

QualitySoftware

Felhasználói Kézikönyv

1. Bevezető

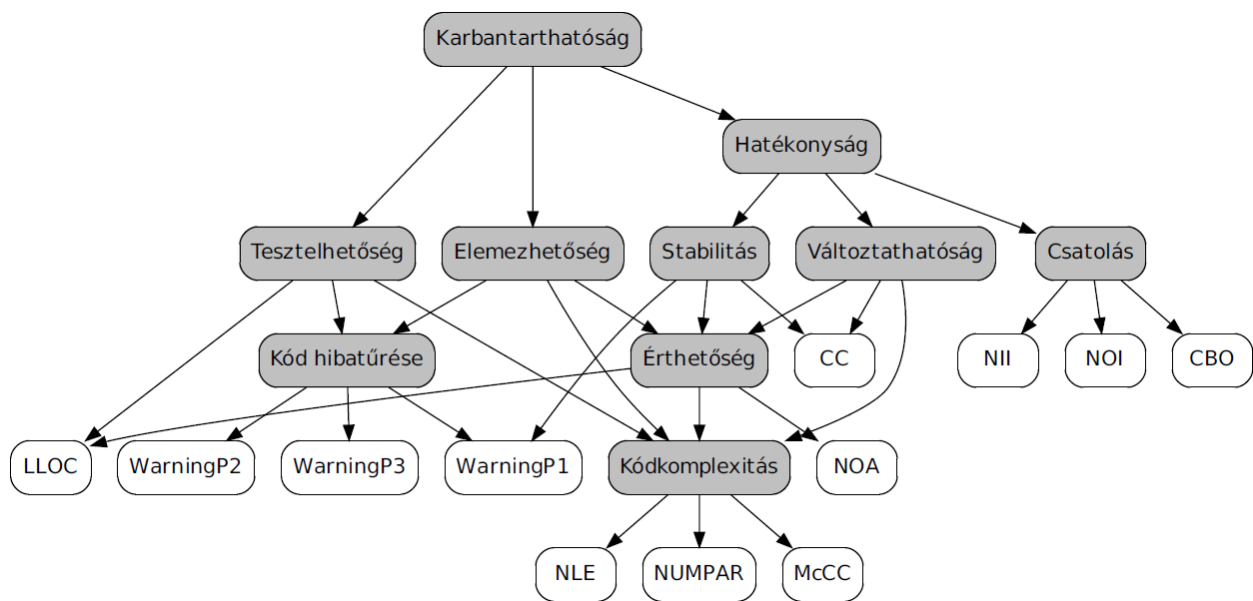
A folyamatos változtatásoknak köszönhetően a szoftverrendszerek állandó minőségromlásnak – szoftvererózióknak – vannak kitéve, amelynek következtében növekednek a fejlesztési és tesztelési költségek, valamint az üzemeltetési kockázatok. Az egyedi fejlesztésű szoftverek karbantartása a teljes életciklus során sokszor meghaladja a kezdeti kifejlesztés költségének többszörösét is.

A nagyszámú, egyedi fejlesztésű rendszereket üzemeltető szervezetek esetében az erózió hatásának mérséklése kulcsfontosságú. A funkcionálisan akár helyes működésű szoftverrendszerek minőségromlásából adódó kockázatokat a fejlesztést megrendelő és az átadott rendszert üzemeltető fél viseli. A forráskód minőségének objektív kiértékelése nélkül csupán az átvételi tesztek eredményeire hagyatkozva lehet megítélni a leszállított rendszer minőségét, ami kizárólag a rendszer funkcionalitását jellemzi, de nem ad képet annak karbantarthatóságáról és üzemeltetési költségeiről.

1.1 A forráskód-karbantarthatóság mérése

A mérés egyik alappillére az ún. referencia adatbázis (*benchmark*), amely nagyszámú szoftverrendszer forráskód-jellemzőjét tartalmazza. A referencia adatbázis összehasonlítási alapként szolgál a kiértékelésre kerülő szoftverrendszerek számára. Ugyanazon referencia adatbázist használva különböző rendszerek, vagy ugyanazon rendszer több verziója esetén a rendszerek karbantarthatósága összehasonlíthatóvá, illetve a karbantarthatóság alakulása időben követhetővé válik.

A forráskód-karbantarthatóság mérésének másik fontos eleme a karbantarthatósági modell, amely alacsony- és magas szintű jellemzőket reprezentáló csomópontokból álló irányított, körmentes gráf (lásd 1. ábra). Az alacsony szintű jellemzők – a gráf bemenő élek nélküli csúcsai (szenzor csomópontok) – a forráskód alapján, statikus elemzés segítségével közvetlenül számítható mérőszámok (pl.: a forráskód-elem komplexitása, súlyos kódolási hibák aránya, kódmásolatok aránya, stb.). A magas szintű jellemzők – a gráf bemenő élekkel rendelkező csúcsai (aggregált csomópontok) – a forrás (pl. karbantarthatóság, változtathatóság, stb.). A gráf csúcsai közötti élek függőség fennállását jelentik. A modell lehetőséget biztosít a gráf éleinek szakértői súlyozására, ezáltal a karbantarthatóság-számítás során a fogalmak szubjektívitasából adódó bizonytalanság kezelhető. Minden szakértő által meghatározott súlyozás egy szavazatnak tekintendő, az egyes éleken pedig a leadott szavazatok valószínűségi eloszlása kerül rögzítésre, amely így minden szakértő véleményét kifejezi.



1. ábra: Forráskód-karbantarthatóság modell

A karbantarthatóság-mérés utolsó építőköve maga a kiértékelés módszere. A kiértékelni kívánt szoftverrendszer esetén először a modellben szereplő alacsony szintű forráskód-jellemzők előállítására kerül sor. Ezután következik azok összehasonlítása a referencia-adatbázisban szereplő szoftverrendszerek mindegyikével. Az összehasonlítás eredménye egy valószínűségi eloszlásfüggvény minden egyes alacsony szintű, modellben definiált jellemző esetén. Ezt követően a modellben szereplő élek mentén a már kiszámított valószínűségi eloszlások – az élekhez rendelt szavazat-eloszlások alapján – statisztikai módszerek segítségével aggregálásra kerülnek, amely objektív mértékül szolgál a magasabb szintű jellemzők számára. A folyamatot többször megismételve, legvégül a gráf gyökerét alkotó *Karbantarthatóság* is mérhetővé válik.

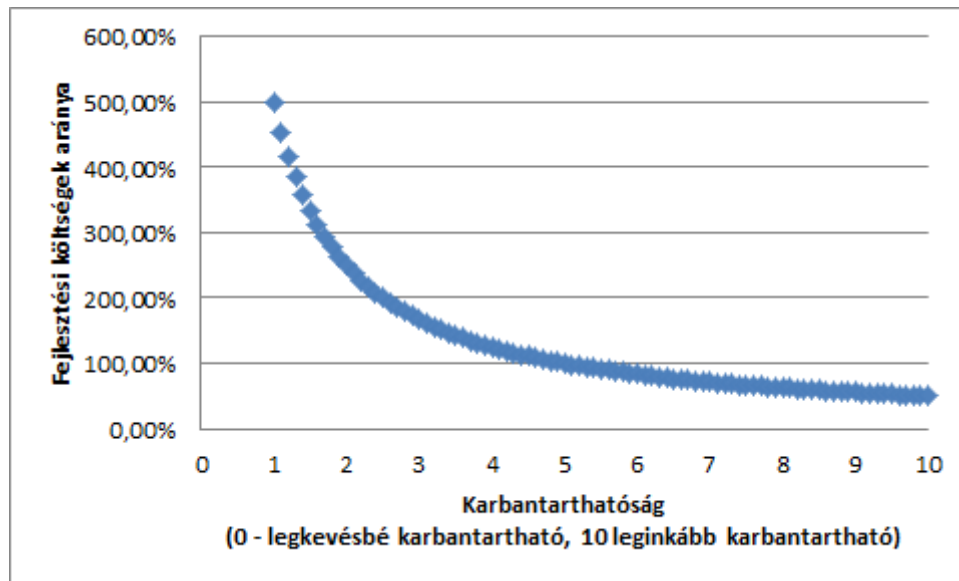
A forráskód-karbantarthatóság mérésének teljes elméleti háttere a mérés alapjait részletesen leíró publikációban olvasható.

