforrások (bemenetek)
- A vizsgálat tárgya, a bemeneti dokumentumok vizsgált tartalmi elemei
- A tartalmi szempontból nem vizsgált elemek, ennek indoklása
- Minőségbiztosítási megállapítások, eltérések és javaslatok: a javasolt korrekciós tevékenységek a feltárt minőségi problémák kiküszöböléséhez
- Az ellenőrzés során feltárt kockázatok.

9.1. Projektkezdeményezés

9.1.1. Célok

A kezdeményezési szakasz célja, az egyes területeken keletkező ötletek létjogosultságának vizsgálata, illetve elfogadás esetén, a projekt eredményeivel szemben támasztott elvárások megfogalmazása.

9.1.2. Áttekintés, termékek

A kezdeményezés szakaszban még nem beszélhetünk projektről, itt csak az „előkészületek” történnek meg. Minőségbiztosítási szempontból ez a szakasz is fontos, mivel hatással van a projekt lefutásának minőségére, így a termék minőségére is.

A projektkezdeményezés szakasza az alábbi, időben következő lépéseket foglalja magába:

- Igényfelvétel:
  az igények érkeztetése, szűrése, felülvizsgálata és dokumentálása tartozik ehhez a tevékenységhez. A tevékenység végén egy olyan írásos dokumentum készül el,

amely tartalmazza az ötletet, annak rövid leírását, regisztrációs számát, a javaslattevő és a támogató vezető aláírását.

- Előzetes döntés a projekt indításáról (döntés a probléma projekt formában történő megvalósításáról):
  az igények széleskörű bemutatása és véleményezése, azok pontosítása, javaslattétel a projekt elindításáról.

Az eddig a pontig felmerülő tevékenységek az *operatív működés keretében* történnek. Ellenben a továbbiak már egy elfogadott konkrét ötlet további vizsgálatával és adott esetben megvalósításával foglalkoznak. Itt az ötletnek már saját felelőse van, aki végigkíséri és felügyeli az ötlet életútját. Az egyes ötletek egyedi feladatainak megoldása, az ötletek további vizsgálata nem együtt, hanem külön történik a szakasz további lépéseinek megfelelően:

- Megvalósíthatósági tanulmány elkészítése:
  a megvalósíthatósági tanulmány feltárja a követelményeket (előzetes specifikáció elkészítése), azonosítja a környezeti feltételeket és függőségeket, elemzi a megvalósíthatósági alternatívákat, és megoldási javaslatot tesz. A tanulmány tartalmi elemei általában az alábbiak:
  - A megvalósíthatósági elemzés
    - okainak feltárása, az elérendő célok
    - az elemzés mélysége, terjedelme
    - az elemzés korlátjai
    - az elemzés kidolgozásának szervezete, felelősök, bevont szervezeti egységek
  - Jelenlegi helyzet bemutatása, működés célja, jellemzői
  - Környezeti feltételek, függőségek
  - A szervezet által igényelt rendszer áttekintő jellegű bemutatása;
  - Az új rendszerrel szemben támasztott követelmények részletes bemutatása
  - A javasolt megoldási alternatíva rövid bemutatása:
    - a javasolt rendszer előnyei és hátrányai,
    - a rendszer szolgáltatásainak teljesítménye,
    - megtérülés-számítás (költségek és hozadékok) bemutatása,
    - időterv.
  - A beszerzés módja
  - A megvizsgált, de elutasított alternatívák rövid bemutatása, az elutasítás okainak rövid leírása
  - Következtetés, ajánlások

- Projektindítás előkészítése és döntés a projekt indításáról:
  ebben a szakaszban az alábbi tevékenységeket ajánlott elvégezni:
  - javaslattétel a projektmenedzser személyére,
  - előzetes erőforrás és költségbecslés készítése,
  - előzetes időterv készítése,
  - javaslat kialakítása a projekt szervezetére,
  - szerepek, felelősségek, jogok definiálása,
  - projekt indítási javaslat elkészítése, ami a döntést előkészítő dokumentum.

A megvalósíthatósági tanulmány és a projektindítás előkészítésének lépéseit a szervezeti igényeknek megfelelő részletezettségben szükséges végrehajtani és dokumentálni.

9.1.3. Igényfelvétel és az előzetes döntés ellenőrzése

A szakasz minőségi kritériumainak meghatározása nem tartozik a jelen tanulmány hatókörébe, mivel ezek a folyamatok a szervezet operatív működésének részét képezik, így nem a projekt minőségbiztosítása, hanem adott szervezeti egység(ek) minőségirányítási előírásai érvényesek rá.

9.1.4. A megvalósíthatósági tanulmány ellenőrzése

9.1.4.1. Az ellenőrzés célja

A megvalósíthatósági tanulmány ellenőrzésének célja, hogy megvizsgálja

- tartalmazza-e a tanulmány a szükséges tartalmi elemeket,
- a tanulmány kidolgozottsága eléri-e azt a minőséget és részletezettséget, mely megfelelően alátámasztja a vázolt megoldási mód kiválasztását és a projekt szükségességét
- a tanulmány egyes részei konzisztensek-e egymással és a szervezet valós igényeivel.

9.1.4.2. Ki végzi az ellenőrzést?

Az ellenőrzést olyan személynek kell végeznie, aki a kívánt megoldás üzleti és műszaki környezetét ismeri és átlátja, valamint tisztában van a szervezet döntési mechanizmusaival, és a döntésekhez szükséges információkkal.

9.1.4.3. Input

A *vizsgálat bemenetei*, információ forrásai az alábbiak lehetnek:

- Megvalósíthatósági tanulmány és a felhasznált háttéranyagok

- Interjújegyzőkönyvek
- Személyes megbeszélések a szervezet munkatársaival
- A szervezet formai és tartalmi előírásai a dokumentumra vonatkozóan (pl. megtérülés-számítási szabályok, adatok)

9.1.4.4. Minőségellenőrzési feladatok

A megvalósíthatósági tanulmánnyal szemben az alábbi *minőségi követelményeket* támaszthatjuk, és az alábbi ellenőrzési *feladatok* elvégzését javasoljuk:

- Döntéstámogatási funkció megvalósítása:
  vizsgálni kell, hogy a tanulmány alkalmas-e döntéshozatalra, az elemzés mélysége megfelelő-e arra, hogy a problémákat és a megoldásokat bemutassa. A „fejleszteni vagy venni döntés" megfelelően megalapozott-e, valamint megtörtént-e a fejlesztés elmaradásának hatásvizsgálata.
- Megtérülés-számítások helyessége:
  vizsgálni kell, hogy a tanulmány tartalmaz-e megtérülés-számításokat, és a bevétel és költségszámítások kellően megalapozottak-e. Mérlegelni kell, hogy a működési hasznosság világos és részletezett-e.
- Vizsgálat kereteinek bemutatása:
  vizsgálni kell, hogy a tanulmány bemutatja-e az elemzés kereteit és korlátait, világossá teszi-e a döntéshozók számára, hogy a tanulmány milyen peremfeltételek mellett képes a döntés-előkészítésre, és mely eseményekkel, környezeti tényezőkkel nem számol.
- Tervezés megfelelő minősége, konzisztenciája:
  vizsgálandó, hogy a tanulmány tartalmazza-e a fejlesztéshez szükséges erőforrásokat, idő- és költségtervet és ezek összhangban vannak-e, nem szerepelnek-e bennük ellentmondó becslések, valamint a kitűzött feladatok reálisan megvalósíthatók-e ezek alapján.
- Az igények és a leírt követelmények, a javasolt megoldás(ok) összhangja:
  a bemutatott megoldás, vagy megoldások a leírt problémákra nyújtanak megoldást, valamint terjedelmük (scope) megfelelő.
- A megvalósíthatósági tanulmány elkészítésére kialakított szervezeti keretek megfelelősége:
  az elemzést készítők felelőssége és hatáskörei összhangban vannak-e, módjuk van-e a szükséges erőforrások mozgósítására, valamint arányos-e a befektetett munka a várható költségekkel. Azt is javasolt ellenőrizni, hogy a releváns szereplők be vannak vonva a munkába, és mi alapján történt az érintettek bevonása.
- Beszerzési terv helyessége:
  az elemzésben szereplő beszerzési terv összhangban van-e a szervezet szabályzataival és a vonatkozó törvényi szabályozásokkal. Az erre vonatkozó időbeli korlátokat figyelembe vették-e a tervezésnél.

- A környezeti feltételek azonosítása megtörtént:
  figyelembe vették-e a vállalati szabályokat, stratégiákat, fejlesztési irányelveket és előírásokat, technikai, technológiai feltételeket (architektúra, kapacitás), valamint a szervezeti, illetve szervezeteken átívelő befolyásoló tényezőket (pl. várható törvényi változások, szervezeti változások, stb.).
- Megfelelő háttéranyagok felhasználása:
  meg kell vizsgálni a felhasznált háttéranyagok forrását, azok megbízhatóságát, és azt, hogy az elemzés elkészítése és a megoldás (megoldási alternatívák) kidolgozása során figyelembe vették-e a témában meglévő legjobb gyakorlatokat. Vizsgálandó, használtak-e külső független információforrásokat, gyűjtöttek-e információkat az informatikai piacon fellelhető eszközökről, megoldásokról.
- Az elkészült termékek bizonylatolása:
  annak ellenőrzése, hogy a tanulmányt a megfelelő személynek jóváhagyásra megküldik és ennek formai követelményei teljesülnek.
- A tanulmány elkészítési folyamatának helyessége:
  javasolt ellenőrizni, hogy a tanulmány elkészítése során az elvégzett interjúk emlékeztetői elkészültek-e a szükséges mértékben és minőségben, a disztribúció során a verziómenedzsment hiányosságai nem okoztak-e nehézségeket, hibákat. Az elkészítési folyamatba bevont szervezeti egységek ténylegesen közreműködők voltak-e.

9.1.5. A projektindítás előkészítésére és a projekt indításáról történő döntésre vonatkozó ellenőrzés

9.1.5.1. Az ellenőrzés célja

Az ellenőrzés célja, hogy megvizsgálja, hogy a döntés-előkészítő dokumentum megfelel-e a kívánt célnak, a javasolt megoldást megfelelően, konzisztensen alátámasztja-e a szervezetben meglévő követelményeket kielégítve.

9.1.5.2. Ki végzi az ellenőrzést?

Az ellenőrzést a megvalósíthatósági tanulmánynál leírt képességekkel rendelkező személynek célszerű elvégeznie, javasolt, hogy ugyanaz a személy, vagy személyek működjenek közre, akik az előző szakaszokban is.

9.1.5.3. Input

A *vizsgálat bemenetei*, információ forrásai az alábbiak lehetnek:

- Az elkészült döntés-előkészítő dokumentum
- A megvalósíthatósági tanulmány
- Projektmenedzserre tett javaslat, valamint információk a személyről
- A szervezet formai és tartalmi előírásai a dokumentumra vonatkozóan

9.1.5.4. Minőségellenőrzési feladatok

Az ellenőrzés során az alábbi szempontokat kell figyelembe venni:

- A projektmenedzserre történő javaslattétel összhangban legyen a megvalósíthatósági tanulmány javaslataival, a javasolt projektmenedzser rendelkezzen az ott leírt feladatokhoz szükséges képességekkel, tapasztalatokkal.
- Fel kell mérni, hogy a meghatározott szerepek, felelősségek és jogok összhangban vannak-e, illetve megfelelnek-e a projekt szakmai követelményeinek.
- Az elkészített erőforrás, idő- és költségbecslés, valamint a projektszervezetre vonatkozó javaslat, valamint a projektindító javaslat
  - összhangban van-e a megvalósíthatósági tanulmánnyal, és az eltérések indoklása és jóváhagyása megtörtént-e;
  - részletezettsége és tartalma megfelel-e a szervezetben meglévő döntés-előkészítő dokumentumokra vonatkozó előírásokkal, a döntéshozó szervezeti egység követelményeivel.
- Vizsgálni kell, hogy a projekt finanszírozási oldala megfelelően alátámasztott és teljes.

9.2. Projektdefiníciós szakasz

9.2.1. Célok

Jelen szakasz célja a *bemutatott és elfogadott* megvalósítási alternatíva paramétereinek további pontosítása, és részletezése.

9.2.2. Áttekintés

A projektdefiníciós szakasz, a projektcél pontos meghatározását, mérhetővé tételét, a végtermék specifikálását, környezetének vizsgálatát, a projekt definíció megalkotását foglalja magában.

A szakasz további feladata a projekt kereteinek (költségek, emberek, eszközök) becslésének pontosítása az előzetes tervek alapján. A projektdefiniálás egyik legfontosabb része a célrendszer összeállítása, ami tartalmazza a projekt átfogó és részletes céljait.

A definíciós szakasz elvégzendő feladatai a következők:

- Stakeholder analízis
  A stakeholder analízis a projekttel szembeni elvárások meghatározása minden szereplő bevonásával. A stakeholder analízis során azon személyek csoportját keressük, akikre a projekt hatással van, akik hatással vannak a projektre vagy valamilyen módon kapcsolatban állnak a projekttel. Így tudjuk megállapítani a legszélesebb körű igényeket a projekt termékére vonatkozóan.

Stakeholder: Eredeti jelentése „Cölöptulajdonos”. Az Amerikai Egyesült Államokban, azon földfoglalókat hívták így, akik megszerezték egy adott földterületre a tulajdonjogot (nevüket ráírták az azt határoló cölöpre). A szó jelentése a későbbiekben megváltozott, és a stakeholder mint „haszonélvező” került a köztudatba. Napjaink üzleti és vállalati környezetében is inkább a haszonélvező jelentést használjuk, megtartva a szó eredeti, idegen nyelvű alakját.

- Célrendszer meghatározása
  Itt kerülnek megfogalmazásra a projekt céljai, terjedelme, illetve nem-céljai. Meg kell fogalmazni mi az, amit várunk a projekttől, milyen tényezők megvalósulása elengedhetetlen. Fel kell sorolnunk a projekt által létrehozott fizikai és szellemi termékek listáját. Meg kell határozni a sikerkritériumokat.
- Előzetes projektterv elkészítése (becslés)
  Ebben a szakaszban kell elkészíteni a projekttervet, a termékhálót (a projekt végrehajtása során elkészülő résztermékek függőségi kapcsolata), fő feladatcsoportokat, és azok első szintű részletezését, idő-, erőforrás-, költségtervet és a kommunikációs tervet.
- Projektstruktúra létrehozása
  Ennek keretében zajlik a projekt belső működésének megtervezése és felépítése, a projekt megvalósítására javasolt projektszervezet véglegesítése, annak szerepeinek a feladat ellátásához szükséges jogokkal történő ellátása, és a feladat nagyságától, fontosságától függően felelősséggel való felruházása. A kialakult szerepekhez személyeket rendelnek és juttatási, motivációs rendszert alakítanak ki, és megtörténik a projekttagok formális kirendelése a projektbe.
- Projektdefiníció elkészítése
  A projektdefiníciós szakaszban létrejött eredmények dokumentumba foglalása. Ellentmondásmentessé kell tenni, és egymással összhangba kell hozni a projekt céljait, terjedelmét, megközelítését, a sikerkritériumokat, költség-, erőforrás- és időtervet, valamint a projekt szervezetét.
- Döntés a projektdefiníció elfogadásáról
  Az előzőek alapján összeállított és ellenőrzött projektdefiníció elfogadása.

9.2.3. Termékek

A szakasz terméke a *Projektdefiníció* (vagy más néven *Projektalapító Dokumentum*), mely a feladatoknál részletezett elemeket tartalmazza. A dokumentum minimálisan elvárt tartalmi elemei:

- A projekt háttere röviden
- A projekt pontos céljai, célrendszere, sikerkritériumok
- A projekt terjedelme
- A megoldás, megközelítés tömör leírása
- Projektszervezet (szerepek, felelősségek – a résztvevők neveivel)
- Becsült projektterv, költségterv

- Kommunikációs terv
- Kockázatmenedzsment terve

9.2.4. A Projektdefiníció ellenőrzése

9.2.4.1. Az ellenőrzés célja

Az ellenőrzés során meg kell győződni arról, hogy

- a Projektdefiníció pontosan azokat a célokat fogalmazza meg, amiket az eddigiek során kialakítottak, és amire a szervezetnek és az egyes stakeholdereknek igénye van,
- az egyes részcélokat és szempontokat megfelelően hangolták össze,
- a leírt célok, időkeret és erőforrásterv összhangban van.

9.2.4.2. Ki végzi az ellenőrzést?

Az ellenőrzést olyan személy végzi, aki tisztában van a projekt szerepével és céljaival, valamint a szervezet üzleti és műszaki szempontú igényeivel. Javasolt, hogy az előző fázisokban részt vett szakértők végezzék az ellenőrzést.

9.2.4.3. Input

- Stakeholder analízis eredményei (interjújegyzőkönyvek, személyes beszélgetések, elkészített dokumentumok)
- Az előző fázisokban elkészített, a követelményekre vonatkozó dokumentumok.
- Projektdefiníció (vagy Projektalapító Dokumentum)
- A projektet indító szervezet (intézmény) szervezeti felépítése, a munkatársak projektre allokálásának rendje és a projektben résztvevő munkatársak és egyéb erőforrások allokálásával kapcsolatos dokumentumok

9.2.4.4. Minőségellenőrzési feladatok

A projektdefiníció elfogadásáról szóló döntést a projektdefiníció minőségellenőrzésének kell megelőznie, amely tartalmazza az alábbiakat:

- célrendszer megfelelőségének ellenőrzése:
  - a korábban összegyűjtött követelményeknek összhangban kell lenniük a Projektdefinícióban leírt célrendszerrel,
  - stakeholder analízis során elkészített követelményeket tartalmazza-e a projektdefiníció
- a stakeholder analízis során minden stakeholdert, minden érdekeltet bevontak-e a munkába, illetve figyelembe vették-e azok szempontjait. A célok meghatározásánál megfelelő súllyal vették-e figyelembe a szempontokat.

Megjegyzés: csak abban az esetben szabad továbblépni, ha a felhasználók biztosítva látják, hogy a definiált feltételekkel a projekt ellenőrizhető módon, azt a terméket fogja leszállítani, amit ők előre specifikáltak. Ez biztosítja a projekt elfogadottságát a szervezetben.

- A projekt terjedelme egyértelműen, jól körülhatárolt, a célok megvalósulását a bemutatott erőforrás- és időtervek lehetővé teszik.
- Következetességvizsgálat: vizsgálni szükséges, hogy a projektdefinícióban részletesen meghatározott tervek, erőforrások, megoldások, feladatok következetesen levezethetőek és nem állnak ellentmondásban e projektdefiníciót megelőző, előkészítő fázisokban készített dokumentumokkal.
- Projektszervezet: vizsgálandó, hogy elegendő erőforrás jut-e a projekt feladatainak elvégzésére, valamint, hogy a tagoknak munkaköri feladataikra marad-e elegendő idejük, okoz-e majd feszültségeket a szervezeti egységükből történő „kikérés”.
- Projektköltségvetés és erőforrások vizsgálata: a projekt keretszámainak ellenőrzése során meg kell vizsgálni, hogy a rendelkezésre álló információk birtokában, az adott feltételek mellett, a projekt megvalósítható-e.

Ha minden érdekelt fél elfogadja a projektdefinícióban leírtakat, akkor kezdődik a következő – tervezési - szakasz.

9.3. Projekttervezés

9.3.1. Célok

A tervezési szakasz a projekt legkritikusabb fázisa, ennek sikeressége a projekt sikerességének záloga. A tervezés célja egy *minden részletre kiterjedő terv* megalkotása, a projekt végrehajtására vonatkozóan.

