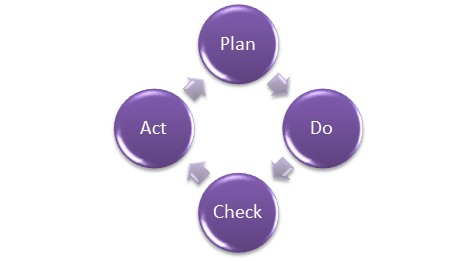
Quality assurance és Quality control

# Quality Assurance

Minőség: Nehezen definiálható fogalom, kb azt értjük rajta, hogy a termék a feladatának elvégzésére és használatra alkalmas. Tudjuk mérni az igények és követelmények teljesítésén keresztül.

Biztosítás: Pozitív állítás a termékről vagy szolgáltatásról, ami bizalmat kelt. Jól fog működni és nem mutat majd problémákat a felhasználás közben. Megint csak az igényeknek és követelményeknek megfelelően.

Minőségbiztosítás: eljárás, aminek célja, hogy biztosítsa a létrehozott termék vagy szolgáltatás minőségét. A szoftver fejlesztési folyamat fejlesztésével foglalkozik, a célja hogy az képes legyen a szükséges minőségi szint elérésére.



Plan: Tervezési lépés. A szervezet tervez és létrehoz célokat, majd meghatározza a folyamatokat, amik ahhoz szükségesek, hogy magas minőségű terméket állítson elő

Do: Változtatási lépés. A szervezet leírja/változtatja és teszteli a folyamatokat

Check: Ellenőrző lépés. A szervezet monitorozza a folyamatokat és ellenőrzi, hogy a célok elérését biztosítják e. (pl gap analysis)

Act: Cselekvési lépés. A szervezet (QA tesztelő) további lépéseket határoz meg, amivel a folyamatokat fejleszteni lehet. (elemezzük az előző 2 lépésben gyűjtött adatot és felkészülünk a következő tervezésre, ez adja a víziót a tervhez)

A fenti lépések ismétlődnek a szervezet folyamatainak fejlesztése alatt.

Milyen feladatai vannak:

* Technology transfer: Létező technológia feldolgozása és beleépítése a termékbe vagy szolgáltatásba.
* Validáció: Az egész termékre vonatkozó validációs terv elkészítése. Ehhez teszt kritériumok és ezek teljesülésének leírása. Erőforrás és végrehajtás megtervezése.
* Dokumentáció: Elérhetővé tétele és tárolása. Változás szabályozása. Elfogadások (aláírások) managelése.
* A minőség biztosítása
* Tervezés a minőség javítására

Lásd még ISO 9000.

# Quality control

Szoftver fejlesztési folyamat, aminek célja a termékben vagy szolgáltatás minőségének biztosítása. A fókusz nem a folyamatokon, hanem a terméken van.

A QC célja az, hogy a QA által definiált és gondozott folyamatokhoz illeszkedve ellenőrizze, hogy a termék vagy szolgáltatás megfelel a leírt követelményeknek és felhasználói igényeknek.

Összehasonlítva

| QA | QC |
| --- | --- |
| Folyamat készítésről és fejlesztésről szól. Standardok, folyamatok és eljárások akár külső példa után való implementálása. Példa feladat egy auditra felkészülés. | Termék teszteléséről szól, mielőtt az eladásra kerül. Mindenféle verifikációs folyamatok ellátása. Például review, statikus tesztelés, manuális tesztelés. |
| Folyamat központú | Termék központú |
| Preventív | Korrektív |
| Proaktív | Reaktív |
| Átfogóan foglalkozik a szervezet fejlesztésével | Termékkel foglalkozik (és ehhez örököl dolgokat a QA-tól) |