1.2 Forráskód-karbantarthatóság és fejlesztési költségek viszonya

A forráskód-karbantarthatóság jelentősége a szoftverfejlesztési költségekhez fűződő nyilvánvaló kapcsolatából fakad. A karbantarthatóságot a leggyakrabban a szoftver viselkedésének módosításával járó erőforrás-szükségletként definiálják.

A kidolgozott költség-modell alapján a forráskód-karbantarthatóság a fejlesztésre fordított erőforrások exponenciális függvénye, azaz a fejlesztésre fordított erőforrások és a szoftvererózió üteme exponenciális kapcsolatban állnak egymással. A kidolgozott modell lehetőséget biztosít arra, hogy a becsült fejlesztési költségek egy átlagos karbantarthatóságú rendszerhez viszonyítva, százalékos formában kifejezésre kerüljenek. Az 2. ábra a különböző karbantarthatósági értékkel rendelkező rendszerek fejlesztési költségeinek mértékét reprezentálja.

A forráskód-karbantarthatóság és fejlesztési költségek viszonyának teljes elméleti háttere a költség-modell alapjait lefektető publikációban olvasható.



1. ábra:Karbantarthatóság és fejlesztési költségek kapcsolata

2. A QualitySoftware koncepció

2.1 Mi a QualitySoftware?

A QualitySoftware a XYZ Kft. által fejlesztett forráskód-minőségbiztosítási keretrendszer, amely az ISO/IEC-9126 szabványon alapul.

- A QualitySoftware egy menedzsment eszköz, amelynek segítségével könnyen feltárhatja a szoftverfejlesztések minőségét.
- A QualitySoftware a forráskód elemzésével méri és monitorozza a szoftverek karbantarthatóságát.
- A QualitySoftware Holisztikus módon ábrázolja a forráskód minőségének időbeli alakulását.
- A QualitySoftware elősegíti a szoftver-erózió kezelését.
- A QualitySoftware javaslatokat tesz a forráskód karbantarthatóságának túlzott romlása esetén.
- A QualitySoftware segítségével a kódminőség és a fejlesztő csapat teljesítménye mérhető és javítható.
- A QualitySoftware képet ad a karbantartási költségek várható alakulásáról.

2.2 Kik a QualitySoftware felhasználói?

2.2.1 Szoftverüzemeltető cégek

- A QualitySoftware a gyenge minőségű rendszereket már a „kapuban” megállítja, megakadályozva a silány minőségű kódok beáramlását.

- Szoftverbeszerzés esetén a QualitySoftware segítségével a különböző szállítók által átadott rendszerek forráskódjának minősége összehasonlíthatóvá válik, újabb döntési szempontot szolgáltatva ezzel a megrendelőnek.
- A QualitySoftware szoftverfejlesztési tudás és kompetencia hiányában is értelmezhető eredményeket szolgáltat az üzemeltetőnek.
- A QualitySoftware felhasználói között vannak pénzügyesek, biztosító társaságok, technológiai cégek, telekommunikációs vállalatok, az energia szektor képviselői és még sokan mások, akik jellemzően egyedi (nem dobozos) szoftverrendszereket fejlesztenek és üzemeltetnek.

2.2.2 Szoftverfejlesztő cégek

- Szoftverfejlesztő cégek a QualitySoftware segítségével mérhetik és nyomon követhetik saját fejlesztéseik minőségét. Ezen információk birtokában megtehetik a szükséges lépéseket, mielőtt még a fejlesztett termék kiadásra kerülne, megakadályozva a cég hírnevének csorbulását és a későbbi karbantartási költségek elszabadulását.
- A szoftverfejlesztő cégek saját folyamataik teljesítményét is monitorozhatják a forráskód-karbantarthatóság alakulásának visszacsatolásával.
- A QualitySoftware segítségével a szoftverfejlesztő cégek webes felületeiken hirdethetik saját rendszerük minőségének valós idejű alakulását is, támogatva ezzel értékesítésüket.

2.3 Hogyan működik a QualitySoftware?

A QualitySoftware a nyílt forráskódú, folyamatos integrációt támogató Jenkins eszközzel együtt kerül telepítésre. A QualitySoftware saját Jenkins bővítménye vezérli a teljes elemzési folyamatot, az alábbiak szerint:

- Rendszeresen ellenőrzi, hogy a verziókövető rendszerben létezik-e újabb, még nem elemzett forráskód-verzió;
- Elindítja a kódbázis adott verziójának statikus kódelemzését, amelyet a keretrendszeren belül a XYZ CodeAnalyzer terméke végez;
- A forráskód-elemzés során, forráskód-metrikák, kódolási szabálysértések, kódmásolatok kerülnek kiszámításra;
- Az elemzés eredményei a QualitySoftware központi adatbázisába kerülnek feltöltésre;
- Kiszámításra kerülnek a megfelelő modellek szerint a forráskód magas szintű jellemzői, így a karbantarthatóság is;
- A QualitySoftware felhasználói felülete lehetőséget ad az adatok megjelenítésére és a különböző riportok elkészítésére illetve a kódminőség változás okainak felderítésére.

2.4 Termékjellemzők

- Támogatott programozási nyelvek: Java
- Integráció egyéb eszközökkel: Jenkins, SVN, LDAP, PMD, XYZ CodeAnalyzer
- Riportgenerálás: PDF (menedzsment szintű), Excel (fejlesztői szintű)

- Támogatott platformok: Windows/Linux (32bit/64bit)
- Támogatott alkalmazás szerverek: Glassfish
- Támogatott adatbázisok: PostgreSQL
- Karbantarthatóság számítása: valószínűségi, referencia-rendszer alapú
- "Drill-down" megközelítés: forráskód-elemekre számított karbantarthatóság-index
- Testreszabhatóság: egyedi referencia adatbázis, egyedi modell kialakításának lehetősége
- Költség-modell: forráskód-karbantarthatóság alapján
- Felhasználói felületek: Web UI
- Forráskód-jellemzők: forráskód-metrikák, kódolási szabálysértések, kódmásolatok

3. A QualitySoftware telepítése

A QualitySoftware rendszert egy dedikált szerverre célszerű telepíteni, majd üzemeltetni.

3.1 Rendszer-követelmények

3.1.1 Minimum követelmények

- Szükséges processzor magok száma: 4
- Szükséges memória: 8 GB
- Szükséges tárhely: 50 GB (projekt(ek) méretétől függően változhat)

3.1.2 Optimális követelmények

- Szükséges processzor magok száma: 8
- Szükséges memória: 24 GB
- Szükséges tárhely: 100 GB (projekt(ek) méretétől függően változhat)

3.2 Támogatott operációs rendszerek

A QualitySoftware az alábbi operációs rendszereket támogatja:

- Windows x86/x64
 - ✦ Windows 7
 - ✦ Windows Server 2008 (R2)
- Linux x86/x64 (debian alapú)
 - ✦ Ubuntu 12.04 LTS
 - ✦ Ubuntu 12.10

3.3 Előfeltételek

A QualitySoftware telepítését megelőzően szükséges néhány előfeltételül szolgáló alkalmazás telepítése és konfigurálása.