Itt történik meg a projektkeretek pontos meghatározása, hiszen a szakasz végén elkészülő projekttervben már azok az adatok szerepelnek, amelyekhez a későbbiekben viszonyítjuk a projekt megvalósulását.

9.3.2. Áttekintés és termékek

A tervezési szakasz elvégzendő feladatai a következők:

- Projektalapító értekezlet (kick-off meeting), munkaterv elkészítése
  A tervezési szakasz a projektindító megbeszéléssel (kick-off meeting) kezdődik. A projektindító megbeszélés a projekt minden résztvevője számára szervezett nyitó értekezlet, de egyben ötlet és információgyűjtés is. Ez a megbeszélés teljes egészében a Projektdefiníció (Projektalapító Okirat) tartalmának megismertetéséről szól (amennyiben az előző szakaszban részt nem vett munkatársak dolgoznak a projektben). Ez alapján történik a projektmegbízó és a projektszervezet közötti megbízás megkötése.

- Menedzsmentterv elkészítése
  A menedzsmentterv elkészítése során történik a szervezet véglegesítése, minőségbiztosítási terv, kockázatelemzés, kockázatmenedzsment-terv, adminisztrációs terv, kommunikációs terv kidolgozása.
  - Minőségbiztosítási terv: a projekt végrehajtása során a minőségbiztosító gondoskodik a projekttermék és a projektfeladatok végrehajtásának elvárt minőségéről. A minőségbiztosítási terv a minőségbiztosító munkájának részletes terve. Definiálja a munka módszerét, időzítését, szereplőit. A projekt méretétől függően lehet formális (írásos) vagy informális tervet készíteni.
    Ennek tartalmát a tanulmány későbbi részében fejtjük ki.
- Egyesített projektterv elkészítése
- Döntés a projekt végrehajtásáról

9.3.3. A menedzsmentterv ellenőrzése

9.3.3.1. Az ellenőrzés célja

Az ellenőrzés célja, hogy a fent felsorolt termékek elkészültét és helyességét ellenőrizze.

9.3.3.2. Ki végzi az ellenőrzést?

Az ellenőrzést a projektek gyakorlati végrehajtásában járatos embernek célszerű végeznie, mivel annak gyakorlati kérdései kerülnek előtérbe.

9.3.3.3. Input

- Projektdefiníció (Projektalapító Dokumentum)
- Projektindító megbeszélés jegyzőkönyve

9.3.3.4. Minőségellenőrzési feladatok

- Ellenőrizni kell, hogy a szervezet a Projektdefinícióban elfogadottaknak megfelelően lett-e kialakítva, és a formai, bizonylatolási követelményeknek (pl. munkaerő-allokációs lapok kitöltése) eleget tettek-e.
- Ellenőrizni kell, hogy a kockázatelemzés, kockázatmenedzsment-terv teljes körűen számba vette-e a kockázatokat és szakmailag helyes kezelési módokat határozott-e meg.
- Vizsgálni kell, hogy az adminisztrációs terv megfelelően rögzíti-e a projekttitkár, illetve az asszisztensek feladatait, és azok elvégzésének módját. Ezek illeszkednek-e a projekt méretéhez és jellegéhez, valamint a szervezet és a törvények által meghatározott környezethez.

- Vizsgálni kell, hogy a kommunikációs tervben leírt belső és külső kommunikációs formák alkalmazhatóak-e és hatékonnyá teszik-e a projektmunkát, a körülményeket, a projekt méretét és jellegét figyelembe véve.

9.3.4. A projekt terv ellenőrzése

9.3.4.1. Az ellenőrzés célja

Az ellenőrzés célja a projekttervezés összehangoltságának és megvalósíthatóságának biztosítása a tervezési szakasz végén, valamint a szakasz folyamán is.

9.3.4.2. Ki végzi az ellenőrzést?

Az ellenőrzést olyan szakértőnek kell végeznie, aki rendelkezik tapasztalattal a fejlesztési projekteket illetően, és szakmailag meg tudja ítélni a tervezés helyességét.

9.3.4.3. Input

- Projektterv
- Projektdefiníció (Projektalapító Dokumentum)
- Projektindító megbeszélés jegyzőkönyve

9.3.4.4. Minőségellenőrzési feladatok

Ellenőrizni kell, hogy a projekt terv

- Biztosítja-e, hogy egy feladat se maradjon ki.
- Biztosítja-e, hogy jusson idő és erőforrás a minőségbiztosítási feladatok ellátására,
- világosan tartalmazza-e a projekt tagjainak felelősségét, kötelezettségét és hatáskörét.
- előre meghatározza-e az egymástól kölcsönösen függő feladatokat, így a funkcionális egymásrautaltságokat is.
- a terv a helyzet, a projekt, a projektmenedzser és a teamtagok eredményességének vagy eredménytelenségének megítéléséhez nyújt megfelelő alapot.
- a terv a megfigyelés, a nyomon követés és a kontrolling mellett a kommunikáció és az ellenőrzés eszköze is lehet.

A projekttervezés eddig leírt kritériumain felül az ellenőrzés során rendszeresen szakmai kontrollt érdemes biztosítani, melyben a tervezés helyességét, valamint a célok, erőforrások és az allokált időtartamok összehangoltságát ellenőrzik.

Megjegyzés: az intézménynek figyelembe kell vennie, hogy a közbeszerzési folyamatra tekintettel a javasolt megoldás leírását és a funkcionális specifikációt az ajánlati felhívás kiadását megelőzően ki kell dolgoznia és jóvá kell hagynia a hatékony kiválasztási procedúra lefolytatásához.

9.4. Szoftverfejlesztési folyamatok

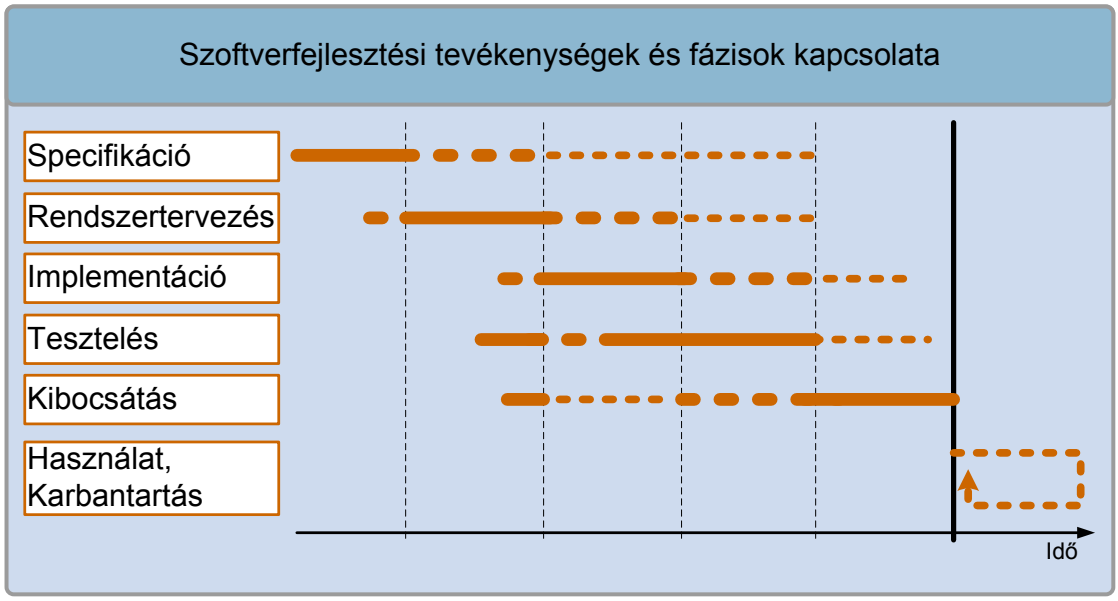
9.4.1. Áttekintés

A jelen dokumentumnak nem célja kifejteni, hogy hogyan kell a szoftverfejlesztési folyamatokat végrehajtani, csak azt határozza meg, hogy hol vannak bennük a javasolt minőségbiztosítási pontok.

A felsorolt fejlesztési folyamatok függetlenek az alkalmazott rendszerfejlesztési életciklus modelltől, a leírt koncepció bármilyen életciklus modell esetében alkalmazható.

A fejlesztési folyamatok egymást követik, de nem feltétlenül kell véget érnie az egyiknek, hogy a következő elkezdődjék. Sőt a fejlesztés során előfordulhatnak ciklusok is, amikor pl. a tesztelés eredményei alapján a követelményeket kell módosítani. A minőségbiztosítási tevékenység is hozzájárulhat ahhoz, hogy a fejlesztési folyamatok tartalma és ütemezése módosuljon.

A szoftverfejlesztési folyamatokat érdemes két dimenzió mentén szemléltetni: a függőleges tengelyen az egyes tevékenységcsoportok, a vízszintes tengelyen az idő előrehaladása és azon a szoftverfejlesztés különböző fázisai láthatók.



Ábra 9-2. A szoftverfejlesztési tevékenységek és fázisok kapcsolata

A különböző módszertanok különböző módon határozzák meg a fázisok számát, elnevezését és határait, de az egyes tevékenységcsoportok megoszlása a fázisok során jellemzően hasonló.

- Specifikációs tevékenységek

- A szoftverfejlesztési folyamat elején meghatározásra kerülnek a követelmények.
- Minél átgondoltabb és teljesebb körű a specifikáció, annál jobb alapanyagot jelent a tervezési és megvalósítási munkához, tehát ideális esetben a tervezés megkezdése előtt előáll egy teljes és lezárt specifikáció. Ez a gyakorlatban ritkán van így, sokkal gyakoribb, hogy a projekt hátralévő időtartalma alatt is némileg finomodnak, pontosabbá válnak a követelmények. Ez a megfelelő változáskezelési folyamatok alkalmazása esetén nem veszélyezteti a projekt sikerét, ugyanakkor lehetővé teszi, hogy a megrendelő igényeinek jobban megfelelő rendszer alakuljon ki. Ezért az ábrán a specifikációtevékenységek egyre csökkenő intenzitással, de sokáig szerepelnek.

- Rendszertervezési tevékenységek
  - A műszaki tervezés a specifikálást követően a legintenzívebb, de már a specifikálással párhuzamosan is megkezdődhet, hiszen a rendszer műszaki koncepciójának kialakítása oda-vissza hathat a követelményekre.
  - Sok módszertanban, illetve a gyakorlatban is, a tervezés és az implementálás nem szigorúan egymást követő tevékenységek, hanem sokszor időbeli átfedés tapasztalható. Tehát a tervek bizonyos készültsége esetén (jellemzően a logikai szint kidolgozását követően) az implementáció már megkezdődik, és a részletes fizikai tervezés az implementációval párhuzamosan halad.
- Implementációs tevékenységek
  - Az implementáció a tervezés bizonyos készültségi szintjénél már megkezdődhet, ezután átveszi a domináns szerepet, majd később a tesztelésekkel párhuzamosan csökken az intenzitása (többnyire már csak hibajavításokra korlátozódik), és végül a kibocsátással lecseng.
  - A tervezéssel párhuzamosan szükség lehet bizonyos implementációs tevékenységek elvégzésére is, pl. prototípus, vagy „deszka modell”[1] (megvalósíthatósági tesztek) készítése esetén.
- Tesztelési tevékenységek
  - A tesztelési tevékenységek megkezdését a tesztek megtervezése jelenti, mely a rendszertervezéssel párhuzamosan már elindulhat.
  - A tesztrendszerek és tesztkörnyezetek felállítása, a tesztadatok beszerzése vagy előállítása, a tesztesetek összegyűjtése mind-mind meg kell, hogy történjen a tervezés végével és az implementálás megkezdésével

[1] Olyan modell, amely egy kezdetleges megvalósítása az alkalmazás egy funkciójának, vagy jellemzőjének. Ez a modell alkalmas arra, hogy a funkció, vagy jellemző megvalósíthatóságára az adott technológiával választ kapjunk. Egyes módszertanok ezt tesztlabornak, vagy koncepció bizonyítási teszteknek nevezik.

párhuzamosan, annak érdekében, hogy az implementációt némi elcsúsztatással folyamatosan kísérhesse a már elkészült részek tesztelése.

- Bár az implementálás és a tesztelés tevékenységei jellemzően párhuzamosan haladnak, kell lennie egy (vagy több iteráció esetén több) előre meghatározott pontnak, melynél a vezető szerep felcserélődik: először az új funkciók fejlesztését kíséri és szolgálja ki a tesztelés, majd az új funkciók fejlesztése leáll, és attól kezdve az implementálók kizárólag a tesztek által felszínre hozott problémák megoldásával foglalkoznak.

- Kibocsátási tevékenységek
  - A kibocsátási tevékenységekbe értjük a rendszer bevezetésével kapcsolatos oktatást, próbahasználatot (pilot), telepítést, migrálást, ami összetett rendszerek esetén bonyolult és hosszú folyamat is lehet.
  - A kibocsátás megtervezése már a rendszertervezéssel párhuzamosan megkezdődik.
  - A kibocsátással kapcsolatos tevékenységek intenzitása a tesztelések vége felé emelkedik meg, az oktatások sokszor például egy már kellően stabil, de még tesztelés alatt levő rendszeren történnek.
  - A rendszer teljes körű éles indulása után még lehet egy intenzív támogatási időszak, mikor a fejlesztési projektcsapat egy része segíti az üzemeltetőket a napi munkához szükséges tudás és tapasztalat megszerzésében.
- Használat / karbantartás
  - Ezek a tevékenységek általában a projekt határain kívül, de a szoftver életciklusán belül zajlanak. A felhasználás során, a rendszer méretétől függően akár évekig is kerülhetnek elő hibák, merülhetnek fel új igények.
  - Próba használat esetén a projekt részeként is előkerülnek az ide tartozó tevékenységek.
  - A használat / karbantartás akkor válik meghatározóvá, ha az üzemeltetés már átvette a feladatokat a megvalósító csapattól, a használat üzemszerűvé válik.

Az alábbiakban rövid áttekintést adunk a szoftverfejlesztési folyamatokról, azok termékeiről, majd felsoroljuk a folyamatokban azonosítható minőségbiztosítási pontokat.

Megjegyzés: A szoftverfejlesztési folyamatok és a minőségbiztosítási tevékenységek végrehajthatók egymástól izoláltan (az előállított termékek „átdobása a falon” a minőségbiztosítókhoz, és viszont), azonban sokkal hatékonyabb, ha a fejlesztést és a minőségbiztosítást személyi és folyamat szinten is integráljuk. Ennek azonban ellene hathat egy másik szempont: a szerződéses viszonyok, illetve a megrendelő és a szállító közötti felelősségi vonal tiszta meghúzása. A fejlesztési és a minőségbiztosítási folyamatok szigorú elválasztásáról vagy integrálásáról tehát egyedileg kell döntést hozni, nem adható általános irányelv.

9.4.2. Specifikációs folyamatok

9.4.2.1. Áttekintés

A specifikációs tevékenységek során a rendszerrel szemben támasztott igények és elvárások felmérése, priorizálása, dokumentálása és számonkérhető kritériumokká alakítása folyik.

Az igények és elvárások különböző forrásból származhatnak (megrendelők, végfelhasználók, üzemeltetők, objektív műszaki korlátok, jogszabályok, közmegegyezés, stb.). Így azok között könnyen előfordulhatnak ellentmondások is. A specifikációs tevékenység részeként, a követelmények összevetése, elemzése során kell az ellentmondásokat azonosítani és feloldani.

Az igények megismerésének elsődleges eszköze a korábbi dokumentációk feldolgozása, meglévő megoldások és a célkörnyezet feltérképezése, az interjú, a megbeszélés és a workshop, de nagy számú résztvevő esetén hasznos lehet a kérdőíves felmérés is. A műszaki peremfeltételek meghatározásához szükség lehet műszaki felmérésekre és számításokra is (pl. jelenlegi hardverkörnyezet felmérése), de figyelembe kell venni az esetleges jogszabályi előírásokat is (pl. adatvédelmi előírások).

Megjegyzés: Fontos tudatosítani a megrendelő és a felhasználó közötti különbséget: a megrendelő az, aki számára a megvalósítandó rendszer hasznos, értékes (és nem utolsó sorban, aki közvetve vagy közvetlenül finanszírozza azt), míg a felhasználó az, aki a szoftvert eszközként használni fogja a napi gyakorlatban. Ezért érdekeik is különbözhetnek, például a felhasználók számára fontosabbak lehetnek egyes kényelmi szempontok, míg a megrendelő inkább az üzleti hasznosságot és a befektetés megtérülését vizsgálja.

9.4.2.2. Célok

A specifikációs folyamatok célja: a projekt további tevékenységei számára pontosan definiálni, hogy MIT kell implementálni. (A HOGYAN-nal a rendszertervezési folyamatok foglalkoznak.)

Megjegyzés: A specifikálás során javasolt az igényeket osztályozni aszerint, hogy mennyire kritikus fontosságúak a rendszer használatának megkezdése szempontjából. Ilyen módon megkülönböztethetünk feltétlenül megvalósítandó („alap”), és nem létfontosságú („kiegészítő”) funkciókat, vagy akár számszerű prioritási szinteket is használhatunk. Ez az osztályozás lehetővé teszi a rendszer több verzióra bontását, így az első verzió előbb elkészülhet, és üzembe állhat, amivel javítható a szoftverfejlesztési befektetés megtérülése, illetve csökkenthetőek a hosszú megvalósítási időtartam és a nagy költség jelentette kockázatok.

9.4.2.3. Termékek

- Interjú és megbeszélés-jegyzőkönyvek, emlékeztetők, kérdőívek (opcionális)
- Műszaki felmérések jegyzőkönyvei (opcionális)

- Követelményspecifikáció (kötelező)
- Fogalomtár (kötelező)

Megjegyzés: A követelményspecifikáció és a fogalomtár tartalmilag egy-egy egység, ugyanakkor formailag különböző módon is megjelenhetnek. Lehetséges, hogy a fogalomtár a követelményspecifikáció dokumentumának egy fejezete, de ugyanígy előfordulhat az is, hogy a követelményspecifikáció több fizikai dokumentumra bomlik: pl. külön a funkcionális és a nem-funkcionális igények. A funkcionális igények formális ábrázolásához készülhet pl. használati esetmodell, ami megint csak külön dokumentumot képezhet.

9.4.2.4. A követelményspecifikáció ellenőrzése

9.4.2.4.1. Az ellenőrzés célja

A szoftverfejlesztési projekt hátralévő tevékenységei a specifikációban lerögzített követelményekre épülnek, ezért a specifikáció minőségi problémái a teljes fejlesztési folyamaton végiggyűrűznek, és minél később kerülnek azonosításra és korrigálásra, annál nagyobb plusz költséget okoznak.

A követelményspecifikáció-dokumentum ellenőrzésének célja tehát arról meggyőződni, hogy a specifikáció helyesen írja-e le az elkészítendő rendszert – nehogy az elkészülés után derüljön ki, hogy rosszul volt meghatározva a feladat.

9.4.2.4.2. Ki végzi az ellenőrzést?

Az ellenőrzést olyan személynek kell végeznie, aki a rendszer üzleti és műszaki környezetét ismeri és átlátja.

Megjegyzés: Előfordulhat, hogy az ellenőrzés szerepe több személy között oszlik meg, tipikusan az üzleti és a műszaki aspektus átlátása és megítélése kívánhat meg más-más kompetenciával rendelkező személyeket.