# Test maturity model (TMM)

| TMM szint | Cél | Mit kell elérni |
| --- | --- | --- |
| Level 1: Alap | A szoftver fusson | * Ezen a szinten nincsenek folyamatok * A tesztelés célja annak biztosítása, hogy a szoftver fusson * Nincs erőforrás, eszköz és hozzáértő munkaerő * Nincs QA check mielőtt szállítunk |
| Level 2: Definiált | Tesztelési és debug célok és eljárások fejlesztése | * A debugging és a tesztelés elkülönül * Tesztelés a kódolás után jön * A tesztelés fő célja az, hogy megmutassuk, hogy a szoftver teljesíti a specifikációkat * Alapvető tesztelő technikák már használatosak |
| Level 3: Integrált | A tesztelés belefoglalása a szoftver életciklusába | * Az egész életciklusba bele van építve a tesztelés * A teszt objective-ek a követelmények alapján vannak megírva * Teszt csapat létezik * A tesztelés mint szakma és értékteremtés van kezelve |
| Level 4: Management és mérések | Mérhetővé tesszük a tesztelést | * A tesztelés mérhető és mért folyamat * A review a fejlesztési folyamat minden lépésben elfogadott tesztelési input * Újrahasználatra és regressziós tesztelésre a tesztesetek össze vannak gyűjtve és fenntartva egy adatbázisban * Defektek fel vannak jegyezve és el vannak látva severity-vel |
| Level 5: | Teszt folyamat optimalizálás | * A tesztelés jól definiált és szervezett * A tesztelés hatékonysága és ára mérhető * A tesztelést lehet finomítani és folyamatosan javítani * QC és defect prevention működik * Folyamatokat fel lehet használni több helyen (reuse) * A teszt metrikákhoz van eszköz támogatás * Eszközök támogatják a tesztesetek készítését és a defektek gyűjtését |

# Dokumentumok

Ezek a tesztelés előtt, vagy közben jönnek létre/kerülnek kitöltésre. A dokumentum csomag függ attól, hogy milyen terméket fejlesztünk, a standardkotól, amiket használunk és a fejlesztési folyamat érettségétől. Figyelni kell rá, hogy mekkora energiát fektetünk bele, mert könnyen költségesebb lehet annál, mint amennyi értéket teremt.

| Típus | Leírás |
| --- | --- |
| Tesztelési irányelvek | Magas szintű dokumentum, ami leírja, hogy milyen irányelveket és módszereket alkalmazunk, és milyen céljai vannak a teszteléssel a szervezetnek |
| Teszt stratégia | Magas szintű dokumentum, ami azonosítja a teszt szinteket és típusokat, amiket a projectben használunk |
| Teszt terv | Egy átfogó tervező dokumentum, ami tartalmazza a tesztelés scope-ját, a megközelítést, az ütemtervet, erőforrásokat, meg ami még fontos lehet |
| Követelmény követési mátrix | Egy dokumentum, ami magában foglalja az információkat a követelmények és a tesztek összefüggéséről |
| Teszt szcenárió | Egy olyan rész, vagy funkció egy termékből, amit egy vagy több tesztesettel tudunk verifikálni |
| Teszteset | Bementi értékek, előfeltételek, kimeneti állapotok és/vagy kimenetek halmaza. A teszt szcenárió része |
| Teszt adat | Olyan adat, ami a teszt lefuttatása előtt is rendelkezésre áll és használjuk a teszthez |
| Defekt report | Dokumentum, ami egy defekt leírását tartalmazza |
| Teszt összegző report (vagy teszt record) | Magas szintű dokumentum, ami összegzi a tesztelés során végzett munkát és az eredményeket |

Amire érdemes figyelni:

* QA csoport korai részvétele a project-ben
* A dokumentumokat nem csak elkészíteni kell, de fenn is kell tartani
* Verziókövetés sok fejfájást tud megelőzni
* Érdemes értékalapon hozzáállni, azt dokumentálni, ami segíti a munka végzését és megértését (minden résztvevőnek), nem pedig felesleges dokumentumokat gyártani. Az egész cél az, hogy segítsük a megértést és megelőzzük a félreértéseket.
* Standard template-ek sokat segítenek
* Közösen elérhető és egységes tárolás sokat segít. Ennek lehet szabályozni az elérhetőségét igény szerint
* Túl általános dokumentumot senki nem fog használni
* Nem csak rendszerszerű megközelítést, de oktató anyagot is nyerünk vele
* Audit, vendég, vásárló a dokumentációt fogja nézni, nem a kódot
* Sok máshol is használható rész van benne (pl rendszer telepítés és teszt környezet telepítés)
* Dokumentáció írás költséges és időigényes
* Sokszor a kompetencia és a hajlandóság nincs együtt dokumentálásnál
* A változások követése borzalmas munka
* Rosszul managelt vagy fenntartott dokumentáció sok félreértést tud előidőzni