1. **Oracle Java JDK 1.7.0_10 (vagy újabb):** az operációs rendszernek megfelelő 32 vagy 64 bites verzió. A telepítő csomag letölthető az Oracle honlapjáról: <http://www.oracle.com/technetwork/java/javase/downloads/index.html>. Telepítést követően a szükséges a JAVA_HOME környezeti változó beállítása a telepített JDK könyvtárára, valamint a JAVA_HOME/bin hozzáadása a PATH-hoz.
2. **LaTeX/MiKTeX telepítése (2.9.x verzió):**
 - a. Windows esetén a telepítő csomag letölthető a <http://miktex.org/download> oldalról.
 - b. A támogatott Linux operációs rendszerek esetén a telepítést a `sudo apt-get install texlive-full` paranccsal lehet elvégezni.
3. **Graphviz telepítése (Linuxon 2.26.3, Windows-on 2.28):**

- a. Windows esetén a telepítő csomag letölthető a <http://www.graphviz.org/Download.php> oldalról, vagy a <http://www.graphviz.org/pub/graphviz/ARCHIVE/> oldalról.
 - b. A támogatott Linux operációs rendszerek esetén a telepítést a `sudo apt-get install graphviz` paranccsal lehet elvégezni.
4. **PostgreSQL 9.1 adatbázis szerver:** a telepítő csomag letölthető a PostgreSQL hivatalos oldaláról (<http://www.postgresql.org/download/>), vagy a támogatott Linux operációs rendszerek alatt telepíthető a `sudo apt-get install postgresql` parancs segítségével. Az adatbázis telepítését követően érdemes az alapértelmezett beállításokat felülrírni, mivel az a teljesítmény jelentős javulásához vezethet (a módosításokat követően az adatbázis újraindítása szükséges). A beállítások Linux operációs rendszer esetén a `/etc/postgresql/[verzió]/main/postgresql.conf` állományban, Windows operációs rendszer (és alapértelmezett telepítés) esetén a `c:\Program Files\PostgreSQL\[verzió]\data\postgresql.conf` állományban található:
- a. **shared_buffers = a rendelkezésre álló rendszer-memóriának 25%-a.** (Linux operációs rendszer esetén a postgresql.conf fájl módosításán kívül a következő teendők vannak:
 - i. PAGE_SIZE méretének lekérdezése a `getconf PAGE_SIZE` paranccsal;
 - ii. `/etc/sysctl.conf` állományba a következők beillesztése:
`kernel.shmmax=renelkezesre_allo_memoria_50%-a_bajtokban`
`kernel.shmall=kernel.shmmax_elosztva_PAGE_SIZE-al`
 - iii. `sudo sysctl -p /etc/sysctl.conf` paranccsal érvénybe helyezhetők a beállítások.
 - b. **effective_cache_size = a rendelkezésre álló rendszer-memóriának 50%-a.**
FIGYELEM: Az `effective_cache_size` tulajdonság előtt található `#` karaktert ki kell törölni, hogy a megadott érték legyen érvényben.
5. **PostgreSQL JDBC driver:** az illesztőprogram letölthető a <http://jdbc.postgresql.org/download.html> oldalról.

FIGYELEM: A TELEPÍTŐ ALAPÉRTELMEZETTEN 9.1-ES VERZIÓJÚ JDBC DRIVERREL RENDELKEZIK. HA ETTŐL ELTÉRŐ VERZIÓJÚ ADATBÁZIS KERÜLT TELEPÍTÉSRE, A MEGFELELŐ JDBC DRIVERT LE KELL TÖLTENI, MAJD LE KELL CSERÉLNI A TELEPÍTŐBEN TALÁLHATÓ .JAR FÁJLT AZ APPLICATION/DATABASE/POSTGRESQL/DRIVER KÖNYVTÁRBAN. (PONTOSAN EGY .JAR FÁJL LEHET A MAPPÁBAN!)

A PostgreSQL rendszerrel kapcsolatos egyéb paraméterezési és optimalizációs lehetőségeket a következő linken lehet találni:

http://wiki.postgresql.org/wiki/Tuning_Your_PostgreSQL_Server

3.4 Telepítési beállítások

A QualitySoftware telepítési paramétereinek beállítása az `install/scripts/configuration.properties` fájlban lévő kulcs-érték párok megadása segítségével történik.

FIGYELEM: ÜGYELJÜNK RÁ, HOGY NE KERÜLJÖN FELESLEGES WHITE-SPACE KARAKTER (PL. SPACE, TAB) A TULAJDONSÁGOK ÉRTÉKE UTÁN!

A *configuration.properties* fájlban az alábbi paraméterek beállítására van lehetőség:

- **xyz-admin-user (OPCIONÁLIS):** a Glassfish alkalmazáserver XYZ termékeket (jelen esetben a QualitySoftware rendszert) biztosító XYZ domainjének adminisztrátor felhasználója (alapértelmezett érték: *admin*).
- **xyz-admin-password (OPCIONÁLIS):** a Glassfish alkalmazáserver XYZ termékeket (jelen esetben a QualitySoftware rendszert) biztosító XYZ domainjének adminisztrátor felhasználójához tartozó jelszó (alapértelmezett érték: *admin*).
- **jenkins-admin-user (OPCIONÁLIS):** a Glassfish alkalmazáserver Jenkins rendszert biztosító domainjének adminisztrátor felhasználója (alapértelmezett érték: *admin*).
- **jenkins-admin-password (OPCIONÁLIS):** a Glassfish alkalmazáserver Jenkins rendszert biztosító domainjének adminisztrátor felhasználójához tartozó jelszó (alapértelmezett érték: *admin*).
- **postgres-port (KÖTELEZŐEN BEÁLLITANDÓ):** adatbázis server elérésének portja (alapértelmezett érték: *5432*).
- **postgres-user (KÖTELEZŐEN BEÁLLITANDÓ):** adatbázis eléréséhez szükséges felhasználónév (alapértelmezett érték: *postgres*). A felhasználónak megfelelő jogosultságokkal kell rendelkeznie adatbázis létrehozásához és törléséhez. Linuxon a PostgreSQL telepítése után alapértelmezettként a postgres felhasználónév nem rendelkezik jelszóval, szükséges ahhoz hozzárendelni egyet, amennyiben ez a felhasználónév kerül alkalmazásra. Ez a következő paranccsokkal tehető meg:
 - ✦ *su postgres*
 - ✦ *psql -U postgres*
 - ✦ `ALTER USER postgres WITH PASSWORD 'postgres';`
- **postgres-password (KÖTELEZŐEN BEÁLLITANDÓ):** az előző pontban megadott felhasználónévhez tartozó jelszó (alapértelmezett érték: *postgres*).
- **postgres-db-name (OPCIONÁLIS):** a XYZ termékek (jelen esetben a QualitySoftware) adatbázisának neve, amelynek kisbetűs megfelelője kerül létrehozásra. Ha ezzel a névvel már létezik adatbázis, akkor a telepítés sikertelen lesz (alapértelmezett érték: *xyz_db*).
- **xyz-instance-port (OPCIONÁLIS):** a XYZ domain port száma (alapértelmezett érték: *10000*).
- **xyz-admin-port (OPCIONÁLIS):** ezen a porton érhető el az alkalmazás server XYZ domainjének adminisztrációs felülete (alapértelmezett érték: *10001*).
- **xyz-java-heap (KÖTELEZŐEN BEÁLLITANDÓ):** a XYZ domain rendelkezésére álló memóriaterület mérete MB-ban kifejezve (alapértelmezett érték: *8192*).