9.4.2.4.3. Input

A követelményspecifikáció ellenőrzése az alábbi inputokra épül:

- Minőségi célelvárások a specifikációval szemben.
- Megvalósíthatósági elemzés (korábbi projektszakasz terméke)
- Projektdefiníció (korábbi projektszakasz terméke)
- Stakeholder analízis (korábbi projektszakasz terméke, ha van)
- Interjú és megbeszélés-jegyzőkönyvek, emlékeztetők (ha van)
- Műszaki felmérések jegyzőkönyvei (ha van)
- Követelményspecifikáció
- Fogalomtár
- Korábban készült, a specifikáció elkészítéséhez használt dokumentációk.

- Ellenőrző interjúk, felmérések, illetve ezek jegyzőkönyvei

9.4.2.4.4. Minőségellenőrzési feladatok

Az ellenőrzés a rendelkezésre bocsátott dokumentumok vizsgálatával történik.

> Megjegyzés: Az inputok formátuma, nyelvezete, a választott ábrázolási módja különböző is lehet; a minőségbiztosítónak ezektől elvonatkoztatva a tartalmának összevetésére kell koncentrálnia. (Kivéve, ha a specifikáció formalizmusaira vonatkozóan is fogalmazott meg elvárásokat a projekt.)

Vizsgálandó, hogy az alábbi témakörök mindegyike megjelenik-e a követelmény specifikációban.

> Megjegyzés: Előfordulhat, hogy egy adott követelményfajta nem releváns egy adott rendszerre nézve, de ebben az esetben ezt explicit módon szerepeltetni kell a specifikációban; máskülönben az ellenőrzést végző személy nem tudja eldönteni, hogy egy követelményfajta hiánya szándékos, vagy véletlen-e.

- Funkcionális követelmények (jellemzően ez képezi a specifikáció túlnyomó részét)
  - Bár a leírás formáját nem kívánjuk megkötni, de az mindenképp követelmény, hogy a leírás alapján a funkciók felsorolhatóak legyenek (ellenőrző listaként felhasználhatóak legyenek), és kifejtettségük szintje egyenletes legyen.

> Megjegyzés: Legjobb gyakorlatnak tekinthető ún. Használati eset modell (Use Case model) készítése, mely szabványos jelölésrendszerben és áttekinthető lebontásban láttatja a funkciókat. A Használati eset modell kialakítását, valamint a megrendelői és a kivitelezői oldal közös nyelvezetének megtalálását nagyban segíti az ún. Használati eset munkaértekezlet, melynek során a két oldal képviselői közösen, szisztematikusan térképezik fel és értelmezik saját maguk és a másik fél számára a megvalósítandó rendszer funkcionalitását. A Használati eset munkaértekezleten elhangzottakat érdemes hangfelvétellel, vagy jegyzőkönyvvel dokumentálni későbbi referencia végett.

- Minőségi, vagy nem funkcionális követelmények: az alábbi minőségi követelmények közül a relevánsak meglétének ellenőrzése
  - Felhasználói felület
    - Ajánljuk rögzíteni a felhasználó és a rendszer közti interfészek logikai szerkezetét. A részletes felületi terveket nem ez a dokumentum tartalmazza, ide csak az irányelveket kell felvenni, melyet természetesen néhány képernyőterv is szemléltethet. Néhány példa, mit érdemes itt rögzíteni: hivatkozás GUI (Graphical User Interface – Grafikus felhasználói felület) standardokra, iránymutatásokra; szabvány betűtípusok, gombok elrendezése, igényelt képernyőfelbontás, üzenetek megjelenítésével kapcsolatos konvenciók; igényelt képernyő-elrendezés.
  - Hardverinterfészek

    - Hardver közeli megoldásoknál itt rögzítjük a szükséges eszközmeghajtókat, a támogatott hardvereszközöket, hardverkommunikációs protokollokat.
- Szoftverinterfészek
    - Itt kerülnek ismertetésre a kifejlesztésre kerülő rendszer és más szoftver komponensek közti kapcsolatok. Itt kell megadni a rendszer működéséhez szükséges adatbázis, operációs rendszer, könyvtárak, komponensek nevét és verzióját. Megadandó a kapcsolat típusa is.
    - Itt kerülnek ismertetésre a kifejlesztésre kerülő rendszerkommunikációs funkcióival kapcsolatos elvárások. Pl. milyen protokollok használhatók (HTTP, FTP), milyen eszközöket (böngésző típus és verzió) kell támogatni, milyen titkossági, adatátviteli, szinkronizációs követelmények léteznek.
- Biztonsági követelmények
    - Felhasználók és rendszerek azonosítása
    - Felhasználók és rendszerek feljogosítása
    - Tárolt adatok védelme
    - Átvitt adatok védelme
    - Audit naplózása
- Használhatósági követelmények
    - Milyen jellegű felhasználói képzés a megcélzott (oktatás vagy önálló betanulás; on-line segédlet, vagy kézikönyv, stb.)
    - Esetleges időkorlátok meghatározása (betanulási idő, adott feladatok elvégzésének időkorlátai, stb.)
    - Használhatósági és „design” szabványok előírása
- Megbízhatósági követelmények
    - Üzemidőszak meghatározása (pl. munkanapokon 8-18h vagy 7x24)
    - Rendelkezésre állás (a teljes üzemidő hány százalékában kell elérhetőnek lennie), Mean Time Between Failures (MTBF) – mennyi időnként következhet be leállás, Mean Time To Repair (MTTR) – mennyi ideig lehet működésképtelen a rendszer egy leállás során
- Teljesítménykövetelmények
    - Válaszidők a különböző műveletekre nézve (átlagos, maximális)
    - Átviteli teljesítmény (pl. tranzakció per másodperc)
    - Kapacitás (pl. maximális felhasználószám, vagy tranzakciószám)
    - Megengedett degradáció mértéke és időtartama (pl. hardver hiba esetén max. 1 órára a felére eshet a teljesítmény)

        - Erőforrás felhasználása (pl. memória, diszkterület, stb.)
        - Skálázhatóság (a hardver növelésével milyen teljesítménynövekedést várunk a rendszertől?)
    - Üzemeltethetőségi követelmények
        - Nyomkövetési és esemény-naplózás
        - Teljesítmény-számlálók
        - Konfigurálhatóság
        - Telepíthetőség
    - Rugalmassági követelmények
        - Mennyire egyszerű a megoldást új funkciókkal kibővíteni.
        - Egy adott funkcionális változtatás a kód egyetlen részének változtatását igényelje, ne kelljen több részhez is hozzányúlni. (refactoring szabályok betartása)
    - Integritásra vonatkozó követelmények
        - Mennyire védett a rendszer adatvesztéstől, vírusfertőzéstől, mennyire biztosított az adatok biztonsága (részben átfedésben van a biztonsági követelményekkel, de más szemszögből nézi azokat).
    - Együttműködésre vonatkozó követelmények
        - Mennyire könnyedén cserél adatot más rendszerekkel, nyíltak-e (dokumentáltak, nyílt szabványokra épülők) az interfészei.
        - Mennyire egyszerű új rendszert integrálni hozzá.
    - Tesztelhetőségi követelmények
        - Mennyire jól lehet a rendszert tesztelni, mennyire lehet ellenőrizni a helyes működést, milyen elhatárolási szintekkel kell a tesztelést támogatni. (Az elhatárolási szintek azt biztosítják, hogy egy esetleges hiba a kód egy adott helyéhez kötődjön és ezt pl. folyamatos kizárásokkal azonosítani lehessen. Pl.: adott modulban, azon belül adott objektum/funkció. Ne legyen olyan, hogy az alkalmazásban minden mindennel összefügg, minden mellékhatásokkal rendelkezhet.)
- Tervezési megkötések
    - A megrendelő műszaki stratégiájából vagy műszaki környezetéből eredő megkötések
    - Kompatibilitási megkötések (pl. mely operációs rendszereken, vagy melyik böngészők melyik verzióin kell futnia)
- Készen felhasználandó komponensek (ha van)
- Betartandó szabványok (jogi, iparági, kommunikációs, stb.)

Vizsgálandók továbbá az alábbi tartalmi szempontok:

- Érthetőség
  - A nyelvezet és a megfogalmazások mind a megrendelő, mind a majdani rendszertervezést végző műszaki csapat számára érthető kell legyen.
  - Mennyire követhetők az egyes igények egymással való alá-fölé ill. mellérendeltségi viszonyai
- Részletezettség
  - A részletezettség szintje megfelelő-e a rendszertervezés megkezdéséhez?
- Teljesség
  - A korábban begyűjtött információk (Megvalósíthatósági elemzés, Projektdefiníció, Interjú és megbeszélés-jegyzőkönyvek, emlékeztetők, Műszaki felmérések jegyzőkönyvei) teljes köre szerepel-e a követelményekben?
- Ellentmondás-mentesség
  - Nincs-e ellentmondás az egyes követelmények között, illetve a specifikáció és a különféle szabványok, szabályozások, illetve más rendszerkomponensek követelményei között?
- Projektcélok támogatása
  - Levezethetők-e a projektdefinícióban megadott célokból a specifikált követelmények?
- Nincsenek rejtett előfeltevések
  - Azonosítva vannak-e azok az előfeltevések, amelyekkel a követelmények specifikációja során éltek?
- Nyitott kérdések lezárva
  - A specifikáció készítése közben felmerült nyitott kérdések mindegyike le lett-e zárva, illetve amelyik nem, arról nyilvánvaló-e, hogy nem akadályozza a rendszertervezés megkezdését?
- Van-e eljárás a követelmények felsorolására? (a majdani követés végett)

Valamint vizsgálható a bizonylatoltság és jóváhagyottság:

- Ha szükséges volt a specifikáció létrehozási folyamatának bizonylatolása, akkor az helyesen történt-e meg? (bizonylatok lehetnek pl. emlékeztetők, jegyzőkönyvek, visszakereshető verzióváltozatok)
- Ha szükséges a specifikáció stakeholder-ek (pl. felhasználók, üzemeltetők, stb.) általi jóváhagyása, akkor az ellenőrizhető módon megtörtént-e? (pl. aláírás; nyilatkozat, stb.)

9.4.2.5. A fogalomtár ellenőrzése

9.4.2.5.1. Az ellenőrzés célja

Mivel a követelményspecifikáció egy kommunikációs eszköz a megrendelő és a műszaki kivitelezést végző szakemberek között, ezért fontos, hogy mindkét fél ugyanúgy értse a specifikációban megjelenő fogalmakat.

A fogalomtár ellenőrzésének célja annak megvizsgálása, hogy az betölti-e ezt a „közös szótár” szerepet.

9.4.2.5.2. Ki végzi az ellenőrzést?

Az ellenőrzést olyan személynek kell végeznie, aki a rendszer üzleti és műszaki fogalmaival is tisztában van (ez lehet több személy is).

9.4.2.5.3. Input

A fogalomtár ellenőrzése az alábbi inputokra épül:

- Követelményspecifikáció
- Fogalomtár

9.4.2.5.4. Minőségellenőrzési feladatok

Az ellenőrzés a rendelkezésre bocsátott dokumentumok vizsgálatával történik.

Az ellenőrzési szempontok:

- Teljesség
  - A specifikációban szereplő összes olyan üzleti és műszaki fogalom, mely nem nyilvánvaló módon közérthető, megjelenik-e a fogalomtárban?
- Érthetőség
  - A fogalomtár legyen világos és érthető.
- Ellentmondás-mentesség
  - A fogalomtárban szereplő definíciók nem mondhatnak ellent a specifikációban történő használatuknak.

9.4.3. Rendszertervezési folyamatok

9.4.3.1. Áttekintés

A rendszertervezés során arra kell megadni a választ, hogy a követelményspecifikációban leírt feladat hogyan valósítható meg.

A különféle módszertanokban a tervezés szakaszaira különféle elnevezések léteznek, de közös vonásuk, hogy egy folyamatosan finomodó tevékenységről szólnak: először egy nagyvonalú koncepció, egy rendszer architektúra alakul ki, majd

egy részletesebb, de logikai szintű modell, végül az implementációs technológiákat is figyelembe vevő, akár megvalósítási részletekig lebontott fizikai terv.

A fizikai terv birtokában már látható, hogy milyen munkadarabokat kell elkészíteni, s ezeket vagy egy egyszerű munkadarab listában, vagy egy hierarchikus struktúrában (WBS - work breakdown structure) fel kell sorolni. Ez alapján lesz mérhető az implementációs feladatok előrehaladása.

Ha a tervezés során olyan kérdések merülnek fel, melyek a projekt további menetét jelentősen befolyásolhatják, de a megválaszolásra nem áll rendelkezésre kellő pontosságú információ, javasolt prototípust, vagy deszkamodell rendszert készíteni, s az ezekkel nyert tapasztalatokat dokumentálni (emlékeztető, jegyzőkönyv).

A prototípus a rendszer egy kiragadott, problematikus részének megvalósítása olyan módon, hogy a rendszer többi része nem, vagy csak jelzésszerűen kerül megvalósításra. Ilyen lehet például egy újszerű felhasználói felület prototípusának megvalósítása azzal a céllal, hogy a felhasználóktól visszajelzéseket lehessen gyűjteni a használhatóságról. A prototípus elsősorban a specifikációs bizonytalanságok csökkentésére használható jól, mert az a felhasználó / megrendelő számára is jól értelmezhető.

A deszkamodell rendszer ehhez hasonló, azzal az apró különbséggel, hogy jellemzően egy műszaki megoldás, vagy technológia alkalmazását próbálja ki, ezért a rendszernek egy szélesebb szeletét kell hozzá megvalósítani, de elnagyoltabb módon, csupán a műszakilag kérdéses részekre koncentrálva. A deszkamodell elsősorban technológiai kockázatok csökkentésére használható hatékonyan.

A prototípus vagy deszkamodell készítése egy kockázatkezelési technika: a lehetséges problémákra minél korábban próbál fényt deríteni, amíg még a megszerzett tapasztalatok alapján van lehetőség alakítani a műszaki megoldáson vagy a projekt menetén.

A rendszerek egyre nagyobb részénél kap különösen nagy hangsúlyt a biztonság kérdése. Az ilyen esetekben javasolt ún. fenyegetettségi modell (Threat Model)[2] készítésével meggyőződni arról, hogy a rendszerterv a különböző fenyegetésekre tartalmaz-e tervezett védekezést. Azonban egy tökéletesen sebezhetetlen rendszer kialakítása olyan költségekkel járhat, vagy a rendszer más jellemzőit, például a használhatóságot olyan módon csorbíthatja, ami már nem éri meg. Tehát nem az a cél, hogy tökéletes biztonságot érjünk el, hanem egy ésszerű szintet, aminek meghatározásához a fenyegetések azonosítására, elemzésére és súlyozására van szükség, ami alapján az ésszerű szintű védekezés definiálható.

[2] A Threat Model-re sajnos nincs egyértelműen elterjedt magyar kifejezés. A szó szerinti fordítás: „fenyegetettségi modell”. A dokumentumban ezt használjuk.

Megjegyzés: a fenyegetettségi modell, vagy Threat Model készítése tehát nem minden esetben követelmény, csak a biztonsági szempontból különösen kényes rendszereknél. A Threat Model maga nem feltétlenül egy önálló dokumentum, lehet a rendszerterv egy fejezete is.

A rendszerterv egy „élő dokumentum”, vagyis a szoftverfejlesztési projekt előrehaladtával módosulhat (pl. új igények felmerülése estén). Fontos, hogy adott mérföldköveknél (minimum a projekt lezárásánál) a rendszerterv aktualizálásra kerüljön, hogy a mindenkori tényleges rendszert tükrözze, ezáltal a későbbi karbantartás és továbbfejlesztés legfontosabb alapanyagát képezi.

9.4.3.2. Célok

A rendszertervezés célja annak dokumentálása, hogy a specifikációban leírtak pontosan HOGYAN legyenek megvalósítva. Ez alapján kerül sor a megvalósításra.

9.4.3.3. Termékek

- Rendszerterv (kötelező)
- Fenyegetettségi modell (threat model) (ajánlott)
- Munkadarab lista vagy WBS (work breakdown structure) (kötelező)
- Prototípus, deszkamodell tesztek eredményei (opcionális)

9.4.3.4. A rendszerterv ellenőrzése

9.4.3.4.1. Az ellenőrzés célja

A cél annak ellenőrzése, hogy a rendszerterv alapján megvalósítható-e egy olyan rendszer, mely a specifikált követelményeket kielégíti. Ellenőrizni kell tehát az érthetőséget, helyességet, teljességet, megvalósíthatóságot és a követelmény specifikációnak való megfelelést.

Megjegyzés: A rendszerterv ellenőrzését segítheti, ha az írásos anyagok tanulmányozása mellett egy terv-értékelő megbeszélésre is sor kerül, ahol lehetőség nyílik egyes kérdések élőben történő megvitatására.

Megjegyzés: Javasolt értékelni a rendszertervben szereplő műszaki megoldások kockázatát is, a projektre érvényes kockázatkezelési folyamatnak megfelelően (ld. 9.7 fejezet).

9.4.3.4.2. Ki végzi az ellenőrzést?

Az ellenőrzést végző személynek magas szintű szoftvertervezési tudással és gyakorlattal kell rendelkeznie, valamint hasznos, ha gyakorlati szinten is ismeri azokat a konkrét technológiákat, melyekre a rendszer épül.

Fontos továbbá, hogy a tervek üzemeltetési szempontból is ellenőrizve legyenek.

9.4.3.4.3. Input

- Követelményspecifikáció (korábbi projektszakasz terméke)
- Fogalomtár (korábbi projektszakasz terméke)
- Rendszerterv
- Prototípus, deszkamodell tesztek eredménye (ha van)

9.4.3.4.4. Minőségellenőrzési feladatok

Mivel a rendszerterv formája és terminológiája a különböző módszertanok esetén meglehetősen eltérő lehet, ezért az alábbiakban azokat a tartalmi szempontokat foglaljuk össze, melyeknek ezektől függetlenül meg kell jelennie valamilyen jól érthető formában.

- Tervezési célok és megkötések ellenőrzési kritériumai (ajánlott)
  - Tervezési célok, megkötések megléte

    A tervezés megkezdése előtt megvoltak-e azok a célok, amelyeket a tervezéssel el kell érni, vagy amelyeket annak figyelembe kellett vennie.
  - Explicit célok

    A tervezés kimondott igények mentén történt-e, vagy kimondatlan feltételezéseket is tartalmaz-e.
  - Összhang a projektcélokkal, megkötésekkel

    A tervezéshez használt célok, megkötések összhangban vannak-e a projekt céljaival, megkötéseivel?
  - Mérhetőség

    A tervezési célok teljesülése eldönthető-e a rendszertervek elkészítése, és a rendszer megvalósítása után? A célokat mérhető kritériumokká kell alakítani.
  - Megkötések jogossága

    A tervezéshez használt megkötések szükségesek-e, vagy szükségtelenek?

Ha ez nem derül ki, akkor a tervek használója kénytelen lenne vakon elfogadni a választott műszaki megoldások helyénvalóságát, mely – a triviálisan egyszerű rendszerektől eltekintve – lehetetlenné teszi a minőségellenőrzést.

- A rendszer logikai felépítésének és működésének ellenőrzési kritériumai (ajánlott)
  - A megoldásválasztás értékelésének kritériumai:
    - Összhang a tervezési célokkal és megkötésekkel

      A javasolt megoldás megfelel-e a tervezési céloknak, teljesíti-e a megfogalmazott megkötéseket? A megoldásnak a célokhoz és megkötésekhez viszonyítva legalább elfogadható (ha nem is optimális) megoldásnak kell lennie.
    - Optimális megoldás

A javasolt megoldás az adott célok és megkötések mellett optimális megoldás-e?

- A választott megoldás értékelésének tartalmi kritériumai:
  - Rendszerkapcsolatok

    Érthető-e belőle, hogy a rendszert kik fogják használni, és mire? (Mik a használat legfontosabb alapesetei?)

    Érthető-e belőle, hogy a rendszernek milyen kapcsolatai vannak más rendszerekkel?
  - Felépítés érthetősége

    Érthető-e belőle a rendszer logikai felépítése (milyen egységekből, modulokból, komponensekből épül fel)? (statikus modell)
  - Működés érthetősége

    Érthető-e belőle a rendszer logikai szintű működése (moduljai, komponensei miként működnek együtt)? (dinamikus modell)
  - Összhang a specifikációval

    A rendszer funkcionalitása megfelel-e a specifikációban elvártaknak, és összhangban van-e a projektcélokkal?

A logikai szintű tervek hiányában rendkívül megnehezül egy bonyolultabb rendszer áttekintése és megértése, mivel hiányzik egy absztrakciós szint a fizikai implementáció felett. Így a tervek helyességének ellenőrzése, illetve a későbbi karbantartás, továbbfejlesztés nehezül meg, vagy lehetetlenül el, ha mindent a fizikai tervekből, vagy még rosszabb esetben a fizikai implementációból kell „kihámozni”.

A logikai nézet kidolgozására sokszor azért nem fektetnek kellő hangsúlyt, mert a fizikai szintű tervek alapján a rendszer gyakorlatilag megvalósítható. Azonban ha a rendszerhez később is hozzá kell nyúlni, akkor világos logikai tervek híján szinte biztos, hogy sérülni fog a rendszer logikai konzisztenciája, amitől megugrik a hibaszám és a karbantartási költségek.

A logikai tervek formai ellenőrzése ki kell, hogy térjen arra, hogy a használt leírás megfelel-e valamely dokumentált és elterjedt rendszertervezési modellnek és jelölésrendszernek. Ezek közül a legelterjedtebb, és a legtöbb tervezőeszköz által is támogatott az UML (Unified Modelling Language – Egységes Modellező Nyelv). Logikai szinten javasolt ennek a logikai csomag és osztály, valamint viselkedési diagramjait használni.

- A rendszer fizikai szintű terveinek ellenőrzési kritériumai (kötelező)
  - Egyértelműség

    A megvalósítást végzők számára egyértelmű-e, hogy az implementáció során milyen fizikai szoftverelemeket kell megvalósítani, és azoknak hogyan kell működnie?
  - Meghatározottság

A fizikai terveknek ki kell térnie a megvalósítandó szoftverelemek felépítésére és kapcsolataira (statikus nézet) és a működésére is (dinamikus nézet). A működést néhány reprezentatív használati eseten illetve funkción keresztül javasolt bemutatni.

- A rendszert megvalósító fizikai processzek és szálak, valamint a köztük folyó kommunikáció módja definiált-e?

  Ennek hiányában a fizikai tervekből kell „kihámozni” azt az aspektust, ami pl. az üzemeltetők számára rendkívül fontos: a rendszer telepítése után konkrétan mik is fognak futni a fizikai gépeken, és milyen kommunikációs módokat fognak használni, aminek fontos kihatása lehet a környező infrastruktúra is (pl. tűzfalak konfigurálása).

- Közérthetőség

  Szabványos jelölési rendszerként használ-e (pl.: az UML fizikai csomag és osztály diagram, komponens diagram, viselkedési diagramok).

- A rendszer telepítési nézetének ellenőrzési kritériumai (kötelező)
  - Egyértelműség

    kiderül-e belőlük a rendszer fizikai elemeinek az azokat futtató hardverekhez történő hozzárendelése,

  - Hatások vizsgálhatósága

    becsülhető-e a rendszer bevezetésének hatása a meglévő infrastruktúrára

Szabványos jelölési rendszerként javasolt az UML modellek használata (deployment modell).

- A rendszer adatkezelési terveinek ellenőrzési kritériumai (ajánlott)
  - Adatok definiáltsága

    Érthető-e a rendszerben közlekedő és tárolt összetett adatszerkezetek felépítése, tartalma (szintaktika és szemantika) (ha van tárolt adat).

  - Adatkapcsolatok definiáltsága

    Definiáltak-e a rendszer összetett adatai közti kapcsolatok.

Ennek hiányában a rendszer adatai (adatbázisa, más rendszerek felé küldött és fogadott üzenetei) „fekete dobozt” jelentenek az üzemeltetők és a későbbi továbbfejlesztést végzők számára.

Ha a rendszernek vannak saját maga által karbantartott összetett adatai, akkor azok logikai adatmodelljének szerepeltetése éppen úgy az érthetőséget és a jövőbeli sikeres továbbfejleszthetőséget szolgálják, ahogy az a logikai modellnél indoklásra került.

Ha a rendszernek van saját adatbázisa, akkor annak fizikai adatmodelljét kötelező szerepeltetni, hogy az adatbázis adminisztrátorok számára referenciát nyújtson a későbbi üzemeltetéshez és karbantartáshoz.

9.4.3.5. Fenyegetettségi modell (threat Model) ellenőrzése

9.4.3.5.1. Az ellenőrzés célja

A cél annak eldöntése, hogy a fenyegetettségi modell a lehetséges sebezhetőségeket teljes körűen feltárja-e, és azokra ésszerű védekezési lépéseket javasol-e.

9.4.3.5.2. Ki végzi az ellenőrzést?

Az ellenőrzést célszerű szoftverbiztonsági szakemberre bízni.

9.4.3.5.3. Input

- Követelmény specifikáció
- Rendszerterv

9.4.3.5.4. Minőségellenőrzési feladatok

Annak vizsgálata, hogy megválaszolja-e az anyag az alábbi kérdéseket?

- Mik azok az értékek a rendszerben, melyeket védeni kell? (leggyakrabban valamilyen információ, de lehet egyéb is, például a működés kieséséből vagy rosszindulatú módosításából származó imázs, mint érték, vesztesége)
- Kik a lehetséges támadók?
- Milyen belépési és kilépési pontok vannak a rendszeren, melyeken keresztül a támadás intézhető?
- Az egyes belépési pontok milyen hozzáférési szinteket definiálnak?
- Mely hozzáférési szintnek milyen értékekhez van hozzáférése?
- Mely hozzáférési pontnál mi lehet a támadó célja?
  - A rendszer azonosítási rendszerének becsapása
  - Adatok manipulálása (tárolt, vagy átvitt adatok)
  - Bizonyítékok eltűntetése
  - Információ megszerzése
  - A szolgáltatás megbénítása
  - Hozzáférési szint megemelése
- Mik a fenyegetések? (Vagyis azok a konkrétan azonosított gyenge pontok, ahol valamilyen műveletsorral a támadó valamilyen értéket kompromittálhat.)
  - Mekkora veszélyt jelent egy adott gyenge pont?
    - Mekkora kár idézhető elő azon keresztül?
    - Mennyire megismételhető a gyenge pont kihasználása?

    - Mekkora a valószínűsége, hogy a támadók megtalálják az adott gyenge pontot?
    - Hány felhasználót érinthet egy támadás?
- A fenyegetés kezelése milyen módokon lehetséges, illetve érdemes?
    - Védelem
    - Észlelés
    - Riasztás

Megjegyzés: A gyakorlatban tökéletes védelem megvalósításáról nemigen beszélhetünk, ezért a háromféle ellenlépés (védelem, észlelés, riasztás) ésszerű kombinációjától várhatjuk a legjobb eredményt.

- A választott kezelési mód milyen mértékben csökkenthető a fenyegetés?

9.4.3.6. A munkadarablista vagy WBS ellenőrzése

9.4.3.6.1. Az ellenőrzés célja

A cél annak ellenőrzése, hogy a munkadarablista a tervekkel összevetve konzisztens és teljes legyen.

A munkadarablista (vagy WBS) alapján lehet mérni a fejlesztési tevékenységek állapotát, előrehaladását. Figyelem: nem azok minőségét, csak számosságát és készültségét.

9.4.3.6.2. Ki végzi az ellenőrzést?

Az ellenőrzést végző személynek ismernie és értenie kell a rendszertervet, célszerűen az a személy, aki a rendszerterv ellenőrzésében is részt vett.

9.4.3.6.3. Input

- Rendszerterv
- Munkadarablista vagy WBS (work breakdown structure)

9.4.3.6.4. Minőségellenőrzési feladatok

A rendszerterv és a munkadarablista összevetésével meg kell állapítani, hogy a rendszertervből következő munkadarabok mindegyike felsorolásra került-e, és csak azok kerültek-e felsorolásra.

Ezen kívül a munkadarablistába ajánlott felvenni az egyéb, nem szoftver jellegű munkadarabokat is, mint pl. az üzemeltetői dokumentáció, felhasználói segédletek, „design” elemek, stb.

Javasolt továbbá felvenni a munkadarablistába a tesztek elvégzéséhez, esetlegesen kifejleszteni a szükséges szoftverkomponenseket (tesztágyak[3], tesztadat-generátorok, stb.), ezeket azonban általában nem a rendszerterv, hanem a tesztelési terv tartalmazza.

9.4.4. Implementációs folyamatok

9.4.4.1. Áttekintés

Az implementáció során készülnek el és kerülnek összeépítésre a rendszert alkotó szoftverelemek.

Az implementáció és a tesztelés folyamata a modern módszertanokban szorosan összefonódik, de azokat az áttekinthetőség miatt külön fejezetekben tárgyaljuk.

Az elkészülő szoftverkomponensek minőségének ellenőrzésére a tesztelésen kívül más technikák is léteznek, pl. a kódanalízis. A szorosan vett tesztelési módszereket a következő fejezetben tárgyaljuk, az egyéb módszereket pedig ebben a fejezetben.

A fejlesztés során vannak bizonyos módszertani elvárások, melyek vizsgálatára a minőségellenőrzésnek szintén ki kell terjednie. A néhány legalapvetőbb ezek közül a fejlesztési szabályok lefektetése, a forráskezelési folyamat (source control) és a forrásból a telepíthető terméket előállító „build” folyamat (melynek nincs elterjedt magyar megnevezése).

Az implementációs tevékenységekbe tartozik a rendszer különböző dokumentációinak elkészítése, elsősorban az üzemeltetői dokumentáció (mely tartalmazza a rendszer telepítésének leírását is), és a felhasználói dokumentációk, segédletek, oktatóanyagok (ha szükségesek).

Az implementációs tevékenységek befejeztével a rendszertervet szükség esetén aktualizálni kell, hogy az mindig a tényleges rendszert tükrözze, s így lehetővé tegye a rendszer későbbi karbantartását, továbbfejlesztését.

9.4.4.2. Célok

Az implementáció célja a rendszertervben meghatározott szoftverkomponensek kifejlesztése, és egyéb kapcsolódó termékek, dokumentációk elkészítése.

9.4.4.3. Termékek

- Fejlesztési szabályok (ajánlott)
- Build szabályzat és/vagy automatizmus (kötelező)

---

[3] Önmagukban nem működőképes modulok, alrendszerek teszteléséhez használt szoftverkomponens, mely szimulálja a teszteléshez szükséges, de még el nem készült, nem elérhető, vagy az adott tesztben nem szereplő egyéb modulokat, vagy akár teljes külső rendszereket.

- Forráskód (kötelező)
- Telepíthető és futtatható rendszer (kötelező)
- Üzemeltetési dokumentáció (kötelező)
- Felhasználói segédletek (opcionális)
- Aktualizált rendszerterv (kötelező)

9.4.4.4. Az implementációs folyamatok ellenőrzése

9.4.4.4.1. Az ellenőrzés célja

A cél annak felmérése, hogy a fejlesztési projektben a minőségbiztosítási szempontból fontos szabályok, gyakorlatok betartásra kerültek-e.

9.4.4.4.2. Ki végzi az ellenőrzést?

Az ellenőrzést a szoftverfejlesztési módszertanokban, folyamatokban jártas személy tudja elvégezni.

9.4.4.4.3. Input

- Fejlesztési szabályok
- Build szabályzat és/vagy automatizmus

9.4.4.4.4. Minőségellenőrzési feladatok

Az ellenőrzés során az alábbi kritériumok teljesülését kell vizsgálni.

- Le vannak-e fektetve írásban a következő fejlesztési szabályok?
  - Standard fejlesztői környezet, pontos szoftververziókkal (kötelező)
  - A források kezelésének módja (kötelező)
  - A build folyamat működtetésének módja (kötelező)
  - A hibák rögzítésének és kezelésének módja (kötelező)
  - Kódolási, elnevezési, kommentezési konvenciók, nyelvhasználat előírása (ajánlott)
- Forráskezelési folyamat
  - Fel van-e állítva forráskezelést támogató rendszer (source control)? (kötelező)
  - A source control alatt levő adatok rendszeres mentésre kerülnek-e? (kötelező)
  - Definiált-e, hogy milyen állapotú forrást szabad berakni a source control rendszer alá? (pl. csak a fejlesztői gépen sikeresen lefordított és letesztelt forrást) (ajánlott)
  - Megkövetelt-e a check-in megjegyzések használata? (ajánlott)

- Egyéb speciális tudnivalók (pl. kódelágaztatás mikor történik, ki melyik kódágban dolgozik, mik a kódágak közötti átintegrálás szabályai) (opcionális)

- Build folyamat
  - Verziók azonosíthatósága
    Minden build egyedi azonosítót kapjon. (kötelező)
  - Reprodukálhatóság
    Ugyanabból a forrásból ugyanaz a build ismételten előállítható legyen. (kötelező)
    A reprodukálhatóságot legjobban az automatizált build rendszer támogatja. (ajánlott)
  - Korábbi build verziók megőrzése
    A folyamat biztosítsa a korábbi build verziók mentését. (kötelező)

9.4.4.5. A forráskód kézi ellenőrzése

9.4.4.5.1. Az ellenőrzés célja

- A kódolási és kommentezési konvenciók betartásának ellenőrzése
- A kód érthetőségének ellenőrzése.
  - Csak az a rendszer lesz később karbantartható, melynek kódja jól olvasható és érthető.
- A kód helyességének ellenőrzése.
  - Az emberi ellenőrzés olyan logikai hibákra deríthet fényt, melyeket a tesztelés csak nehezen, vagy áttételes módon tudhat kiszűrni, vagyis a kódhibákat gyökerüknél ragadja meg, és nem tünetileg kezeli őket.

9.4.4.5.2. Ki végzi az ellenőrzést?

A code review elvégzéséhez nagy fejlesztői gyakorlattal rendelkező, és a rendszer terveit jól ismerő személyre van szükség.

> Megjegyzés: A kódellenőrzés egy igen hatékony módja a „pair programming”, vagyis ha a forrás írója és az ellenőrzést végző személy közösen odaülnek a forráskód elé, a vitás kérdéseket azonnal feloldják, és a talált hibákat azonnal kijavítják – ennek azonban a hátránya az, hogy nincs dokumentált nyoma.

9.4.4.5.3. Input

- Fejlesztési szabályok
- Forrás kód
- Rendszerterv (korábbi projektszakasz terméke)

9.4.4.5.4. Minőségellenőrzési feladatok

Hatékony forráskód ellenőrzésének elvégzéséhez igen nagy gyakorlatra van szükség. A kódvizsgálat során ellenőrizendő összes dolgot itt lehetetlen felsorolni, csupán a legfontosabb aspektusokat tudjuk kiemelni.

A kódellenőrzés legfontosabb kritériumai:

- Érthetőség
  A kódellenőrzés alapszabálya: ha nem érthető, akkor nem jó!
- Kódkommentezés
  - A kommentezés követi a fejlesztési szabályokban lefektetett formát és elveket?
  - A megjegyzések a mögöttes „miért”-ekre fókuszálnak? (nyilvánvaló információk helyett)
- Naplózás (trace)
  - Minden szükséges ponton történik nyomkövetési naplózási hívás? (hibák, figyelmeztetések, elindul valamilyen feldolgozás, befejeződik a feldolgozás, külső rendszer hívása, stb.)
  - Kívülről kapott paraméterértékek naplózásra kerülnek?
- Hibakezelés
  - Mindenhol történik-e hibakelezés, ahol nem kívánjuk a hiba felterjedését?
  - Ha egy külső paramétert nem kap meg a rendszer, akkor beépített alapértelmezett értéket csak figyelmeztetéssel használ a rendszer?
  - Hiba elkapása esetén nem marad el a cleanup logika?
- Bemenő paraméterek ellenőrzése
  - Külső interfészeken minden paraméter ellenőrzése, belső interfészeken csak amit az ésszerűség megkíván.
- Konfigurálhatóság
  - Világosan elválnak az üzemeltetési környezettől függő paraméterek (pl. gépnevek) a rendszer viselkedését befolyásoló paraméterektől?
  - Telepítési/futtatási környezettől függő paramétereknek (pl. gépnevek, IP címek, portok) TILOS alapértelmezett értéket fixen beépíteni. Ha hiányzik a paraméter, akkor inkább álljon le hibával a rendszer, minthogy pl. csendben rossz adatbázisba kezdjen írni.
  - A programkódba TILOS konstansokat írni (kivétel a triviális konstansok: 0, 1, null). Minden konstanst (szövegek, darabszámok, kódok, stb.) definiáljunk konstansként, a forrásban jól elkülöníthető módon.
- „Rendszer minimalitás” elve

- Ne kódoljunk le olyasmit, ami készen is megvan és használható (pl. platform szolgáltatás)
- Soha ne másolgassunk le kódrészleteket több helyre. Oldjuk meg függvényhívással vagy más, az adott technológia által biztosított módon (makró, include, stb.)

- „Funkció lokalitás” elve
  - Egy függvény egyetlen, jól definiált feladatot hajtson végre, mellékhatások nélkül.
- Kódolvashatóság
  - Az adott projektben lefektetett fejlesztési szabályok betartása, általánosságban pedig tagolt, „szellős” kód, a bekezdések megfelelő használatával.
  - Trükkös kódot írni TILOS, mivel az a továbbfejlesztések során időzített bombát jelent.
  - A kódbeli változók, paraméterek, stb. elnevezése kövesse a lefektetett fejlesztési szabályokat, általánosságban. Értelmes, az adott elem céljára utaló nevek használata.

9.4.4.6. A forráskód automatizált ellenőrzése

9.4.4.6.1. Az ellenőrzés célja

A cél minden olyan potenciális probléma azonosítása, mely automatizált eszközökkel lehetséges. A lehetőségek technológiánként igen változóak, de tipikusan az alábbiak.

- Kódolási stílus ellenőrzése
  - Léteznek olyan eszközök, melyek ellenőrizni tudják a helyes bekezdések/igazítások használatát, a fejkommentek meglétét és formáját, sőt akár az angol nyelvű objektumnevek helyes betűzését is.
- Statikus kódanalízis
  - Az ilyen eszközök a kód futtatása nélkül keresnek lehetséges hibákat: pl. inicializálatlan változó használatát, hibakezelési kód hiányát, potenciálisan null pointer használatát, stb.
- Dinamikus kódanalízis
  - Az ilyen eszközök a kód futtatása során keresnek potenciális problémákat, pl. biztonsági réseket, buffer túlcsordulást, stb.

9.4.4.6.2. Ki végzi az ellenőrzést?

Az ellenőrzés automatizáltan történik, a fejlesztési folyamatban előre meghatározott időpontokban, vagy időközökben (pl. minden sikeres új build-re).

Az automatikus ellenőrzés által generált jelentés vagy naplófájl kiértékelését, és ez alapján a módosítási/javítási feladatok kiadását a fejlesztőknek tipikusan a fejlesztői csapat vezetője végzi.