- **xyz-max-perm-size (OPCIONÁLIS):** a Java által az osztálydefiníciók számára fenntartott memóriaterület a XYZ domainben MB-ban kifejezve (alapértelmezés szerint 512).
- **jenkins-home (OPCIONÁLIS):** a Jenkins elemzésvezérlő könyvtára. Itt keletkeznek a projektek elemzési eredményei (binárisok, riportok). A paraméter üresen hagyása esetén közvetlenül a telepítő mappában a JENKINS_HOME-ban jönnek létre az állományok. Példa:
 - ⊕ Windows operációs rendszer esetén `C:\jenkins_home` (az útvonalban szereplő `\` jelek helyett `\\`-t kell írni)
 - ⊕ Linux operációs rendszer esetén `/home/username/jenkins_home`
- **jenkins-instance-port (OPCIONÁLIS):** a Jenkins alkalmazás elérésének port száma (alapértelmezett érték: 20000).
- **jenkins-admin-port (OPCIONÁLIS):** ezen a porton érhető el az alkalmazás szerver Jenkins domainjének adminisztrációs felülete (alapértelmezett érték: 20001).
- **jenkins-java-heap (OPCIONÁLIS):** a Jenkins domain rendelkezésére álló memóriaterület mérete MB-ban kifejezve (alapértelmezett érték: 512).
- **jenkins-max-perm-size (OPCIONÁLIS):** a Java által az osztálydefiníciók számára fenntartott memóriaterület a Jenkins domainben MB-ban kifejezve (alapértelmezett érték: 256).
- **rmi-url (OPCIONÁLIS):** a QualitySoftware által Java RMI-n keresztül kipublikált szolgáltatások elérhetősége. A Jenkins ezek segítségével tölti fel az adatbázisba a forráskódelemzés eredményeit (alapértelmezett érték: localhost:1198).

FIGYELEM! FONTOS, HOGY NE ÜTKÖZZÖN ESETLEGESEN MÁS, SZINTÉN RMI-N ELÉRHETŐ SZOLGÁLTATÁSOKKAL.

- **toolchain-max-memory (KÖTELEZŐEN BEÁLLITANDÓ):** a XYZ CodeAnalyzer eszközök által használható maximális memória mérete MB-ban kifejezve (alapértelmezett érték: 4096).

FIGYELEM: A TELEPÍTÉS MEGKEZDÉSE ELŐTT GONDOSAN ELLENŐRIZNI KELL AZ ÖSSZES KÖTELEZŐEN BEÁLLITANDÓ PARAMÉTERT! AZ ÚTVONALAK ESETÉBEN VALÓS ÚTVONALAT KELL MEGADNI, AHOVA ÍRÁSI JOGA KELL, HOGY LEGYEN A TELEPÍTÉST VÉGZŐ FELHASZNÁLÓNAK. A PORTOK ESETÉBEN NEM LEHET OLYAN PORTOT MEGADNI, AMELYET MÁR MÁS ESZKÖZ VAGY SZOLGÁLTATÁS HASZNÁL. A MEMÓRIA PARAMÉTEREK ESETÉBEN ÜGYELNI KELL ARRRA, HOGY A MEGADOTT MEMÓRIA ÉRTÉKEK ÖSSZEJE, TEHÁT A QUALITYSOFTWARE ÁLTAL HASZNÁLT ÖSSZES MEMÓRIA NE HALADJA MEG A RENDSZER MEMÓRIAMÉRETÉNEK 75%-ÁT!

FIGYELEM: A TELEPÍTÉS SORÁN MEGADOTT xyz-instance-port ÉS jenkins-instance-port PORTOK ELÉRHETŐSÉGÉT ENGEDÉLYEZNI KELL A MEGFELELŐ TÚZFAL BEÁLLÍTÁSOKNÁL, AMENNYIBEN TÁVOLI GÉPEKRŐL EL SZERETNÉNK ÉRNI A FELÜLETEKET!

A telepítés megkezdése után a telepítő ellenőrzi a megadott útvonalak helyességét, illetve a megadott portokat, hogy használatban vannak-e. Az adatbázis szerver portja esetében amennyiben

nem aktív a port, tehát nem lehet elvégezni a szükséges adatbázis műveleteket, úgy szintén jelez a telepítő.

3.5 A telepítés menete

Az előfeltételek telepítését és a telepítési paraméterek beállítását követően elvégezhető a QualitySoftware rendszer telepítése. Ehhez a telepítő csomagban lévő install könyvtárból a következő parancsot kell kiadni:

- Windows operációs rendszer esetén
install.bat
- Linux operációs rendszer esetén
sh install.sh

A telepítés az *Optimális követelmények* c. fejezetben megadott környezet esetén 40 percet vesz igénybe.

FIGYELEM: A TELEPÍTÉS VÉGEZTÉVEL A TELEPÍTŐ MAPPA NEM TÖRÖLHETŐ, MIVEL AZ A RENDSZER MŰKÖDÉSÉHEZ SZÜKSÉGES ÁLLOMÁNYOKAT TARTALMAZI!

Sikeres telepítést követően:

- a XYZ termékeinek közös kiválasztó felülete, a XYZ ProductDashboard felület elérhető a [http://\[host\]:\[xyz-instance-port\]/XYZ-ProductDashboard](http://[host]:[xyz-instance-port]/XYZ-ProductDashboard) címen, ahol *[host]* jelöli a szerveret, amelyen a telepítés történt, *[xyz-instance-port]* pedig a configuration.properties fájlban megadott **xyz-instance-port** értékét (alapértelmezett érték: 10000). Erről a felületről egyetlen kattintással elérhető a XYZ Kft. összes forráskód elemzésen alapuló terméke.
- a QualitySoftware felülete elérhető a [http://\[host\]:\[xyz-instance-port\]/QualitySoftware](http://[host]:[xyz-instance-port]/QualitySoftware) címen, ahol *[host]* jelöli a szerveret, amelyen a telepítés történt, *[xyz-instance-port]* pedig a configuration.properties fájlban megadott **xyz-instance-port** értékét (alapértelmezett érték: 10000).
- az elemzések vezérlését végző Jenkins felület elérhető a [http://\[host\]:\[jenkins-instance-port\]/jenkins](http://[host]:[jenkins-instance-port]/jenkins), ahol *[host]* jelöli a szerveret, amelyen a telepítés történt, *[jenkins-port]* pedig a configuration.properties fájlban megadott **jenkins-instance-port** értékét (alapértelmezett érték: 20000).

3.6 Licenc fájl telepítése

A telepítés végeztével a QualitySoftware használatának megkezdése előtt telepíteni kell az eszközhöz vásárolt licenc fájlokat a következők szerint:

- **QualitySoftware licenc állomány telepítése:** a vásárolt licenc állományt a [http://\[host\]:\[xyz-instance-port\]/XYZ-ProductDashboard](http://[host]:[xyz-instance-port]/XYZ-ProductDashboard) felületen a **Browse** gomb megnyomásával lehet telepíteni, majd az **Activate license** nyomógomb segítségével lehet aktiválni az eszközt. Sikeres aktiválás után a felületen található QualitySoftware ikon szürkére, háttere pedig zöldre vált, és az ikonra kattintással lehet eljutni az eszköz belépő képernyőjére.

- **Forráskód-elemzés vezérlő licenc állomány telepítése:** folyamatos forráskódelemzés céljából létrehozott XYZ CodeAnalyzer Plugin job konfigurációs oldalán információ található azzal kapcsolatban, hogy a forráskód-elemzés vezérlő licenc érvényes-e. Amennyiben nincs érvényes licenc, úgy a rendszer erre figyelmezteti a felhasználót. Érvényes licenc hiányában nincs lehetőség a Job konfigurációjának módosítására, illetve elemzések indítására. A **Browse** gomb segítségével lehet kiválasztani az érvényes licenc állományt, amelyet az **Activate license** nyomógombbal lehet telepíteni.

A XYZ CodeAnalyzer minden forráskód elemzés indítása előtt ellenőrzi a licenc fájl érvényességét. Amennyiben nem érvényes a licenc fájl, úgy az aktuális elemzés hibára fut (**Build failed**), valamint a Job *Disabled* állapotba kerül. Amennyiben aktiváljuk az új licenc fájlt a Job konfigurációs felületén, úgy folytathatóak az elemzések a Job *Enabled* állapotra állítását követően.