9.4.4.6.3. Input

- Forráskód
- Telepíthető és futtatható rendszer

9.4.4.6.4. Minőségellenőrzési feladatok

A feladat az automatikus kódanalizáló eszközök meghatározása, használatuk beillesztése a fejlesztési folyamatba, az eszközök által adott eredmények folyamatos kiértékelése, és ezek alapján a fejlesztői feladatok kiadása.

9.4.4.7. Az üzemeltetési dokumentáció ellenőrzése

9.4.4.7.1. Az ellenőrzés célja

A cél az üzemeltetési dokumentáció teljességének és helyességének ellenőrzése.

9.4.4.7.2. Ki végzi az ellenőrzést?

- Üzemeltető: teljesség és érthetőség ellenőrzése.
- Fejlesztő: teljesség ellenőrzése.
- Tesztelő: helyesség ellenőrzése.

9.4.4.7.3. Input

- Rendszerterv (korábbi projektszakasz terméke)
- Telepíthető és futtatható rendszer
- Üzemeltetési dokumentáció

9.4.4.7.4. Minőségellenőrzési feladatok

Az ellenőrzést érdemes többfelé bontani: teljesség, érthetőség, helyesség ellenőrzése.

- Teljesség ellenőrzése – ellenőrizzük az alábbi tartalmi elemek meglétét.
  - A rendszer bemutatása
    - Architektúra áttekintése
    - Felhasznált technológiák azonosítása
    - Működés szemléltetése néhány reprezentatív folyamattal
    - Teljesítmény-jellemzők

    - Rendelkezésre állási jellemzők
    - Biztonsági jellemzők
- Telepítési leírás
    - Külön telepítendő összetevők azonosítása
    - Lehetséges kiépítések magyarázata
    - Előfeltételek megadása és lépésenkénti telepítési leírás minden külön telepítendő összetevőhöz
    - A rendszer kezdeti konfigurálása
    - A rendszer inicializálása
    - A telepítés és konfigurálás helyességének ellenőrzése
- Elindítás, leállítás, újraindítás leírása
    - A rendszer külön-külön leállítható/elindítható szolgáltatásainak azonosítása
    - Az egyes szolgáltatások leállításának, elindításának módja
    - Mikor és mely szolgáltatás újraindítására lehet szükség?
- Mentés/visszaállítás leírása
    - Perzisztens, nem reprodukálható adatok tárolásának azonosítása a rendszerben
    - Mentésre javasolt adatok körének meghatározása, javasolt mentési ütemezés
- Konfigurálási lehetőségek leírása
    - A rendszer összes konfigurációs tárolójának azonosítása (paraméterfájlok, adatbázis-bejegyzések, stb.)
    - Az egyes konfigurációs tárolók teljes referenciája: paraméter neve, szerepe, lehetséges értéke, alapértelmezett értéke (ha van)
- Biztonság
    - Biztonsági konfigurációs pontok azonosítása (hol történik a rendszerben azonosítás, feljogosítás, titkosítás, auditnaplózás, stb.)
    - Ajánlott biztonsági konfiguráció meghatározása
- Adminisztrátori feladatok leírása
    - Az adminisztrátori feladatok felsorolása (pl. felhasználó felvétele, jogosultságok megadása, stb.)
    - Minden feladathoz a végrehajtás módjának leírása.
- Monitorozás leírása

        - Monitorozási lehetőségek felsorolása (eseménynaplózás, teljesítményszámlálók, stb.)
        - Monitorozásra javasolt paraméterek, küszöbértékek és teendők megadása
    - Diagnosztika és hibaelhárítás leírása
        - Diagnosztikai eszközök felsorolása (egyedi eszközök esetén használatuk leírása is)
        - Javasolt hibakeresési folyamat, hibakeresési technikák
        - Tudásbázis (ismert problémák és megoldásuk)
- Érthetőség ellenőrzése
    - Az érthetőség ellenőrzéséhez célszerű a dokumentum célközönségének képviselőjét, tehát egy vagy több üzemeltetőt felkérni a dokumentum elolvasására és észrevételezésére.
- Helyesség ellenőrzése
    - A helyesség ellenőrzése a dokumentumban leírt különféle eljárások lepróbálását jelenti (telepítés, mentés, visszaállítás, stb.), melyet a következő fejezetben, a tesztelési folyamatoknál tárgyalunk.

9.4.5. Tesztelési folyamatok

9.4.5.1. Áttekintés

A tesztelést szokás azonosítani a szoftver minőségbiztosításával, holott – ahogy a korábbi fejezetekből látható – a kettő nem ugyanaz, de valóban a tesztelés, illetve a tesztelés során feltárt hibák a leglátványosabb megnyilvánulásai a nem-megfelelőségnek.

A tesztelési folyamatok a rendszertervezéssel körülbelül egy időben kezdődnek, a tesztek megtervezésével. A tesztek megtervezése során azonosításra kerülnek, hogy milyen aspektusokat, milyen módon, a projekt mely fázisában kívánunk tesztelni.

A tesztelési terv elkészítése után kerülnek definiálásra a konkrét tesztesetek. Egy teszteset akkor definiált, ha meg van adva, hogy milyen kiindulási állapot és bemenet mellett, milyen lépések végrehajtásával milyen végállapotot és kimenetet várunk. Ennek alapján a teszt reprodukálható, ami az egyik legfontosabb elvárás a tesztekkel szemben.

Megjegyzés: A definiáltság és reprodukálhatóság különösen jól megvalósítható a tesztek automatizálásával. Mivel a teszteket jellemzően többször is le kell futtatni, ezért az automatizálás sokszor jelentős munkamegtakarítást jelenthet, kivéve azokat az eseteket, amikor az automatizálás annyira bonyolult lenne, hogy jobban megéri a kézi végrehajtás.

A tesztek végrehajtását dokumentálni kell: mikor lett végrehajtva, a termék mely verzióján, milyen eredménnyel.

Megjegyzés: A tesztek fajtái különböző szempontok szerint is osztályozhatók. Az egyik osztályozás azt veszi alapul, hogy a tesztesetet definiáló számára a rendszer belső felépítése és működése ismert-e (ez a white-box teszt), vagy nem ismert (ez a black-box teszt). A unit tesztek például, melyek a rendszernek egy alsó szintű építőelemét (függvényt, osztályt) tesztelnek, tipikusan white-box tesztek. A biztonsági réseket felderíteni próbáló behatolási tesztek tipikusan black-box tesztek.

Egy másik osztályozás szerint beszélhetünk funkcionális tesztekről, melyek azt vizsgálják, hogy a rendszer valóban képes-e elvégezni a követelményekben szereplő funkciókat, illetve minőségi, vagy nem-funkcionális tesztekről, melyek egyéb jellemzőket tesznek próbára: megbízhatóságot, sebességet, skálázódást, stb.

Szokás regressziós tesztekről is beszélni: ez arra utal, mikor egyszer már sikeresen végrehajtott teszteket, vagy feltárt, de kijavított hibákat újra letesztelünk azért, hogy meggyőződjünk arról, hogy az újonnan végzett fejlesztések, hibajavítások nem rontottak-e el olyan dolgot, ami korábban már helyesen működött. Regressziós teszt nem csak a funkcionalitást vizsgálhatja, hanem bármely más típusú tesztet (pl. egy sebességtesztet) is ha fennáll a veszélye, hogy bizonyítani kívánjuk, hogy egy korábban már tesztelt jellemző továbbra is megfelel a követelményeknek – ez is regressziós teszt. Regressziós teszteket tipikusan üzletileg kritikus rendszerek módosításai esetén szokás végezni.

Az automatizált tesztek egyik hatalmas előnye, hogy az egyszer megírt teszt később akárhányszor újra lefuttatható regressziós tesztelési céllal.

A tesztek során talált minden nemű problémát rögzíteni kell a hibanyilvántartásban. A hibák száma, és annak időbeli alakulása fontos metrikák, melyek segítenek a projekt állapotának és a rendszer minőségének megítélésében.

Megjegyzés: A hibanyilvántartás nem csak a tesztelés során talált nem-megfelelőségek nyilvántartására alkalmazható, hanem bármely minőségellenőrzési tevékenység során talált probléma (pl. dokumentumhiányosságok) rögzítésére is. De a jelen fejezetben csak a tesztelési folyamatokra koncentrálunk.

Elvileg a követelményspecifikációban lévő minden követelmény megvalósulását tesztelni kell, de ez a gyakorlatban már egyszerű esetekben sem teljesíthető maradéktalanul. A gyakorlatban engedményt kell tenni: nem a teljes, hanem az ésszerű szintű leteszteltségre érdemes törekedni. Hogy mi az ésszerű szint, azt jellemzően a teszteléssel fel nem tárt hibák okozta kockázatok súlya, a tesztelés költségei, és a rendelkezésre álló idő összevetése adja meg.

Mindezeken túlmenően szükség lehet a tesztelés hatékonyságának mérésére is. Egy lehetséges technika, ha a rendszerbe bizonyos számú hibát szándékosan beépítünk, majd azt vizsgáljuk, hogy a tesztelés ezekből hányat talált meg.[4] Kevésbé egzakt, de szintén használható módszer annak ellenőrzésére, hogy a tesztelések valóban megtörténnek-e, ha a felhasználási statisztikákat naplózzuk és elemezzük.

[4] A módszer nem teljes mértékben korrekt, ehhez kellene, hogy a hibák egymástól függetlenek és véletlenszerűek legyenek.

Az automatizált unittesztek hatékonyságát az ún. lefedettségi (code coverage) méréssel lehet megállapítani. Ennek során a lefedettséget mérő eszköz megállapítja, hogy a tesztek futásakor a forrás mely sorai futottak le, s ezek a teljes kódbázis mekkora hányadát teszik ki. A unit tesztekre előzetesen kitűzhető egy pontos szám, hogy a forrássorok hány százalékát kívánjuk lefedni velük. A cél jellemzően nem a 100%, hanem inkább az 50-80% szokott lenni.

9.4.5.2. Célok

A tesztelési folyamatok célja annak ellenőrzése, hogy a rendszer viselkedése megfelel-e a specifikációban leírt követelményeknek, illetve a józan elvárásoknak.

9.4.5.3. Termékek

- Tesztelési terv (kötelező)
- Tesztesetek terve (kötelező)
- Automatizált tesztek forrása és végrehajtható változata (opcionális)
- Teszteredmények jegyzőkönyvei (kötelező)
- Kód lefedettség (code coverage) mérési eredmények (opcionális)

9.4.5.4. A tesztelési terv ellenőrzése

9.4.5.4.1. Az ellenőrzés célja

A cél annak ellenőrzése, hogy a tesztelési terv a szükséges tesztfajtákat és tesztkörnyezeteket teljes körűen definiálja-e.

9.4.5.4.2. Input

- Követelményspecifikáció (korábbi projektszakasz terméke)
- Rendszerterv (korábbi projektszakasz terméke)
- Tesztelési terv

9.4.5.4.3. Minőségellenőrzési feladatok

A tesztelésnek ki kell terjednie minden lényeges követelmény ellenőrzésére.

A követelményspecifikáció alapján ellenőrizendő, hogy az adott típusú követelményekhez a megfelelő tesztfajták megjelennek-e a tesztelési tervben.

Ellenőrizendő az is, hogy minden teszthez mérhető célok vannak-e definiálva.

- Funkcionális követelményekhez
  - Unittesztek
    - A szoftver kisebb építőegységeit teszteli (metódusok, osztályok).
  - Integrációs tesztek

        - A szoftver moduljainak, alrendszereinek összeépítését, valamint külső rendszerekkel való összeépítés során teszteli a működést.
    - Felhasználói tesztek
        - Felhasználói felületen keresztül végrehajtott tesztek.
- Teljesítménykövetelményekhez
    - Kapacitástesztek
        - A rendszert a követelményekben előírt kapacitásig terheljük.
    - Sebességtesztek
        - A feldolgozási és válaszidők mérése.
    - Skálázódási tesztek
        - A rendszer kapacitásának és/vagy sebességének mérése egyre nagyobb kapacitású hardveren, közben a kapacitás- és/vagy sebességnövekedés mérése.
    - A tesztekhez szükséges műterhelés kiváltásához sokszor tesztalkalmazást kell készíteni.
- Használhatósági követelményekhez
    - Használhatósági tesztek, pl. az alkalmazást nem ismerő felhasználó esetén adott feladat elvégzési idejének mérése.
- Megbízhatósági követelményekhez
    - Stressz tesztek
        - Szélsőséges körülmények mesterséges előidézése: a követelményekben meghatározottnál nagyobb mértékű műterhelés, szűkös erőforrások szimulálása, stb.
    - Redundancia tesztek
        - Redundáns kiépítésű rendszer esetén (pl. fürtözött kiszolgáló) az egyik rendszerkomponens kiiktatása közben az alkalmazás helyes működésének ellenőrzése.
    - „Monkey” tesztek
        - A felhasználói felület szándékoltan szakszerűtlen, véletlenszerű használata során annak ellenőrzése, hogy a rendszer kellően „védi-e magát” az ilyenektől. (Ez a tesztfajta elüt az összes többitől, ugyanis nem követelmény a tervezettség és megismételhetőség.)
- Biztonsági követelmények esetén
    - A funkcionális tesztekhez hasonló módszerek a biztonsági funkciók ellenőrzéséhez
    - Betörési tesztek

    - Lehet white-box, vagy black-box, attól függően, hogy a betöréssel kísérletező kapott-e információkat a rendszer belső felépítéséről és működéséről, vagy sem.
- Telepíthetőségi követelményeknél
  - Telepítési tesztek

További ellenőrizendő tartalmi elem, hogy a tervben a különböző tesztekhez szükséges környezetek is definiálva vannak-e.

- Lehetséges jó megoldás például, ha minden tesztfajtához le van írva az ahhoz szükséges tesztkörnyezet, azonban ez a gyakorlatban túl terjengős lehet.
- Egy tömörebb leírási mód, ha néhány tesztkörnyezet kerül definiálásra, s az egyes tesztfajtáknál az szerepel, hogy mely környezeten kell végrehajtani. (Példa: egy kis kapacitású környezet a unittesztek elvégzéséhez, egy nagyobb kapacitású, de nem redundáns kiépítésű környezet a felhasználói tesztekhez, és egy, a leendő éles rendszerrel megegyező kiépítésű tesztrendszer a stress és performancia tesztek elvégzéséhez.)
- Általánosságban elmondható, hogy egy tesztkörnyezettől azt várjuk el, hogy az adott tesztelni kívánt aspektust tekintve megegyezzen az élessel. Ha funkcionális teszteket végzünk, akkor megfelelhet sokkal kisebb teljesítményű környezet is, de például fontos, hogy a releváns szoftververziók megegyezzenek. Ellenben egy sebességtesztnél elvárás, hogy a tesztkörnyezet teljesítménye megegyezzen az élessel, vagy legalábbis a kettő között nyilvánvaló kapcsolat legyen, ami lehetővé teszi az extrapolálást.

Megjegyzés: A fejlesztés folyamata során előfordulhatnak olyan hibák is, melyek egészen drasztikus formában mutatkoznak, vagyis egész modulok, vagy akár az egész rendszer lehet működésképtelen, esetleg nem is telepíthető. Annak érdekében, hogy az ilyen esetek minél előbb kiderüljenek, és ne a tesztelésre szánt időt vigyék el, szokás BVT-t (Build Verification Test – Build helyességellenőrzési teszt) készíteni[5]. Ez néhány egyszerű, automatizált tesztet jelent, melyet minden build-re lefuttatnak, hogy lássák, nincs vele alapvető probléma, érdemes továbbadni „komoly” tesztelésre.

9.4.5.5. A tesztesetek ellenőrzése

9.4.5.5.1. Az ellenőrzés célja

A cél annak ellenőrzése, hogy a konkrét tesztesetek jól definiáltak és teljeskörűek-e.

[5] Ezt a tesztet nevezik „Smoke Test”-nek is, mely onnan ered, hogy állítólag régen az elektronikus eszközök gyártásánál a kipróbálás első lépése az áram alá helyezés volt – ha füst szállt fel, akkor az eszköz kipróbálását nem volt érdemes folytatni.

9.4.5.5.2. Input

- Követelményspecifikáció (korábbi projektszakasz terméke)
- Rendszerterv (korábbi projektszakasz terméke)
- Tesztelési terv
- Tesztesetek terve

9.4.5.5.3. Minőségellenőrzési feladatok

- Tesztesetek teljeskörűségének ellenőrzése
  - A tesztelési tervet használva megállapítható, hogy vannak-e olyan tesztfajták, melyek tesztesetei teljesen hiányoznak.
  - Egy tesztfajtán belül (pl. felhasználói tesztek) a tesztesetek teljeskörűségét vagy hiányosságát csak a követelményspecifikáció és/vagy a rendszerterv alapján tudjuk megállapítani. (Például a felhasználói tesztekből esetlegesen hiányzó tesztesetek csak a felhasználói felületre vonatkozó funkcionális követelmények végigjárása során azonosíthatók.)
- Tesztesetek jól definiáltságának ellenőrzése
  - A tesztesetekre meg kell vizsgálni, hogy a jól definiáltsághoz szükséges minden jellemzőjük le van-e írva:
    - kiindulási állapot és bemenet,
    - milyen lépéseket kell végrehajtani,
    - milyen végállapotot és kimenetet várunk.
- Tesztadatok ésszerűségének ellenőrzése
  - A teszteseteknél vizsgálni kell, hogy a megadott, vagy meghivatkozott tesztadatok a célnak megfelelnek-e.
    - Fontos, hogy a tesztadatok jól mintázzák a valós adatokat – legjobb, ha valós adatokból kerülnek ki.
    - Ha a valós adatok használata titokvédelmi problémákat vet fel, akkor megoldás lehet a valós adathalmaz felismerhetetlenné tétele olyan módon, ami az adatok jellegzetességeit (pl. ismétlődések, jellegzetes értékek) nem módosítja lényegesen. Erre általános algoritmus nem adható, egyedileg lehet ilyet meghatározni annak függvényében, hogy az adatok mely aspektusa védendő.
    - Ha valós tesztadat nem elérhető, akkor a tesztadatok meghatározására, vagy generálására kell nagy gondot fordítani.

9.4.5.6. A tesztelési folyamatok ellenőrzése

9.4.5.6.1. Az ellenőrzés célja

A cél annak eldöntése, hogy a projektben megvalósított tesztelési folyamatok megfelelnek-e az iparági legjobb gyakorlatoknak.

9.4.5.6.2. Input

E feladatok inputját nem csak dokumentumok képezik, hanem a ténylegesen működő tesztelési folyamat maga, pl. annak ellenőrzése, hogy a hibanyilvántartó rendszerben valóban minden szükséges adat rögzítésre kerül-e.

- A tesztelési folyamatok tényleges gyakorlata
- Tesztesetek terve
- Teszteredmények jegyzőkönyvei
- Automatizált tesztek (ha vannak)

9.4.5.6.3. Minőségellenőrzési feladatok

Az ellenőrzés során meg kell győződni az alábbi szempontok megvalósításáról.