3.7 QualitySoftware jogosultságok beállítása

A QualitySoftware alkalmazás LDAP szerver segítségével képes a felhasználók azonosítására. A szerverhez történő kapcsolódáshoz szükséges beállítások a telepítő csomagban a *QualitySoftware-securityContext.xml* fájlban található. Ez a telepítő mappájában létrehozott Glassfish alkalmazás szerveren belül található a *glassfish3/glassfish/domains/XYZDomain/lib/classes* könyvtárban, később ezen a helyen módosítható, amennyiben szükséges. A helyes működéshez a következő információkat kell megadni:

- **LDAP használata:** az LDAP autentikáció QualitySoftware felületen történő engedélyezéséhez a loginRequired paramétert kell beállítani. A paraméter helye a konfigurációs állományban:

```
<bean id="loginConfiguration"
class="com.XYZ.QualitySoftware.security.LoginConfiguration">
    <property name="loginRequired" value="true"></property>
</bean>
```

Amennyiben a paraméter értéke *true*, akkor a felület a bejelentkező képernyővel indul, és oda irányítja a felhasználót amíg érvényes felhasználónév-jelszó páros nem kerül megadásra. Ha a paraméter *false*, akkor a felület nem kér azonosítást, de a következő funkciók nem elérhetőek:

- ✦ benchmark létrehozás, módosítás és törlés
- ✦ modell létrehozás, módosítás, törlés, szavazás és kalibrálás
- ✦ minősítés indítás, törlés
- ✦ minősített rendszer küszöbértékének beállítása

- **LDAP szerver:** LDAP autentikáció használata esetén definiálni kell a felhasználói adatokat biztosító LDAP szerver elérhetőségét az alábbi bejegyzés *value* attribútumában:

```
<bean id="contextSource"
class="org.springframework.security ldap.DefaultSpringSecurityContextSource">
    <constructor-arg value="ldap://localhost:389/dc=XYZ,dc=com"/>
```

```
</bean>
```

- **LDAP csoportok:** az alábbi bejegyzés *value* paraméterben, vesszővel elválasztva meg kell adni, hogy az LDAP-ban szereplő csoportok (Groups) milyen QualitySoftware-beli szerepköröknek (Role) felelnek meg:

```
<bean id="grantedAuthoritiesMapper"  
class="com.XYZ.QualitySoftware.security.LdapGrantedAuthoritiesMapper" >  
    <constructor-arg value="user=USER"></constructor-arg>  
</bean>
```

FIGYELEM: JELENLEG A QUALITYSOFTWARE CSAK A USER SZEREPKÖRÖKET TÁMOGATJA!

- **Felhasználó DN minta:** azon könyvtárbejegyzés, amely szerint a felhasználók definiálva vannak. Itt kell megadni, hogy a felhasználók DN-je (Distinguish Name) milyen konvenció szerint szerepel az LDAP-ban. Amennyiben szükséges, a *value* attribútumban lehetséges átállítani. Alapértelmezetten a felhasználókat *uid* szerint azonosítja a **People** bejegyzésben. Több érték megadása is lehetséges többszörös *value* attribútummal. Az alapértelmezett érték általában megfelelő.

```
<bean  
class="org.springframework.security.ldap.authentication.BindAuthenticator"  
>  
    <constructor-arg ref="contextSource"/>  
    <property name="userDnPatterns">  
        <list><value>uid={0},ou=People</value></list>  
    </property>  
</bean>
```

- **Csoportokra vonatkozó beállítások:** lehetőség van megadni a csoportokat tartalmazó OU-t (Organization Unit), a csoportban levő felhasználókat definiáló szűrőfeltételt, valamint a csoportok név szerinti azonosítására szolgáló attribútum nevét. A csoport OU-ja a második *constructor-arg value* attribútumában, a csoporthoz tartozást megadó szűrőfeltétel az első *property value* attribútumában, míg a megkülönböztető attribútum neve a második *property value* attribútumában állítható be. Alapértelmezetten a csoportokat a **Groups** bejegyzés tartalmazza, a csoportokban a felhasználók **uniqueMember** attribútumokban vannak megadva, ahol a {0} paraméter egy felhasználó teljes DN-je (ezt a keretrendszer biztosítja), a csoportok nevének attribútuma pedig a **cn**. Általában itt is megfelelő az alapértelmezett beállítás:

```
<bean  
class="org.springframework.security.ldap.userdetails.DefaultLdapAuthoritiesPopulator">  
    <constructor-arg ref="contextSource"/>  
    <constructor-arg value="ou=Groups"/>  
    <property name="groupSearchFilter"  
value="(uniqueMember={0})" />
```

```
<property name="groupRoleAttribute" value="cn"/>
</bean>
```

3.8 Jenkins jogosultságok beállítása

A Jenkins rendszer saját autentikációs megoldást biztosít a felhasználók számára, amelyet a következőképpen lehet beállítani:

1. Lépünk be a [http://\[host\]:\[jenkins-instance-port\]/jenkins](http://[host]:[jenkins-instance-port]/jenkins) felületre, ahol *[host]* jelöli a szerveret, amelyen a telepítés történt, *[jenkins-port]* pedig a configuration.properties fájlban megadott **jenkins-instance-port** értékét (alapértelmezett érték: 20000);
2. A bal oldali menüből válasszuk ki a **Manage Jenkins** opciót;
3. A jobb oldalon megjelenő menüsorból válasszuk ki a **Configure Global Security** opciót;
4. Engedélyezzük a **Enable security** opció segítségével a biztonsági beállításokat;
5. A megjelenő beállítások felületen az **Access control/Security Realm** beállításnál válasszuk ki az LDAP opciót;
6. Az **Advanced...** gombra kattintva értelem szerűen beállíthatóak az LDAP autentikációhoz szükséges adatok (Server, root-DN, User search base, Group search base, Manager DN).

3.9 Lehetséges hibák a telepítés során

3.10 A QualitySoftware rendszer törlése

A XYZ termékek és a Jenkins teljes törlése a telepítő *install* mappájában kiadott következő paranccsal végezhető el:

- Windows operációs rendszer esetén
uninstall.bat
- Linux operációs rendszer esetén
sh uninstall.sh

A törlés művelet sikeresnek tekinthető a következők esetében:

- A *configuration.properties* fájlban megadott *[postgres-db-name]* adatbázis törlésre került, vagy nem is létezett.
- A telepítés során feltelepített Glassfish domain-ek nem futnak.
- A telepítő mappában található glassfish3 mappa törlésre került.

3.11 Rendszer leállítása/elindítása

A XYZ termékek és a Jenkins, a telepítés végeztével automatikusan elindításra kerülnek. Amennyiben szükséges a leállításuk, akkor az megtehető a telepítő *install* mappájában kiadott alábbi parancs segítségével:

- Windows operációs rendszer esetén

stop.bat

- Linux operációs rendszer esetén

sh stop.sh

A telepített QualitySoftware rendszerek újra elindíthatóak a következő paranccsal:

- Windows operációs rendszer esetén

start.bat

- Linux operációs rendszer esetén

sh start.sh

4. Forráskód-elemzés vezérlése

A QualitySoftware telepítését követően a forráskód-elemzést vezérlő Jenkins felület az alábbi címen érhető el:

[http://\[HOST\]:\[PORT\]/jenkins/](http://[HOST]:[PORT]/jenkins/)

ahol [HOST] a számítógép neve, amelyen telepítésre került a rendszer, [PORT] pedig a Jenkins rendszert futtató domain portja, amely a telepítés során kerül beállításra (lásd **A QualitySoftware telepítése** fejezet).