- Hiba nyilvántartási folyamat és eszköz ellenőrzése (kötelező)
  - Ellenőrizni kell, hogy a hibák nyilvántartása során megvalósulnak-e az alábbi, minimálisan elvárt jellemzők.
    - A talált hibák regisztrálásra kerülnek-e. (Erre célszerű egy erre kialakított eszközt használni.)
    - A rögzített hibákat adott időnként osztályoznia kell egy erre jogosult személynek, vagy bizottságnak: el kell dönteni, hogy valóban hibáról van-e szó, javítandó-e a hiba a jelen fázisban, illetve mi a sürgőssége és fontossága.
    - A javítandónak ítélt hibákat ki kell szignálni a javítást végzőkhöz.
    - A javítás után a hiba állapotát javítottra kell állítani (nem lezártra!).
    - A javított hibát a tesztelőnek ellenőriznie kell, hogy valóban elhárult-e, s csak ezután zárhatja le azt.
  - Ellenőrizni kell, hogy a hibanyilvántartásban rögzítésre kerülnek-e legalább az alábbi adatok:
    - A hiba felvételekor
      - A hiba azonosítója
      - A hiba megnevezése
      - A hiba állapota (nyitott)
      - A hiba reprodukálásának módja

        - A termék verziója, melyben a hibát találták
        - A hiba rögzítésének dátuma
        - A hibát rögzítő személy neve
    - A hiba elbírálásakor
        - A hiba súlyossága
        - A hiba prioritása
        - A javítást végző személy neve
    - A hiba javításakor
        - A javítás leírása
        - A javítása dátuma
        - A hiba állapota (javítva)
    - A javítás ellenőrzésekor
        - A javítás ellenőrzését végző személy neve
        - A javítás ellenőrzésének dátuma
        - A hiba állapota (az ellenőrzés eredménye alapján: nyitott vagy lezárt).
- Tesztesetek és teszteredmények nyilvántartásának ellenőrzése (kötelező)
    - Ellenőrizni kell, hogy létezik-e nyilvántartás az összes tesztesetről. A tesztesetek nyilvántartásánál minimálisan elvárt adatok
        - Teszteset azonosítója
        - Kiindulási állapot és bemenő adatok azonosítása
        - Végrehajtandó lépesek azonosítása
        - Elvárt eredményállapot és kimenő adatok azonosítása
    - Ellenőrizni kell, hogy létezik-e nyilvántartás a tesztek eredményéről. A teszteredmények nyilvántartásánál minimálisan elvárt adatok
        - Teszteset azonosítója
        - Végrehajtás időpontja
        - A teszt eredménye (megfelelt, vagy ha nem felelt meg, akkor a rögzített hibák azonosítói)
        - A tesztet végrehajtó személy neve
        - A termék verziója, melyen a tesztet végrehajtották
- Tesztek automatizálásának ellenőrzése (javasolt)
    - Nem követelmény, de javasolt automatizálni azokat a teszteket, melyeknél ez ésszerű ráfordítással megtehető.

- Ha vannak automatizált tesztek, akkor ellenőrizni kell, hogy azok kielégítik-e a tesztek tervezettségének követelményeit, vagyis egy automatizált tesztre igaz-e az alábbi?
  - Reprodukálható módon létrehozza-e a szükséges kiindulási állapotot?
  - Reprodukálható módon létrehozza, vagy megszerzi-e a bemenő adatokat?
  - Reprodukálható módon végrehajtja-e a teszteset által megkívánt lépéseket?
  - Ellenőrzi-e a kapott kimenő adatot egy helyesnek tekintett referenciaadathoz viszonyítva?
  - Ellenőrzi-e a rendszer végállapotát egy helyesnek tekintett referenciaállapothoz képest?

9.4.5.7. Hibastatisztikák és trendek elemzése

9.4.5.7.1. Az ellenőrzés célja

A hibastatisztikákból és trendekből következtetések levonása a rendszer készültségére és a tesztelések hatékonyságára vonatkozóan.

9.4.5.7.2. Input

- A hibanyilvántartó rendszerből nyert statisztikák

9.4.5.7.3. Minőségellenőrzési feladatok

A hibastatisztika-trendek elemzése során vizsgálandó kritériumok.

- Van-e hibastatisztika

  Ha nincsenek hibastatisztikák, az egyértelmű probléma.

- Hatékony-e a tesztelés

  Ha az egyes tesztek (pl. felhasználói tesztek) megkezdése után semmiféle változás nem történik a hibatrendekben, az arra utalhat, hogy a tesztelés nem folyik hatékonyan.

- Elszabaduló hibaszám

  Ha a megtalált hibák száma jóval nagyobb ütemben emelkedik, mint a javított hibák száma, az arra utalhat, hogy érdemes felfüggeszteni az új funkciók fejlesztését, hogy a fejlesztőknek lehetősége legyen stabilizálni az aktuális kódbázist, mielőtt a túl sok nyitott hiba további hibákat eredményez.

- Konvergáló hibaszám

  Ha a megoldott hibák száma meghaladja az újonnan talált hibákét, és ez nem csak pillanatnyi állapot, hanem trend (Bug Convergence), akkor a stabilizálás

hatékonyan halad, és viszonylag pontos becslés is mondható a kibocsátási mérföldkő elérésére.

- Ha a nyitott hibák száma kitartóan csökken, majd eléri a nullát (ZBB – Zero Bug Bounce), az még nem feltétlenül jelenti azt, hogy a rendszer hibamentes, de jelzi, hogy a fejlesztők utolérték a tesztelőket a hibajavításban. Ha a tesztelések folytatásával a hibaszám csak kis mértékben emelkedik, majd újra csökkenni kezd, akkor valószínűsíthető, hogy a termék hamarosan kiadható állapotba kerül.

9.4.6. Kibocsátási folyamatok

9.4.6.1. Áttekintés

Egy rendszer kibocsátásának bonyolultsága igen tág határok között változhat. Egyszerű esetben a kibocsátás egy egyetlen telephelyet érintő éles telepítésből állhat, bonyolultabb esetben viszont együtt járhat egy hosszas oktatási és pilot tevékenységgel, a sok telephelyes és időben is elhúzódó bonyolult telepítési folyamattal, esetleg egy régi rendszer kiváltása esetén adatmigrációval, ami lehet egy automatizált adatkonverzió, de lehet egy hosszas manuális adatfeltöltési procedúra is.

Bonyolultabb esetekben mindenképp javasolt kibocsátási tervet készíteni, ami pontosan kidolgozva tartalmazza a tennivalókat, s így azok nagy kockázatú esetben előzetesen ki is próbálhatók.

9.4.6.2. Célok

A kibocsátási folyamatok célja a kifejlesztett rendszer éles üzembe helyezése, beleértve a sikeres éles működéshez szükséges feltételek megteremtését is.

9.4.6.3. Termékek

- Kibocsátási terv (opcionális)

9.4.6.4. A kibocsátási terv ellenőrzése

9.4.6.4.1. Az ellenőrzés célja

A cél annak biztosítása, hogy a kibocsátás zökkenőmentesen történjen.

9.4.6.4.2. Input

- Követelmény specifikáció (korábbi projektszakasz terméke)
- Rendszerterv (korábbi projektszakasz terméke)
- Kibocsátási terv

9.4.6.4.3. Minőségellenőrzési feladatok

Ha a kibocsátási folyamat bonyolultsága indokolta a kibocsátási terv elkészítését, akkor szükséges annak minőségellenőrzése is.

A kibocsátási terven az alábbi vizsgálatok javasoltak:

- Az oktatási terv ellenőrzésével kapcsolatos kritériumok
  - Megfelelő mennyiségű idő van-e fenntartva az oktatásokra az élesbe menetel előtt?
  - Adottak-e az infrastrukturális követelmények az oktatások lebonyolításához?
- Üzletileg kritikus rendszerek kibocsátásával kapcsolatos kritériumok
  - A kockázat csökkentése érdekében van-e egy pilot időszak, ami a felhasználók egy kisebb csoportját érinti, esetleg csak szűkített funkcionalitással, sokszor földrajzilag is korlátozott módon (pl. csak egy telephelyen).
  - A pilot időszak alatt a felhasználók és az üzemeltetők számára szerepel-e kiemelt támogatás nyújtása, rövid válaszidejű rendelkezésre állással.
  - Van-e a kibocsátás teljes folyamatára előre kidolgozott forgatókönyv, beleértve a visszaállási folyamatot is, ha előre nem látott probléma merül fel.
  - A kibocsátási forgatókönyv előre el lett-e próbálva, beleértve a visszaállási folyamatot is.
- Technológiai újdonságokat tartalmazó rendszerek kibocsátásának ellenőrzési kritériumai
  - Az üzemeltetők oktatásra kerülnek-e és/vagy az üzemeltetők számára az élesbe menetelt követően kiemelt támogatás nyújtására sor kerül-e.
- Adatmigrációs feladatokat is igénylő kibocsátás ellenőrzésének kritériumai
  - Az adatmigráció ismételhető-e, tesztelhet-e, van-e automatizmus a migrációra?
  - Ha a migráció nem oldható meg egy lépésben, akkor kezelve van-e az átmeneti időszak?
- Nagyszámú kliens vagy szerver telepítését igénylő kibocsátás ellenőrzésének kritériumai
  - Rendelkezésre áll-e, és kipróbált-e a telepítési technológia?
  - Kellően produktív, automatizált-e a telepítési technológia?

9.5. Projektzárás és értékelés

9.5.1. Áttekintés és célok

Az értékelő és záró szakasz célja a projektindítás tárgyát képező termék értékelése, átadása, illetve a megbízó által történő átvétele. Ezzel azonban még nem záródik le a tennivalók sora. Itt történik meg a projektmenedzser/tagok, projektfolyamat értékelése is. Összegyűjtésre kerülnek a projekt során sikeresnek/kudarcnak vélt teendők és a megfelelő konklúziók levonása után megtörténik ezek archiválása.

A projekt értékelés és a zárás folyamata a következő elvégzendő feladatokat tartalmazza:

- projektfolyamat értékelése:
  értékelnünk kell mindent, ami a termék előállításának folyamatával kapcsolatos. Ilyen
  - az együttműködés értékelése (a projekt belső szervezetének működése kommunikációs, illetve munka-kulturális szempontból),
  - a munkafolyamatok értékelése (leglényegesebb és legkritikusabb feladatok kiválasztása),
  - a menedzsmentfolyamatok (menedzsmenttevékenységek értékelése, tervezés, szervezés, irányítás, adminisztrációs folyamatok, kockázatmenedzsment) értékelése.
- projektzárás:
  a projektzárás során történik meg a projektszervezet feloszlatása, erőforrások re-integrációja, utókalkuláció, kimutatások elkészítése (idő, költségek), projektzáró okirat jóváhagyása.
- projekt tudásbázisának archiválása:
  a projekt folyamán számos olyan tudás, tapasztalat (szakmai, adminisztratív) halmozódott fel, ami egy későbbi projektben nagy segítség lehet, ezért javasolt a tapasztalatokat a szervezet számára újrafelhasználhatóvá tenni. Ennek egyik módja az írásos anyagok archiválása jól visszakereshető formában, másik módja egy, a projektvezető által tartott workshop. Ezen workshop keretében előadja tapasztalatait, észrevételeit, ezzel elősegítve a többiek fejlődését és reakcióképességét az esetlegesen felmerülő későbbi projektbeli problémák kezelésére.

Megjegyzés: A projekt értékelésének egy lehetséges módja az ún. Post Mortem megbeszélés. Erre javasolt meghívni a projektvezetőn kívül minden projektszerep egy-egy képviselőjét, hiszen más-más szempontból tudják értékelni a projektet. Javasolt a projekten kívülálló *moderátort* megbízni a megbeszélés levezetésével, hogy a projektrésztvevők a tartalmi kérdésekre tudjanak koncentrálni, ugyanakkor legyen valaki, aki segít a formális keretek betartásában (pl. mindenki kapjon szót). A *résztvevők száma ne haladja meg a 10-et;* nagyobb projekt esetén inkább külön-külön értékelő megbeszélés javasolt az egyes csapatoknak, majd egy „fő” megbeszélés, ahol az egyes csapatokat már csak egy-egy ember képviseli. A *megbeszélés napirendje az alábbi*.

• A megbeszélés céljának ismertetése: a projektből származó mindenféle tanulság összegyűjtése, jó és rossz egyaránt. A jót ugyanis továbbvinni, a rosszat pedig elkerülni szeretnénk a jövőben.

• A megbeszélés attitűdjének megadása: a megbeszélés célja semmiképpen nem a hibáztatás, felelősségre vonás, hanem a tanulás.

• A projekt rövid összefoglalása, felidézése, például a projektvezető által (mettől meddig tartott, mi volt a cél, fő fázisok, eredmények).

• Ezek után ötletroham (brainstorming) formában következhet a gondolatok összegyűjtése, mind a jó, mind a rossz tapasztalatait mindenki bemondja, ezeket a moderátor összegyűjti, de egyelőre a csapat nem értékel és nem priorizál, csak összegyűjt.

• Ha sok megvitatandó dolog merült fel, akkor priorizálni kell, majd a rendelkezésre álló idő függvényében csak a legfontosabb néhányat megvitatni.

• Ezután következik a felhozott problémák, vagy jó tapasztalatok átbeszélése, kötetlen formában, a moderátor irányítása alatt.

• A megbeszélésről egy összefoglaló dokumentumot kell készíteni, ami a jó és rossz tapasztalatokon kívül konkrét teendőket is tartalmazhat.

Ha az értékelő beszélgetés nehezen indul be, a moderátor segítsen néhány *gondolatébresztő* értékelési szemponttal, például az alábbiakkal:

• Tiszták voltak-e a célok?

• Mennyire volt teljes a tervezés, mielőtt megkezdődött a megvalósítás?

• Hogyan lehetne javítani a tervezésen?

• Elegendő erőforrás volt?

• Realisztikus volt-e az ütemterv?

• Melyik feladatnak volt kisebb/nagyobb az időigénye a tervezettnél?

• Változások kezelése jól ment-e?

• Szoftverfejlesztési módszertan használata tisztázott volt?

• A projekt külső függőségei hatékonyan voltak kezelve?

• A kockázatok hatékonyan voltak kezelése?

• Elegendő tesztelés volt?

• Volt olyan probléma, ami nem derült ki a tesztelés során? Hogyan lehetne javítani ezen?

• Hogyan sikerült a problémák kommunikálása és kezelése?

• Jó volt-e a kommunikáció a projekttagok között?

• Milyen volt a kommunikáció más, projekten kívüli szereplőkkel?

• Tisztázott volt, hogy kinek mi a feladata és a felelőssége a projektcsapatban?

• A megrendelő elégedett-e az elkészült termékkel? Miért/miért nem?

• A felhasználók elégedettek-e az elkészült termékkel? Miért/miért nem?

• Az üzemeltetők elégedettek-e az elkészült termékkel? Miért/miért nem?

• Mennyire volt hatékony a projekt eszköztámogatottsága? (hibanyilvántartás, forrás kezelés, stb.)

Ilyen megbeszélést hosszabb projekt esetén nem csak a végén, hanem *minden nagyobb fázis után is javasolt* megtartani.

9.5.2. Termékek

A szakasz során elkészítik a *projektzáró dokumentumot*. A projektzáró okirat, a projekt célja és megvalósulása mellett hivatkozásokat tartalmaz az értékelés eredményeire vonatkozóan. Ez nem csak a termék értékelését jelenti, hanem beletartozik maga a projektfolyamat értékelése is, valamint az átadás/átvétel dokumentációja és a lényeges, projekttel kapcsolatos dokumentációk gyűjteménye.

9.5.3. A projektzárás és ellenőrzés minőségellenőrzése

9.5.3.1. Az ellenőrzés célja

A minőségellenőrzés célja, hogy ellenőrizze, hogy a fenti tevékenységeket megfelelő minőségben hajtották-e végre, a projektben nem maradt-e lezáratlan tevékenység, vagy kötelezettség.

9.5.3.2. Ki végzi az ellenőrzést?

Javasolt, hogy az ellenőrzést olyan minőségbiztosító munkatárs végezze, aki a projekt eddigi fázisaiban részt vett.

9.5.3.3. Input

- A projektzárás és értékelés tevékenységeiből származó dokumentumok, illetve arra vonatkozó személyes interjúk.
- A projekt során keletkező termék átvétele és értékelése során keletkezett dokumentumok.

9.5.3.4. Minőségellenőrzési feladatok

- A projekt folyamatának értékelése során ellenőrizni kell, hogy az olyan módon történt-e meg, hogy a szervezet és a résztvevők az értékelésből a szükséges információkat megkapták, megértették és profitálhattak belőle.
- A projektzárásnál ellenőrizni kell, hogy a zárási teendőket elvégezték-e, a projektzáró dokumentumban leírtak érthetőek és megfelelnek a valóságnak.
- A projekt tudásbázisának archiválása során ellenőrizni kell, hogy az archiválás megtörtént-e, olyan anyagok archiválása történt-e, melyek fontosak a tudás megőrzése és újrafelhasználása szempontjából, és olyan módon történt, mely lehetővé teszi a könnyű visszakereshetőséget.

9.6. Projektmenedzsment

A projektmenedzsment tevékenysége a projekt egészén végigvonul, így annak minőségbiztosítása nem szakaszokhoz kötődik, hanem rendszeres tevékenységet jelent a projektmenedzserrel szorosan együttműködve.

A minőségbiztosító a projektmenedzser által végzett tevékenységeket véleményezi, ellenőrzi, szakmai tanácsokat ad. Biztosítja, hogy a projekt tervezetten és kontrolláltan, dokumentumokkal alátámasztva haladjon előre, az ehhez szükséges tevékenységek elvégzését és minőségét ellenőrzi.

A minőségbiztosító részt vesz a rendszeres projektmegbeszéléseken, ellenőrzi azok meglétét, minőségi és hasznos lefolytatását.

A minőségbiztosító a minőségbiztosítási tervben meghatározott időközönként minőségbiztosítási jelentéseket ad a projektről a projektszponzor számára (a szakaszok zárásakor készülő minőségbiztosítási jelentések mellett).

9.7. Kockázatmenedzsment

9.7.1. Áttekintés

A projektek sikerét számos dolog fenyegeti. A kockázat minden projekt természetes velejárója, a kockázatok kezelése ennek megfelelően alapvető feladat.

A kockázatokkal kapcsolatban az emberek alapvetően ellenszenvvel viselkednek, tartanak, félnek tőlük. Éppen ezért fontos felhívnunk az alábbi hozzáállásra a figyelmet:

- *A kockázat minden projektnek szerves része.* A kockázat minden projektben jelen van, de nem jelent biztos veszteséget, csak annak lehetőségét. E veszteség jelentheti a termék minőségének romlását, a költségek megemelkedését, a határidők elmulasztását, vagy akár a projekt teljes sikertelenségét is.
- *A kockázat önmagában nem „jó” és nem „rossz”.* A kockázat valami, amit el kell kerülni, különösen, miután tisztában vagyunk vele, hogy minden projektben jelen van. Egy felismert kockázat felér egy felismert lehetőséggel, ezért önmagában nem pozitív és nem negatív dolog.
- *A kockázatot nem félni, hanem kezelni kell.* A sikeres csapatok felismerik és csökkentik a projektben rejlő bizonytalanságot, és megelőző jelleggel kezelik a felismert kockázatokat.