A Jenkins felületre történő belépés után a alábbi ábrán látható kép jelenik meg a felhasználó előtt:



The screenshot shows the Jenkins web interface. At the top, there is a search bar and a navigation menu. The main content area displays a table of jobs with columns for 'S' (Status), 'W' (Weather icon), 'Name', 'Last Success', 'Last Failure', and 'Last Duration'. Below the table, there is a 'Build Queue' section showing 'No builds in the queue.' and a 'Build Executor Status' table with columns for '#', 'Status', and 'Name'. The footer contains a localization link, page generation information, and version details.

S	W	Name ↓	Last Success	Last Failure	Last Duration
●	☁	Sample Job	N/A	23 min (#1)	39 sec
●		Sample MultiJob	N/A	N/A	N/A

#	Status	Name
1	Idle	
2	Idle	

4.1 Forráskód-elemzés indítása

A XYZ CodeAnalyzer Plugin a Jenkins keretrendszer egy olyan bővítménye, amely lehetővé teszi a szoftver rendszerek forráskódjának folyamatos elemzését. A rendszerek elemzését ún. *Job*-ok vezérlik. Minden rendszer elemzéséhez dedikált Job létrehozása szükséges, az alábbiak szerint:

1. Új job a Jenkins főmenüben lévő **New Job** menüpont alatt hozható létre.
2. A projekt nevének megadása után ki kell választani a **XYZ CodeAnalyzer Plugin**-t.
3. Először le kell ellenőrizni, hogy az elemzőeszközhöz használt licenc nem járt-e még le. Amennyiben igen, erre a "**License expired**" üzenet figyelmeztet. Abban az esetben, ha a licenc lejárt szükséges a **Browse** gomb segítségével egy érvényes fájl kiválasztása, majd az **Activate license file** segítségével való feltöltése.
4. Amennyiben az **Upload as benchmark job** opció kerül kiválasztásra, akkor az adott Job, mint *benchmark rendszer* kerül feltöltésre. A benchmark rendszert nem lehet minősíteni és csak egy verziót lehet belőle feltölteni.
5. A Job beállításainak oldalán kiválaszthatók a minősítésre használni kívánt modellek.

6. Azt, hogy az elemzés során mely mappák kerüljenek elemzésre, és melyek ne, az a **Use filter** opció beállításával lehetséges.
Például az **svn.apache.org/repos/asf/logging/log4j/trunk/** alatt található **tests** mappa nem kerül elemzésre a következő filter fájl használata esetén:
/
-svn.apache.org/repos/asf/logging/log4j/trunk/tests/
A / hatására az összes **svn.apache.org/repos/asf/logging/log4j/trunk/** alatt található forráskód elemzésre kerül. Azonban a második sorban a - jel hatására a tests mappa kiszűrésre kerül.
7. Megfelelő verziókövető rendszer kiválasztása:
 - a. **SVN** esetén:
 - i. Meg kell adni a repository eléréséhez szükséges felhasználó nevet és jelszót, amennyiben nem szükséges hagyjuk üresen.
 - ii. Fel kell sorolni az elemezni kívánt forráskód elérési útvonalait. Minden egyes elérési útvonalat külön sorban kell megadni, amelynek következtében a sor végén megjelennek az adott útvonal esetén létező revízió számok. Új elérési útvonal megadása az **Add** gomb segítségével, az elérési útvonalak törlése pedig a sor végén megjelenő **Delete** gomb megnyomásával lehetséges. Az elérési útvonalak ugyanazon SVN repository-n belül kell lenniük, amely az **SVN Repository Root** szövegmezőben jelenik majd meg.
 - iii. A **Select SVN revisions** mezőben meg kell adni az elemezni kívánt revíziókat, vagy a konkrét revízió számok vesszővel történő felsorolásával, vagy pedig a revíziók intervallumának megadásával (például: 1-12345, vagy 1-HEAD). HEAD megadása esetén az elemzés a legutolsó elérhető revízióig fog folytatódni, azt követően pedig minden újonnan beérkező revízió elemzése automatikusan megtörténik.
 - b. A lokális fájlrendszeren elérhető kódbázis esetén a **Directory** opció kiválasztásakor:
 - i. Meg kell adni a kódbázis lokális elérési útvonalát a Jenkins-t is futtató rendszeren
 - ii. Meg kell adni a kódbázis verzióját az elérési útvonal mellett megjelenő szövegmezőkben. Amennyiben a könyvtárnév tartalmazza a kódbázis verziójának megfelelő dátumot *Verzió_Év_Hónap_Nap_Óra_Perc_Másodperc* formában, úgy a rendszer automatikusan kitölti a megfelelő verzióinformációkat, ellenkező esetben ezeket kézzel szükséges megadni. Példa: 3426_2013_02_12_14_25_19
8. Amennyiben szükséges, hogy minden elemzés után automatikusan előálljanak az adott verzióra vonatkozó PDF és XLS riportok, abban az esetben be kell jelölni a **Generate reports** checkbox-ot. Alapértelmezett értéként szerepel a riportok helye is, ami a `${JENKINS_HOME}\jobs\${JOBNAME}\workspace\reports\` könyvtár, azonban ezt meg is változtathatjuk. A funkció használata esetén a Job-hoz tartozó felületen táblázatos formában rendelkezésre állnak a sikeres elemzések során előálló riportok.
9. A **Discard Old Builds** opció kiválasztásával a korábbi Jenkins build információk automatikusan törlésre kerülnek a kiválasztott stratégia szerint. Log Rotation strategy esetén

megadható hány napig, illetve az utolsó hány elemzéshez tartozó build információk kerüljenek megőrzésre. Nagyszámú elemzés futtatása esetén ezt érdemes limitálni, mivel jelentős lassulást okoz a Jenkins keretrendszer futtatásában. Az elemzések során előálló releváns információk archiválásra kerülnek, így a funkció használata nem jár jelentős tárhely megtakarítással.

10. Beállítások elmentése.

11. A Job oldalán szereplő **Build now** gomb megnyomásával indítható el az elemzés és a minősítés. **Ezen gomb megnyomásával a kódbázisnak minden esetben csak egy, a soron következő verziójának elemzése történik meg.** A kódbázis folyamatos elemzésének elindításához ún. **MultiJob** létrehozása szükséges (lásd **MultiJob létrehozása** fejezet).

4.2 MultiJob létrehozása

Több rendszer folyamatos elemzésének végrehajtásához egy ún. **MultiJob** típusú Job létrehozása szükséges, az alábbiak szerint:

1. Jenkins főmenüben lévő **New Job** menüpont kiválasztása.
2. A Job nevének megadása után ki kell választani a **MultiJob Project** lehetőséget.
3. Az időzítés beállítása a **Build Periodically** részben megadható **cron** típusú kifejezés (https://en.wikipedia.org/wiki/Cron#CRON_expression) segítségével történik. A leggyakrabban használt kifejezések:
 - ⊕ Job futtatása minden percben: * * * * *
 - ⊕ Job futtatása 20 percenként: 0,20,40 * * * *
 - ⊕ Job futtatása minden éjfélkor: 0 0 * * *
4. A futtatandó kódelemzést végrehajtó Job-ok az **Add build step** alatt elérhető **MultiJob Phase** segítségével adhatók meg. A **Phase name** mezőbe kell beírni a MultiJob fázisának nevét, a **Phase jobs** mezőbe pedig a futtatni kívánt kódelemző Job nevét. További Job-ok hozzáadásához az **Add jobs...** opciót kell kiválasztani.
5. Beállítások elmentése.
6. Az így létrehozott MultiJob az időzítésnek megfelelően el fogja indítani a megadott jobokat.

Több rendszer párhuzamos elemzésének futtatásához a **Manage Jenkins/Configure System** menüpont alatt található **# of executors** mező értékét a párhuzamosan futó elemzések számánál nagyobb értékre szükséges beállítani. Az **Optimális követelmények** fejezetben megadott optimális rendszer használata esetén ennek az értékét javasolt 4-re állítani, így egy szálon az elindított MultiJob, 3 szálon pedig az elemzések fognak futni.

4.3 Elemzés menetének nyomon követése

A Jenkins által vezérelt összes elemzés a Jenkins vezérlőfelületen követhető nyomon:

[http://\[HOST\]:\[PORT\]/jenkins/](http://[HOST]:[PORT]/jenkins/)

Itt láthatók a folyamatban lévő elemzések, az utolsó sikeres illetve sikertelen végrehajtás, valamint az adott Job utolsó lefutásának időtartama is.

A táblázatban szereplő egyes Job-okra kattintva az adott Job oldalára jutunk, amely bal oldalán szerepel a korábbi és aktuális végrehajtások listája, a végrehajtás sorszámával azonosítva. Egy konkrét lefutás kiválasztása után, a **Console output** menüpontot választva a lefutás során előálló napló felület jelenik meg, amelyen szerepelnek a futtatott eszközök, esetlegesen fellépő hibák, stb. Hibás végrehajtás esetén ezen napló segíthet a probléma okának felderítésében.

4.3 Példa console output

```
Started by user anonymous
[EnvInject] - Loading node environment variables.
Building in workspace jenkins_home/jobs/Sample_Job/workspace
Actual revision is: 308873
922 revisions more.
Collecting source to the jenkins_home/jobs/Sample_Job/workspace/src directory.
Update: http://svn.apache.org/repos/asf/logging/log4j/trunk 308873
Modified files and directories:
Sources were succesfully collected!
[scripts] $ ant -file toolchain.xml -DJOB_WORKSPACE=jenkins_home/jobs/Sample_Job/workspace -
DSVN_PROPERTIES=jenkins_home/jobs/Sample_Job/svnconfig.properties -DMODELS=XYZ-Model| -DBENCHMARK_JOB=false -
DJOB_UID=db35caff-5216-4ad6-9dd8-045aed94ab42 -DpackDir=application/binaries/toolchain/x64-linux/toolkit
Buildfile: application/binaries/toolchain/x64-linux/scripts/toolchain.xml
checkUserProperties:
makeDirs:
  [mkdir] Created dir: jenkins_home/jobs/Sample_Job/workspace/results/2013-07-10_15-17-05
  [mkdir] Created dir: jenkins_home/jobs/Sample_Job/workspace/results/2013-07-10_15-17-05/graphs
  [mkdir] Created dir: jenkins_home/jobs/Sample_Job/workspace/results/2013-07-10_15-17-05/logs
setGraphAndRulProperties:
runTools:
JavaAnalyzer:
  [echo] Running Java analyzer...
filter:
doFilter:
JAN2ChangePath:
  [echo] Running JAN2ChangePath...
JAN2Lim:
  [echo] Running JAN2Lim...
DCF:
runDCF:
  [echo] Running duplicated code finder...
JavaCheckGEN:
LIM2Bsm:
LIM2SeaSeb:
JavaCheckSTA:
LIM2Metrics:
PMD:
runJavaCheckGEN:
runLIM2SeaSeb:
runLIM2Bsm:
runJavaCheckSTA:
runPMD:
runLIM2Metrics:
```

```

[echo] Writing metrics...
[echo] Running PMD...
CostPredictor:
runCostPredictor:
graphMerge:
[echo] Running graphMerge
HeaderEditor:
[echo] Running graph HeaderEditor...
[echo] args: -graph jenkins_home/jobs/Sample_Job/workspace/results/2013-07-10_15-17-05/graphs/Sample_Job-merged.bin -rev 308873 -svnurl http://svn.apache.org/repos/asf/logging/log4j/trunk -user - -pass ***** -toolsetver 1 -wc ' ' -projectUID db35caff-5216-4ad6-9dd8-045aed94ab42
UploadGraph:
runUpload:
[echo] Running GraphUploader and Evaluation...
[echo] args: -jobname Sample_Job -graph jenkins_home/jobs/Sample_Job/workspace/results/2013-07-10_15-17-05/graphs/Sample_Job-merged.bin -prevgraph jenkins_home/jobs/Sample_Job/workspace/results/2013-07-10_15-16-29/graphs/Sample_Job-merged.bin -model XYZ-Model| -reportDir ${env.REPORT_DIR}
archiveResultDir:
deleteResultDir:
[delete] Deleting directory: jenkins_home/jobs/Sample_Job/workspace/results/2013-07-10_15-16-29
ArchiveMergedGraphs:
[propertyfile] Updating property file: jenkins_home/jobs/Sample_Job/workspace/results/lastModified.properties
UploadSVNInfo:
runSVN:
default:
BUILD SUCCESSFUL
Total time: 7 seconds
Finished: SUCCESS

```

4.4 Elemzés felfüggesztése és újraindítása

Egy adott Job futtatásának felfüggesztése a Job oldalán található **Disable Project** gomb megnyomásával lehetséges. Ezt követően újabb verzió elemzése nem fog elindulni. Amennyiben egy adott Job lefutása során hiba lép fel, az automatikusan felfüggesztésre kerül. Amennyiben a hiba oka felderítésre és megszüntetésre került, lehetőség van a Job futásának újraindítására, amely szintén a Job oldalán található **Enable Project** gomb megnyomásával lehetséges.

4.5 Beállítások módosítása

Az egyes Job-ok beállításai módosíthatóak, a Job oldalán található **Configure** menüpont segítségével, amely a Job létrehozása kapcsán már ismertetett beállító felületre irányítja a felhasználót. A módosított beállítások a soron következő végrehajtástól kezdődően lesznek érvényben.

Az aktivált és már legalább egy sikeres elemzést végrehajtó Job csak bizonyos beállításai módosíthatóak, amelyek közül a következők átállítása befolyásolhatja az elemzési és minősítési eredményeket:

- Quality modell - sikeres futás után már nem módosítható
- Use filter - megadható illetve módosítható a használni kívánt filter, amely átállítása jelentősen befolyásolhatja a kapott elemzési eredményeket.

- Verzió követő rendszer beállítások - megváltoztathatók az SVN felhasználónév és jelszó beállítások. Felvehetőek új, illetve törölhetőek a már meglévő SVN branch útvonalak, azonban szerkeszteni már nem lehet őket.

4.6 Job-ok törlése

Egy adott Job törlése a Job oldalán szereplő **Delete Project** gomb kiválasztásával lehetséges. Ebben az esetben minden beállítás, korábban előállt elemzési eredmény törlésre kerül.

FIGYELEM: A JOB TÖRLÉSÉVEL TÖRLŐDNEK A QUALITYSOFTWARE ADATBÁZISÁBAN SZEREPLŐ, A JOB-HOZ KAPCSOLÓDÓ ADATOK, ÍGY AZOK A TOVÁBBIAKBAN NEM LESZNEK ELÉRHETŐEK A QUALITYSOFTWARE WEBES FELÜLETEIN!

4.7 További információk

A Jenkins keretrendszerrel, használatával és további beállításával kapcsolatban a keretrendszer hivatalos felhasználói dokumentációi nyújtanak iránymutatást, amelyek elérhetők az alábbi címen:

<http://jenkins-ci.org/>

4.8 Hibaüzenetek

A forráskód-elemzés vezérlése során az alábbi hibák léphetnek fel, amelyeket a következő hibaüzenetek jeleznek:

- **Database is not available.** A QualitySoftware nem érhető el a Jenkins környezetéből. Lehetséges okok, és megoldások:
 - ⊕ A QualitySoftware-et futtató alkalmazás domain nem fut, vagy annak újraindítása szükséges. Ebben az esetben le kell állítani a folyamatban lévő elemzéseket Jenkins felületen (**Elemzés felfüggesztése és újraindítása** fejezet), majd újra kell indítani a QualitySoftware alkalmazás domain-jét.
- **This field is required.** Az adott mező kitöltése kötelező, de még nincs kitöltve.
- **[PATH] is not exist.** A Jenkins-t futtató gépen a megadott [PATH] útvonal nem elérhető.