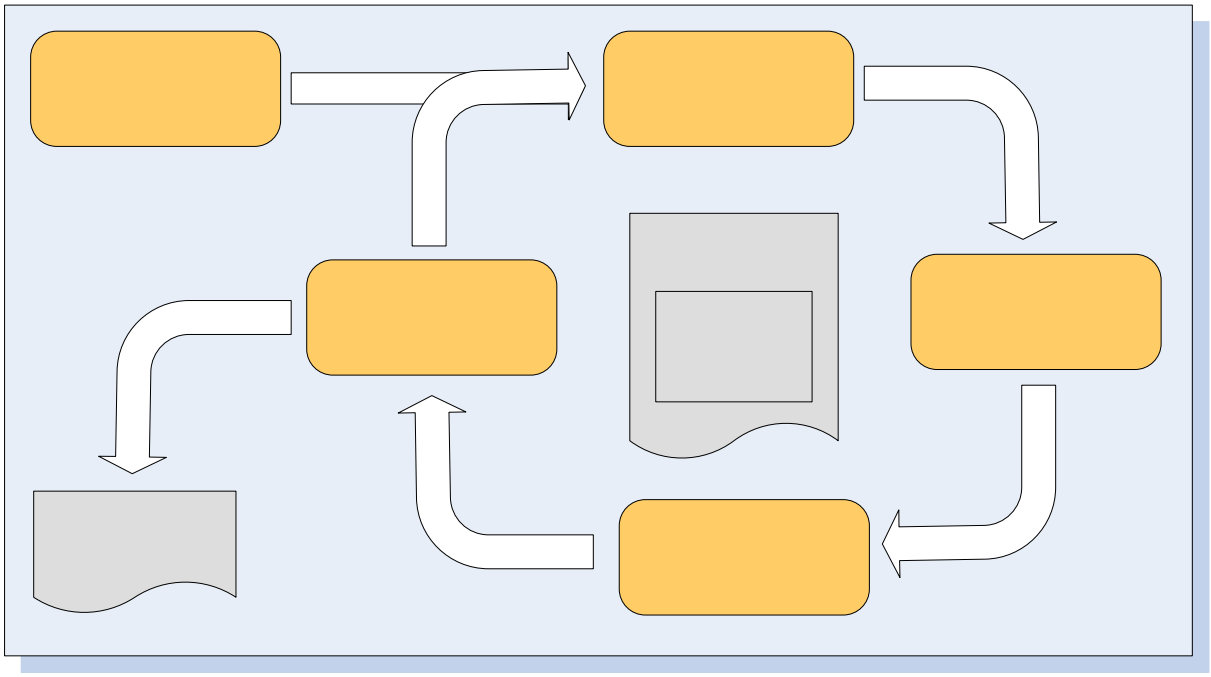
A kockázatkezelés a következő elveken kell, hogy alapuljon:

- *A kockázatok értékelését a projekt teljes időtartama alatt folytatni kell.* A sikeres kockázatkezelés nem korlátozódhat a veszélyforrások projekt kezdetén történő azonosítására. Kockázatok a projekt folyamán is adódhatnak, meglévők bekövetkezési valószínűsége változhat, értékelésüket tehát a projekt időtartama alatt folytatni kell.

- *Alkalmazzunk kockázatalapú döntési módszereket*. A projekt feladatainak elvégzési sorrendjénél figyelembe kell venni az azokban rejlő kockázatokat. A nagyobb kockázatú feladatokat kell először elvégezni. Így elkerülhető, hogy egy nagy kockázatú feladat a projekt 90%-os készültségekor megsemmisítse addigi munkánkat.
- *Alakítsunk ki egyszerű formalizmusokat.* A siker érdekében előre rögzített kockázatkezelési eljárást kell követnünk, de ennek egyszerűnek, a csapat által könnyen használhatónak kell lennie.
- *Vonjuk be valamennyi kulcs-résztvevőt és -folyamatot.* A siker érdekében a kockázatokat a projekt valamennyi területén azonosítanunk kell, különben jó eséllyel nem fedünk fel fontos kockázatokat.
- *A kockázatok azonosítását kezeljük pozitívan*. A kockázatkezelés hatékonysága érdekében a csapattagoknak hajlandónak kell lenniük a kockázatok felfedésére anélkül, hogy kritikától vagy büntetéstől kéne tartaniuk.

A kockázatok „fontosságát” bekövetkezési valószínűségük és hatásuk szorzataként mérjük. Ez alapján a legfontosabbakat kezeljük akár csökkentésük, áthárításuk (pl. biztosítás), akár bizonyos funkcionalitás elhagyása által.

A kockázatkezelés folyamata több lépésből áll, amelyet az alábbi ábra szemléltet:



Ábra 9-3. A kockázatkezelés folyamata

- *A kockázat azonosítása*. A kockázat felszínre hozása annak érdekében, hogy a csapat kezelhesse azt, mielőtt a projektre hatással lehetne.
- *A kockázat értékelése*. A kockázat átalakítása a csapat által felhasználható *információvá*, melyet döntések meghozatala során felhasználhatunk.
- *Tervezés*. Tervek kidolgozása a döntések támogatása és a szükséges tevékenységek elvégzése érdekében. A tervek vonatkozhatnak a bekövetkezési

valószínűségek csökkentésére (megelőzés), és a hatások csökkentésére (kármentés).

- *Követés*. A kockázat és a kezelésére hozott intézkedések állapotának követése.
- *Ellenőrzés*. A kockázatkezelés napi projektmenedzsment-feladattá tétele biztosítja annak elfogadottságát.

A fenti folyamat eredménye egy „élő”, az idő haladtával folyamatosan aktualizált dokumentum.

A kockázatkezelés persze erőforrás-igényes, ezért általában csak az aktuálisan legfontosabb kockázatok (mondjuk a 10 legfontosabb) kezelésével foglalkozunk.

A kockázatok egy része az idő előre haladtával elveszti jelentőségét. Ezeket nyugalmazhatjuk, azaz törlés nélkül elkülönítjük az aktív kockázatainktól.

A kockázatok leírásával kapcsolatban a legfontosabb információk:

- Azonosító
- Megnevezés
- Leírás

Megjegyzés: A kockázatok sokszor ködös megfogalmazásban bukkannak fel, ahol nehéz eldönteni, hogy pontosan mi is a veszély, és mi annak hatása, illetve, hogy nem mostunk-e össze több különböző kockázatot egybe (pl. „ennek az új technológiának a használata egy kockázat”).

Ezért javasolt a kockázatok leírását egységes „ha – akkor” formára hozni – ezzel rákényszerítjük a csapatot a kockázat átgondolására, és külön-külön azonosítani tudjuk a feltételt (ez hordozza a bizonytalanságot), és a következményt (ez hordozza a problémát). (Például: „ha az új technológiát a fejlesztők menet közben sajátítják el, akkor a határidők csúszni fognak”).

A jól megfogalmazott kockázatra könnyebb kezelést találni. (A példánál maradva: az új technológiában igazoltan jártas szakemberek szerződtetése; vagy a fejlesztők tanfolyamra küldése még a projekt előtt; vagy tartalékidő beszúrása a projekttervbe tanulási feladatokra.)

- Fontossága
  - Bekövetkezésének valószínűsége
  - Bekövetkezés hatásának mértéke

Megjegyzés: Ha egy kockázat bekövetkezési valószínűsége 100%, akkor az már nem kockázat, hanem probléma, s így kezelni is máshogy kell, mint a kockázatokat: priorizálással, akcióterv meghatározásával és feladatok kiosztásával.

Megjegyzés: A valószínűség és a hatás becslésére, beskálázására számos bonyolult módszer létezik, de meglepően jó eredményt hoz a sokkal egyszerűbb, 1-5 skálázás intuitív használata.

- Mérséklésének terve (ha készül)
- Kármentesítés terve (ha készül)

Megjegyzés: A kármentesítési tervhez rendelhetünk őrfeltételt is, melyik segít azonosítani azt az eseményt, vagy időpontot, amikortól a kármentesítést meg kell kezdeni.

Példa:

- kockázat: „ha a megrendelt szerver nem érkezik meg időben, akkor az éles indulás csúszik”

- kármentesítési terv: „egy tesztszerver ideiglenes beállítása éles kiszolgálónak”

- őrfeltétel: „ha az éles indulás előtt 10 munkanappal nem érkezett még meg az új szerver, akkor meg kell kezdeni a teszt szerver fizikai átmozgatását és újratelepítését”

- Nyugdíjazás oka (ha értelmezett)

9.7.2. Célok

A kockázatmenedzsment célja a projekt céljainak elérését veszélyeztető kitettségek, fenyegetések, kockázatok azonosítása és kezelése.

9.7.3. Termékek

- Kockázatkezelési és osztályozási stratégia
- Kockázatkezelési dokumentum

9.7.4. A kockázatkezelési stratégia ellenőrzése

9.7.4.1. Az ellenőrzés célja

Az ellenőrzés célja, hogy megbizonyosodjunk arról, hogy a projekt rendelkezik olyan eljárással, mely képes a projekt sikerességét veszélyeztető kockázatok időben történő azonosítására és kezelésére.

9.7.4.2. Ki végzi az ellenőrzést?

Az ellenőrzést bármely független minőségbiztosító végezheti (a megrendelő is)

9.7.4.3. Input

- Kockázatkezelési stratégia

9.7.4.4. Minőségellenőrzési feladatok

Ellenőrizendő, hogy a kockázatkezelési stratégia megfelel-e az alábbi kritériumoknak:

- Definiáltság
  A kockázatkezelési folyamat egyértelműen definiálja-e az alábbiakat:
  - Kockázatkezelési folyamat lépései
  - A bekövetkezési valószínűség becslésének módszerét
  - A hatás becslésének módszerét

- Mely kockázati súly felett kötelező a kockázatok kezelése (tervezés, akciók, …)
- Eszkalációs szabályokat a kockázatokkal kapcsolatban.

- Egyszerűség
  Elég egyszerű-e a választott módszer ahhoz, hogy a résztvevők azt megértsék, és napi szinten alkalmazzák?
- Teljesség
  Tartalmaz-e a stratégia minden olyan elemet, amelyekkel a kockázatkezelés elérheti célját?

9.7.5. A kockázatkezelés ellenőrzése

9.7.5.1. Az ellenőrzés célja

Az ellenőrzés célja annak biztosítása, hogy a projekt hatékonyan és eredményesen azonosítsa és kezelje a projekt sikerességét veszélyeztető kockázatokat.

9.7.5.2. Ki végzi az ellenőrzést?

Az ellenőrzést formai szempontból bármely független minőségbiztosító végezheti (a megrendelő is).

Az ellenőrzést tartalmi szempontból a projektmenedzsmentben és a kockázatkezelésben járatos szakembernek kell végeznie.

9.7.5.3. Input

- Kockázatkezelési és osztályozási stratégia
- Kockázatkezelési dokumentum
- Státuszjelentések
- Változáskezdeményezések és döntési lapok
- Projekttervek, munkadarablisták

9.7.5.4. Minőségellenőrzési feladatok

Ellenőrizendő, hogy a kockázatkezelés megfelel-e az alábbi kritériumoknak:

- Van kockázatmenedzsment
  - Létezik kockázatkezelési stratégia
  - A projekt rendszeresen azonosítja a kockázatokat, és dönt azok kezeléséről
- Folyamatosság
  - Megjelenik-e a kockázatmenedzsment napi szinten?
  - Része-e a státuszjelentéseknek?

- Téma-e projekt megbeszéléseken?

- Kiterjedtség
  Bevonásra kerül-e minden olyan érintett, aki tudhat kockázatok létezéséről, és segíthet azok azonosításában, kezelésében?
- Helyesség
  - A kockázatok és az azokhoz kötődő tevékenységek helyesen vannak súlyozva, kezelve?
  - A 100% valószínűségű „kockázatok” már problémaként vannak kezelve?
  - Döntések meghozatalánál figyelembe veszik-e a kockázatokat?
- Teljesség
  A kockázatmenedzsment kitér valamennyi kockázati területre, amely a projektet fenyegetheti?
  - Követelménykockázatok (Nem definiáltak, hiányosak, ellentmondóak, …)
  - Technológiai kockázatok. (Új technológiák, megbízhatatlan technológiák, egymásra hatások, …)
  - Kompetenciakockázatok (Mennyiség, képzettség, …)
  - Politikai kockázatok (Vezetői támogatás hiánya, ellenérdekelt stakeholderek, …)
  - Szervezeti kockázatok (Szervezeti bizonytalanság a megrendelői oldalon, hiányzó szponzor, …)
  - Ütemezés-kockázatok (Irreális becslések, kritikus utak, bemenetektől függés, …)
  - Pénzügyi kockázatok (Pénzügyi források bizonytalansága, …)

Személyi kockázatok (Tapasztalatlan projektvezető, személyi ellentétek a projektcsapatban, nagy fluktuáció, …)

9.8. Konfigurációmenedzsment

9.8.1. Áttekintés

A konfigurációmenedzsment kifejezést különféle értelemben szokták használni. A továbbiakban azt a folyamatot és eszközkészletet értjük alatta, mely:

- leírja a projekt által előállítandó termék szerkezetét, azonosítja annak elemeit,
- megadja a projekt termékeinek, munkadarabjainak, bemeneteinek és a projekt követéséhez, a minőségbiztosítás és a projektmenedzsment bizonylatainak illetve ezek különböző verzióinak verziózási és tárolási szabályait
- megadja, hogy mely elemek mely verziói alkotnak egy konzisztens egészet,
- biztosítja a korábbi verziók visszakereshetőségét.

A termékszerkezet elemei azok, melyek önállóan azonosíthatók és verziózhatók. Tehát termékelem pl. egy forrásfájl, de önmagában nem termékelem egy kódsor.

A konfigurációmenedzsment alá vont termékstruktúra elemei például az alábbiak:

- specifikációk,
- tervek,
- a forrásfájlok,
- a termék lefordított, telepíthető változatának komponensei,
- a termék telepítési környezetére vonatkozó megkötések és információk,
- a termék előállításához használt eszközök,
- a projekt által előállított dokumentumok és egyéb információk,
- a projekt által készen kapott dokumentumok és egyéb információk,
- a hibanyilvántartás,
- a tesztesetek és tesztjegyzőkönyvek nyilvántartása,
- stb.

A konfigurációmenedzsment eszközei például:

- source control rendszer,
- dokumentumtár(ak),
- build rendszer (ha van),
- hibanyilvántartás,
- teszteset és tesztjegyzőkönyv-nyilvántartások,
- minőségellenőrzési jegyzőkönyvek nyilvántartásai,
- a szoftver- és hardverkörnyezeteket leíró nyilvántartások,
- átadás-átvételi jegyzőkönyvek

Megjegyzés: Egy lehetséges konfigurációkezelési gyakorlat például a source control rendszer használatának kiterjesztése a forrásfájlokról az összes dokumentumra (a projekt által előállított és készen kapott dokumentumokra is), beleértve a különféle nyilvántartási célokat szolgáló dokumentumokat is (tesztesetek listája, hardverkörnyezetek listája, sőt akár a hibanyilvántartás is, ha az egy dokumentumban tárolt).

Eszerint a megközelítés szerint minden fájl, dokumentum és egyéb információ, aminek a tárolására nincs külön cél-eszköz (pl. hibanyilvántartó rendszer) a source control rendszerbe kerül, így biztosított minden információ egységes megtalálása és elérhetősége, verziózása, illetve a teljes információ-csomag egységes mentése, szükség esetén visszaállítása. Ezen kívül még azon fájlok, dokumentumok nem kerülnek a source control rendszerbe, melyek ott már meglévő elemekből reprodukálható módon előállíthatók. Így tehát az egyes build verziók sem kerülnek vissza a source control-ba – azoknak külön tárolója van, pl. egy fájlkiszolgáló, vagy CD archívum.

A konfigurációmenedzsment fontos feladata még annak nyilvántartása, hogy hányféle informatikai rendszerrel dolgozik a projekt (fejlesztői, teszt és éles környezetek), melyiknek milyen a konfigurációja (hardver és szoftver összetevők és aktuális verziószámok), és melyik környezetet pontosan milyen célra lehet használni, valamint hogy milyen folyamat szerint és kritériumrendszer mellett lehet a terméket, vagy annak egy összetevőjét egy másik környezetre átvinni.

Megjegyzés: Nem a szorosan vett szoftverfejlesztési projekthez tartozik, hanem annak utóéletéhez, ezért csak megjegyzésként említjük, hogy az élesben üzemelő rendszerekhez érdemes „javítócsomag-nyilvántartást" vezetni, mely tartalmazza, hogy a legelső élesbe menetele óta a rendszerre mikor és milyen javítócsomagok lettek telepítve, milyen problémát javítottak és mely komponenseket módosítottak (korábbi és aktuális verziószám).

9.8.2. Célok

A konfigurációmenedzsment célja a terméket alkotó elemek összegyűjtése, azonosítása, visszakereshetővé tétele, valamint az elemek különféle verzióiból a termék egy-egy konzisztens verziójának kijelölése.

9.8.3. Termékek

- Konfigurációmenedzsment terve(kötelező)

Megjegyzés: A konfigurációmenedzsment terve nem feltétlenül kell, hogy önálló dokumentum legyen; lehet pl. a projektterv fejezete is.

- Terméktár (repository) (kötelező)

Megjegyzés: A terméktár nem feltétlenül egyetlen tároló, hanem a projekttermékek és dokumentumok tárolóinak összessége.

9.8.4. A konfigurációmenedzsment ellenőrzése

9.8.4.1. Az ellenőrzés célja

A cél annak ellenőrzése, hogy követhető-e, milyen terméket állít elő a projekt, az milyen elemekből áll, a termék melyik verzióját az elemek melyik verziói alkotják, és melyik környezetben a termék melyik verziója található meg. Ezen felül annak ellenőrzése, hogy a termékek egyértelműen azonosíthatóak-e, illetve egyértelmű-e, hogy egy új terméket hogy kell tárolni, hol kell keresni.

9.8.4.2. Input

- Projektdefiníció
- Követelményspecifikáció
- Fogalomtár
- Rendszerterv

- Munkadarablista
- Fejlesztési szabályok
- Build szabályzat és/vagy automatizmus
- Üzemeltetési dokumentáció
- Tesztelési terv
- Teszteredmények jegyzőkönyvei

9.8.4.3. Minőségellenőrzési feladatok

Az ellenőrzés során vizsgálni kell egyrészt azt, hogy van-e végiggondolt konfigurációmenedzsment a projektben, másrészt érdemes-e megvizsgálni az alábbi konkrét aspektusokat külön-külön is.

Megjegyzés: A konfigurációkezelési folyamatok a gyakorlatban összefonódnak a többi szoftverfejlesztési folyamattal (pl. az implementálás során a fejlesztő folyamatosan dolgozik a forráskezelő rendszerrel; a tesztelő a hibanyilvántartással, stb.), ezért az alábbiakban visszaköszönhetnek olyan kritériumok, melyeket valamely más szoftverfejlesztési folyamatnál már tárgyaltunk.

- Áttekinthetőségi kritériumok
  - Világosan kiderül-e, hogy mi az a termék (mik azok a termékek), melyet a projekt előállít?
  - Világosan kiderül-e, hogy a projekttermékeket milyen elemek építik fel, és azok hol tárolódnak?
  - Világosan kiderül-e, hogy milyen inputból milyen folyamat során állnak elő a projekttermékek elemei?
  - Azonosíthatóak-e a projekthez tartozó elemek?
  - Azonosíthatóak-e a projekten kívülről kapott termékek?
  - Azonosíthatóak-e a projekt átadott termékei?

Megjegyzés: Hasznos gyakorlat összeállítani egy „újonc dokumentumot”, mely rövid, vázlatos formában tartalmazza azt, hogy hol és milyen jellegű információkat kell átnéznie annak, aki újonnan kapcsolódik a projektbe. Egy ilyen „projekt-tartalomjegyzék” kitűnően rávilágít a konfigurációmenedzsment esetleges hiányosságaira.

- Forráskezelés ellenőrzésének kritériumai
  - Minden forrásfájl a source control rendszerben tárolódik-e?
  - Definiált-e az a logika, ami alapján a forrásfában az egyes elemek megtalálhatók, illetve van-e olyan „tartalomjegyzék”, mely a forrásfában való eligazodást segíti?
  - Definiált-e a forrás fájlok verziózásának módja? (A forráskezelő rendszer ezt jellemzően magától biztosítja.)

- Definiáltak-e a forráskezelés szabályai? Ki, mikor, mit kölcsönözhet ki magához, és milyen feltételek mellett teheti vissza?

- Külső komponensek kezelésének ellenőrzési kritériumai
  - Definiált-e, hogy a külső komponensek (értve ez alatt bármilyen, a fejlesztendő rendszerhez készen felhasznált összetevőt) mely verzióját kell használni, hol kell tárolni, és kell-e azokat projekten belüli azonosítóval vagy verziószámmal ellátni?
- Build előállításának és tárolásának ellenőrzési kritériumai
  - Definiált-e az a folyamat, ahogy a forrásból build, azaz futtatható komponensek és telepítőkészlet készül?
  - Definiált-e, hogy hogyan kell a build-eket verziózni és archiválni?
- Dokumentumok előállításának és tárolásának ellenőrzési kritériumai
  - Definiált-e, hogy az előállításra kerülő dokumentumokat hogyan kell elnevezni, tárolni, azonosító adatokkal ellátni, verziózni?
  - Definiált-e, hogy a projekten kívülről származó dokumentumok kapnak-e a projekten belüli azonosító, vagy verziószámot?
  - Definiált-e, hogy mely dokumentumot hol kell tárolni? Egyszerűen kikövetkeztethető-e az egyes dokumentumok tárolási helye, vagy van-e olyan „tartalomjegyzék”, mely alapján egy adott dokumentum fellelhető?
- Informatikai környezetek leírásának ellenőrzési kritériumai
  - Definiált-e, hogy hányféle informatikai környezetet használ a projekt? (fejlesztői környezetek, tesztelési és köztes környezetek, éles környezet)
  - Definiált-e az a folyamat és kritériumrendszer, mely alapján a termék egy adott környezetből egy másikba átvihető?
  - Van-e olyan nyilvántartás, melyből egyértelmű, hogy mely környezetben a termék mely verziója található meg?
- Hibanyilvántartás és a hibák kezelésének ellenőrzési kritériumai
  - Definiált-e a hiba-nyilvántartási folyamat?
  - Biztosított-e az összes hibabejegyzés azonosíthatósága és visszakereshetősége?
  - Biztosított-e, hogy minden hibáról eldönthető legyen, hogy a termék mely verziójában találták, és melyikben került javításra?
  - Biztosított-e, hogy egy hiba csak akkor kerülhessen lezárásra, ha az nem csak kijavítva lett, de leellenőrizve is, hogy valóban megszűnt-e?
- Tesztesetek, tesztadatok és teszteredmények nyilvántartásának ellenőrzési kritériumai
  - Van-e nyilvántartás a tesztesetekről?

- Eldönthető-e egyértelműen minden tesztesetről, hogy az volt-e már végrehajtva, mikor, ki által, milyen adatokkal, a termék mely verzióján, és milyen eredménnyel?

9.9. Változásmenedzsment

9.9.1. Áttekintés

Legkésőbb a projektdefiníció készítésének elkezdésekor tisztázásra kerülnek a projekttől elvárt eredmények, a projekt költségvonzatai, a folyamatok elvégzésére allokált idők és erőforrások. A fenti peremfeltételek általában a felek közötti szerződésben is szerepelnek.

A projekt tipikusan egy folyamatosan változó környezetben valósul meg, ezek a változások pedig a projekttel szemben támasztott követelményekre is hatással vannak; változtatják az elkészült termékekkel szembeni elvárásokat, ennek következtében pedig módosulhat a projekt hatóköre és vele együtt a feladatokra rendelkezésre álló idő, erőforrások, stb.

A változásmenedzsment az a folyamat, amely révén a változások az alábbi peremfeltételek kielégítése mellett kerülnek kezelésre:

- tervezetten,
- ütemezetten,
- tervek szerint megvalósítva,
- kitesztelve,
- megfelelően kommunikálva,
- projektszereplők által elfogadott és
- nyomon követhető módon.

Szoftverfejlesztési szempontból a változások általában forráskód módosításával, de akár új modulok kifejlesztésével is együtt járhatnak.

A változásoknak projekten belüli és projekten kívüli hatásaival is számolni kell.

9.9.2. Célok

A változás menedzsment célja a projekt során bekövetkező változások átgondolt és kontrollált keretek között való kezelése. A változásmenedzsment elsődleges célja, hogy a változások úgy váljanak a projekt részévé, hogy az a projekt eredményeinek integritását és megbízhatóságát semmiképpen ne csökkenthesse.

Fontos továbbá az is, hogy a változások teljes életciklusa nyomon követhető legyen, és a változásmenedzsment folyamata kontrollált és szisztematikus módon integrálódjon a projektfolyamatokba.

9.9.3. A változásmenedzsment folyamatának ellenőrzése

9.9.3.1. Dokumentációk, házirendek és eljárások ellenőrzése (opcionális)

9.9.3.1.1. Az ellenőrzés célja

A cél annak eldöntése, hogy létezik-e formálisan definiált változásmenedzsment-folyamat.

9.9.3.1.2. Input

- Változásmenedzsment terve (opcionális)

9.9.3.1.3. Minőségellenőrzési feladatok

Az ellenőrzés során meg kell győződni az alábbi szempontok megvalósításáról.

- Létezik dokumentált változásmenedzsment folyamata
  - A változások menedzselésére és koordinálására léteznek kijelölt felelősök (akik jóváhagyják a változásokat és akiknek erre felhatalmazásuk is van)
  - A változásmenedzsment folyamata tisztázott (előkészítés, javaslattétel, döntés, kihirdetés)
  - A változásmenedzsment folyamatokban érintett projekttagok tisztában vannak a változásmenedzsment folyamataival
  - A változásokhoz kapcsolódó termékek definiáltak
  - A változásigények nyomon követhetősége biztosított
  - Rögzítésre került, hogy milyen esetekben lehet eltérni a normál változásmenedzsment eljárásaitól (pl. sürgősségi esetek)
- A változások dokumentálására van kidolgozott folyamat
- A változások nyomon követésére van kidolgozott folyamat
- Létezik eljárás a változásmenedzsment-folyamatok, folyamatleírások frissítésére

9.9.3.2. Változásigénylések ellenőrzése (kötelező)

9.9.3.2.1. Az ellenőrzés célja

A cél annak eldöntése, hogy a változásigények megfelelően vannak-e előkészítve ill. jóváhagyva.

9.9.3.2.2. Input

- Változásigénylő lapok (kötelező)

9.9.3.2.3. Minőségellenőrzési feladatok

Az ellenőrzés során meg kell győződni az alábbi szempontok megvalósításáról.

- Az igénylőlap tartalma megfelelő-e. Az igénylőlappal szemben elvárás, hogy minimum az alábbi adatokat tartalmazza.
  - Egyedi azonosító
  - Igénylő adatai
  - Igénylő aláírása
  - A változásigény oka
  - A változási igény típusa (hiba, módosult igény, új igény, stb.)
  - Az érintett modulok, termékek felsorolása
  - A változásigény megvalósításához szükséges idők, erőforrások, költségek
  - Tervezett ütemezés
  - Felettes neve (opcionális)
  - Felettes aláírása (opcionális)
  - Dátum
- A változási igények, igénylőlapok kezelésével kapcsolatos folyamatok tisztázottak
- Az igénylőlapok nyomon követése megoldott
  - Minden változtatási igényhez tartozik igénylőlap
  - Az igénylőlapok egyedi azonosítóval vannak ellátva
  - Az igénylőlapokkal kapcsolatos események rögzítésre kerülnek (elutasítva, jóváhagyva, megoldva, stb. állapotok, ezekhez tartozó dátumok, stb.)

9.9.3.3. Változásigények végrehajtásának ellenőrzése (kötelező)

9.9.3.3.1. Az ellenőrzés célja

A cél annak eldöntése, hogy a változás igények végrehajtása megfelelően illeszkedik-e a projektfolyamatokba.

9.9.3.3.2. Input

- Forráskódok (kötelező)
- Érintett dokumentációk (kötelező)
- Verziókezelő szoftver (kötelező)
- Fejlesztési környezet (kötelező)

9.9.3.3.3. Minőségellenőrzési feladatok

Az ellenőrzés során meg kell győződni az alábbi szempontok megvalósításáról.

- A változásokkal kapcsolatos tervek és egyéb, ebben a fázisban érintett dokumentumok frissítésre kerülnek
- A változások végrehajtása a megfelelő forráskód-verziókon kerül megvalósításra
- A változások a teszt és éles környezetektől független fejlesztői környezetben kerülnek végrehajtásra
- Megfelelő változás (és konfiguráció) kezelést támogató szoftver áll rendelkezésre
  - A változások implementálása nem vagy minimálisan akadályozhatja az aktuális fejlesztéseket (csoportmunka támogatása; checkin, checkout, stb.)
  - A változások elmentése historikusan történik, a verziók egyértelműen azonosíthatók

9.9.3.4. Változásigényekhez tarozó eredmények tesztelésének ellenőrzése (kötelező)

9.9.3.4.1. Az ellenőrzés célja

A cél annak eldöntése, hogy a változásigények következtében módosult termékek tesztelése megfelelően illeszkedik-e a projektfolyamatokba.

9.9.3.4.2. Input

- Tesztelési tervek (kötelező)
- Tesztjegyzőkönyvek (kötelező)
- Tesztkörnyezet (kötelező)
- Telepítő készletek (kötelező)
- Teszteléshez kapcsolódó folyamatleírások (opcionális)

9.9.3.4.3. Minőségellenőrzési feladatok

Az ellenőrzés során meg kell győződni az alábbi szempontok megvalósításáról.

- A változások tesztjei a fejlesztői és éles környezetektől független tesztkörnyezetben kerülnek végrehajtásra
- A változások ütemezetten, a megfelelő személy által és megfelelően kommunikálva kerülnek át a tesztkörnyezetbe
- Létezik eljárás a kód “befagyasztására”, azaz arra, hogy a tesztkörnyezet bizonyos ideig ne módosulhasson (pl. UAT tesztelés előtt és alatt)
- Tisztázott legyen a fejlesztők és egyéb szereplők feladata a tesztelések alatt
- A tesztek végrehajtása a tesztelési folyamatnál leírt módon történik, a megfelelő személyek által jóváhagyva
- Létezik folyamat a megváltozott kódok, termékek eltávolítására a teszt környezetből (pl. hibás verzió, vagy további módosítási verziók miatt)

- Létezik folyamat a tesztelés alatt történő események dokumentálására és kezelésére

9.9.3.5. Változásigényekhez tarozó eredmények kibocsátásának ellenőrzése (kötelező)

9.9.3.5.1. Az ellenőrzés célja

A cél annak eldöntése, hogy a változásigények következtében előállt termék kibocsátása megfelelően illeszkedik-e a projektfolyamatokba, azaz megfelelően minőségbiztosított és jóváhagyott modul kerül-e kibocsátásra.

9.9.3.5.2. Input

- Forráskódok (kötelező)
- Dokumentációk (kötelező)
- Telepítő készletek (kötelező)
- Éles környezet (kötelező)

9.9.3.5.3. Minőségellenőrzési feladatok

Az ellenőrzés során meg kell győződni az alábbi szempontok megvalósításáról.

- Léteznek módszerek annak ellenőrzésére, hogy az éles környezetbe telepítendő verzió megegyezik a tesztkörnyezetben kitesztelt verzióval
- A kibocsátás folyamatának személyi kérdései tisztázottak (ki végzi a telepítés egyes feladatait)
- A kibocsátás folyamata megfelelően definiált és dokumentált
- Létezik folyamat a kibocsátott termékek eltávolítására az éles környezetből (pl. hibás verzió kibocsátása miatt)
- A változásban érintett dokumentációk megfelelően módosításra kerültek

9.9.3.6. Rendkívüli változási igények kezelésének ellenőrzése (opcionális)

9.9.3.6.1. Az ellenőrzés célja

A változásmenedzsment folyamatokba beletartozik azon változásigények kezelése is, ahol a rendelkezésre álló szűkös végrehajtási idő nem teszi lehetővé a normál folyamatok alkalmazását (sürgős beavatkozások). Tipikusan ilyen esetek azok, amikor az éles rendszerben lép fel olyan probléma, ami sürgős fejlesztői beavatkozást igényel. A cél annak eldöntése, hogy az ilyen esetek kezelésére léteznek-e megfelelő folyamatok.

9.9.3.6.2. Input

- Folyamatleírás (opcionális)

9.9.3.6.3. Minőségellenőrzési feladatok

Az ellenőrzés során meg kell győződni az alábbi szempontok megvalósításáról.

- Létezik folyamat és az megfelelően dokumentált
- Léteznek az ilyen feladatok elvégzéséhez aktiválható jogosultságok
- Ezen jogosultságok birtokában végzett tevékenységek megfelelően auditáltak
- A sürgősségi feladatok jóváhagyásának jogkörei tisztázottak
- Tisztázottak, hogy a feladatok elvégzése milyen egyéb feladatokkal jár együtt (pl. dokumentációk módosítása a feladatok elvégzése *után*, de bizonyos időn belül)
- Létezik folyamat a kibocsátott termékek eltávolítására az éles környezetből (pl. hibás verzió kibocsátása miatt)

10. A minőségbiztosítás folyamatának dokumentumai

A projektben a minőségbiztosítási tevékenység során tipikusan az alábbi típusú dokumentum keletkeznek:

- minőségbiztosítási terv,
- a projekt termékek minőségellenőrzése eredményeként előállt dokumentumok,
- a projektben ellenőrzési, illetve szakértői tevékenység során készített dokumentumok (észrevételek, javaslatok).

A továbbiakban a minőségbiztosítási tervet részletezzük.

10.1. Minőségbiztosítási terv

A minőségbiztosítási tevékenység a minőségbiztosítási terv alapján történik a projektben. Ez lényegében egy magas szintű munkaterv: a rendszeres és eseti tevékenységeket, valamint azok termékeit rögzíti. Célja, hogy áttekintést adjon a projektben alkalmazandó minőségbiztosítási előírásokról mind a termékekre, mind az eljárásokra vonatkozóan.

Ebben a dokumentumban rögzítik a projektben használt módszereket, a vizsgálatok, ellenőrzések mélységét. A használt módszerek – ahogy korábban említettük – függnek a projekt jellegzetességeitől (pl. projekt értéke, komplexitása, kockázatok, stb.), valamint a rendelkezésre álló erőforrások mennyiségétől.

Megjegyzés: a minőségbiztosítási terv kijelölheti, hogy a projektben mely termékeket, vagy folyamatokat fognak ellenőrizni egyenként, és melyeket csak szúrópróbaszerűen. Mindkét esetben szükséges meghatározni a vizsgálatok mélységét. Ez függ a vizsgálat tárgya által hordozott kockázatoktól, valamint a rendelkezésre álló erőforrásoktól és a vizsgálandó objektumok számától. Ha ismétlődő tevékenységet vizsgálnak, a tevékenység végzésének kezdetén gyakoribb, később ritkább ellenőrzéseket terveznek általában.

A tervet a projekt minőségbiztosítási vezetője készíti és a projekt szponzor véleményezi, határozza meg a súlyokat és fogadja el. A dokumentum a projekttervezés szakaszban készül el.

A *minőségbiztosítási tervvel szembeni minimális tartalmi elvárások* a következők.

- **Bevezetés**
  - Dokumentum célja
  - A minőségbiztosítás hatóköre
    A minőségbiztosítás pontosan mely tevékenységekre és termékekre vonatkozik.
  - Hivatkozások
    A terv létrehozásánál figyelembe vett dokumentumok, kiegészítő anyagok.

- Fogalomtár
  A dokumentumban használt speciális fogalmak, rövidítések.

- **Minőségbiztosítási célok**
  A projekt kitűzött minőségcéljai, minőségi kritériumok (vagy hivatkozás azokra). Mivel a minőség több dimenziós fogalom, többféleképpen kell mérnünk. A minősítési, illetve kiértékelési szempontok meghatározásánál azonban világosan rögzítenünk kell, hogy mit akarunk ellenőrizni, vizsgálni, hogyan.

A minőségi célok teljesülésének megítélésekor egyértelműen tudni kell megrendelői és teljesítői oldalról is, hogy mikor tekintünk teljesítettnek egy-egy követelményt, végül az egész követelményhalmazt. A részletesebb minőségi kritériumok meghatározását ezért legkésőbb a projekt fázisok elején egyeztetni kell.

- **Minőségbiztosítás módja**
  - A minőségbiztosítás módja, terjedelme és mélysége
  - Használt módszertanok (opcionális)
- **Minőségbiztosítási szervezet**
  - A szervezet felépítése (esetleg viszonya az intézmény más szervezeti egységeivel) és helye a projekt szervezetben.
  - Feladatok és felelősségek
- **Ellenőrzési és audit terv**
  A projekt egyes munkafolyamataira vonatkozó ellenőrzési és auditálási terv
  - Előre definiált és szúrópróbaszerű ellenőrzések (termékellenőrzések, folyamatellenőrzések)
  - Az ellenőrzések tervezetten felhasznált bemenetei, azok vizsgálati módjai.
- **Dokumentációk**
  A minőségbiztosítás egyéb dokumentumai, vagy olyan dokumentumok, melyekben a projekt minőségbiztosításával kapcsolatos részek vannak.
- **Szabványok és útmutatók**
  Azok a szabványok, előírások, szabályzatok, amelyek a fejlesztés során használatra kerülnek, illetve amelyeket figyelembe kell venni. (Pl. fejlesztési minta, felhasználói interfész útmutató, modellezési útmutató, tervezési útmutató, programozási útmutató és teszt útmutató, ellenőrzési listák, stb.)
- **Metrikák**
  A szoftver termékhez, folyamathoz, vagy kapcsolódó dokumentációhoz tartozó használt mérési típusok.
- **Minőségi jelölések**
  A projekt dokumentumok, azok szerzőinek és az ellenőrzőinek azonosítására vonatkozó előírások.
- **Minőségbiztosítási oktatás**
  A projekt jellemzőinek megfelelően kell kialakítani a minőségbiztosítással

kapcsolatos oktatási tervet, az oktatás tartalmát, jellegét (pl. személyes, vagy dokumentumon / weben keresztül) és mennyiségét.

11. Vonatkozó szabványok

Szabvány kódja	Neve
ISO/IEC 9126-1:2001	Software engineering -- Product quality -- Part 1: Quality model
ISO/IEC TR 9126-2:2003	Software engineering -- Product quality -- Part 2: External metrics
ISO/IEC TR 9126-3:2003	Software engineering -- Product quality -- Part 3: Internal metrics
ISO/IEC TR 9126-4:2004	Software engineering -- Product quality -- Part 4: Quality in use metrics
IEEE 730-2002	Standard for Software Quality Assurance Plans
IEEE 1061-1998 (R2004)	Standard for Software Quality Metrics Methodology
ISO/IEC 21827:2002	Capability Maturity Model® for Software

12. Kapcsolódó információk

http://www.sei.cmu.edu/cmmi/

http://www.sei.cmu.edu/cmm/

http://www.microsoft.com/msf

http://msdn.microsoft.com/vstudio/teamsystem/msf/msfcmmi/default.aspx

http://www.ibm.com/software/awdtools/rup/

http://www.ibm.com/developerworks/rational/library/5318.html

http://www.extremeprogramming.org/

http://www.pairprogramming.com/

http://www.sei.cmu.edu/iso-15504/

http://www.iso.org/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=38932&COMMID=&scopelist=

http://www.sqi.gu.edu.au/spice/