Lehetséges okok, és megoldások:

 - ⊕ Az alapértelmezett mappától eltérő útvonal lett megadva riport mappának. Ebben az esetben le kell ellenőrizni a megadott útvonalat, illetve létre kell hozni a kívánt könyvtárat.
 - ⊕ Könyvtár elemzésekor a Folder URL mezőben a megadott útvonal nem létezik. Ebben az esetben ki kell javítani az útvonalat, hogy az ténylegesen az elemezni kívánt mappára mutasson.
- **Please check the svn urls.** Ha nincs megadva SVN útvonal, vagy pedig, rossz útvonal lett megadva. Ebben az esetben meg kell adni egy SVN elérhetőséget, vagy pedig ki kell javítani a hibás (**Please check the svn url**) branch-et.
- **Please check the svn url.** Rossz a megadott SVN útvonal.

Lehetséges okok, és megoldások:

- ✦ Rossz a megadott SVN útvonal, ellenőrizhető például a böngészőbe történő bemásolással, ekkor ki kell javítani az útvonalat.
- ✦ Rossz a felhasználónév/jelszó, ebben az esetben a megfelelőt kell megadni.
- **Revision [REVISION] is not in the repository.** Vesszővel felsorolt revíziók esetén a lista tartalmaz olyan verziót, ami nem szerepel egyik megadott SVN repository-ban sem. Ebben az esetben ki kell javítani a felsorolást.
- **There should be at least one real revision in the selection!** A megadott SVN intervallum nem tartalmaz egyetlen verziót sem. Ebben az esetben olyan intervallumot kell megadni, például 1-HEAD, ami már tartalmaz legalább egy tényleges verziót.
- **The field can only contains one type of svn selection. Series or an interval of svn revisions.** Egyszerre csak egy fajta verziókijelölést lehet használni, vagy vesszővel elválasztva adható meg az SVN verziók sorozata, vagy pedig kötőjellel azoknak egy intervalluma.
- **The field can only contains the following characters 0 1 2 3 4 5 6 7 8 9 , - and the HEAD word.** A verziókijelölés során kizárólag a 0 1 2 3 4 5 6 7 8 9 , - karakterek és a HEAD szó használható. Ebben az esetben ki kell javítani a kijelölést.
- **Wrong Date format. Example is in the Directory help.** Nem megfelelő formátumban lett megadva a dátum a könyvtár elemzésének beállítása során. Le kell ellenőrizni, hogy nem került-e be felesleges karakter az egyes mezőkbe. A mezőhöz tartozó helpben található egy példa a kitöltésre.
- **These fields are required.** Kötelező kitölteni a könyvtár elemzésének beállításakor az összes mezőt.
- **Year must be between 1970 and [YEAR].** Az évnek 1970 és az aktuális év között kell lennie.
- **Month must be between 1 and 12.** A hónapnak 1 és 12 között kell lennie.
- **Day must be between 1 and 31.** A napnak 1 és 31 között kell lennie.
- **Hour must be between 0 and 23.** Az órának 0 és 23 között kell lennie.
- **Minute must be between 0 and 59.** A percnek 0 és 59 között kell lennie.
- **Second must be between 0 and 59.** A másodpercnek 0 és 59 között kell lennie.
- **The date should be earlier than now.** Az aktuális dátum utáni dátum lett megadva. Javítani kell a tényleges dátum megadásával.

5. A QualitySoftware használata

A QualitySoftware felhasználói felülete szoftverfejlesztési projektek vezetői, szoftverüzemeltetők, valamint IT vezetők számára nyújt holisztikus képet a szoftverrendszereik minőségéről, karbantarthatóságáról. A rendszer felhasználói számára nincs szükség mély szakismeretekre, csupán a szoftverkarbantarthatóság fogalmának és jelentőségének megismerésére.

A QualitySoftware használatához jogosultsági beállításoktól függően (lásd **Jogosultságok beállítása** fejezet) felhasználói bejelentkezésre lehet szükség. Bejelentkezést követően a felhasználó a rendszer három fő funkciócsoportját láthatja egy-egy menüpontban:

- Rendszer-minősítések kezelése
- Minősítő modellek kezelése
- Benchmarkok kezelése

A fejezet további részében a felhasználói felületek fontosabb használati eseteit mutatjuk be.

5.1 Benchmarkok kezelése

Egy rendszer minősítéséhez egy referencia adatbázis (úgynevezett benchmark) megléte szükséges. Egy rendszer minőségét ezen referenciaadatbázisban szereplő rendszerek viszonylatában lehet számszerűsíteni. A QualitySoftware a telepítést követően rendelkezik egy alapértelmezett benchmark adatbázissal, amely száz nyílt és zárt forráskódú szoftverrendszert és azok elemzési eredményeit tartalmazza. A rendszerben található benchmarkok listája a **Benchmark** oldalon tekinthető meg.


5.1.1 Benchmark részleteinek megtekintése

A **Benchmark** oldalon listázott benchmarkokat jelölő információs dobozra kattintva lehetőség van egy adott Benchmark áttekintő információinak illetve statisztikai adatainak megtekintésére. A felület tartalmazza a benchmark nevét, leírását, a benne található rendszerek számát, azok listáját és utolsó módosításának idejét. A benchmarkban található rendszerek főbb forráskódjellemzőinek (sorok, csomagok, osztályok, metódusok száma) átlag illetve szórás értékei is megtekinthetők. A benchmarkban található rendszerek mellett a kapcsolódó minőségmodellek is fel vannak sorolva, valamint az összesített metrikaértékeket kördiagramok is ábrázolják.


5.1.2 Új benchmark létrehozása

Benchmark az **Új benchmark létrehozása** gomb segítségével hozható létre. A gomb megnyomása után a felületen kiválasztható, hogy mely rendszerek mérési eredményeiből készüljön referenciaadatbázis a későbbi minősítésekhez. Mentés után a Benchmark oldalon megjelenik az újonnan létrehozott referenciaadatbázisnak megfelelő információs doboz.

5.1.3 Benchmark szerkesztése

Egy már létező benchmark szerkesztésére a Benchmark oldalon lévő benchmarkokat ábrázoló információs dobozok bal alsó sarkában található  ikonra kattintva nyílik lehetőség. Ekkor betöltődnek a kiválasztott benchmark adatai, amelyeknek szerkesztésére is van mód, a benchmark neve, leírása, illetve a tartalmazott rendszerek is módosíthatók. A név és leírás módosításától eltekintve nem szerkeszthetők azok a benchmarkok, amelyekhez létezik legalább egy kalibrált minőségmodell (lásd **Minőségmodell létrehozása** fejezet).

5.1.4. Benchmark törlése


Már létező benchmark törlése a Benchmark oldalon lévő benchmarkokat ábrázoló információs dobozon található  ikon kiválasztásával lehetséges. Ez a művelet csak akkor végezhető el, ha nincs olyan minősítő modell, amely ezt a benchmarkot használja.

5.2 Minőségmodellek kezelése

A szoftverek minősítése egy minősítő modell alapján történik. A modell alacsony- és magas szintű jellemzőket reprezentáló csomópontokból álló irányított, körmentes gráf (lásd ábra). Az alacsony szintű jellemzők – a gráf bemenő élek nélküli csúcsai – a forráskód alapján, statikus elemzés segítségével közvetlenül számítható mérőszámok (pl. a forráskód-elem komplexitása, súlyos kódolási hibák aránya, kódmásolatok aránya, stb.). A magas szintű jellemzők – a gráf bemenő élekkel rendelkező csúcsai – a forráskódból közvetlenül nem számítható, általában szubjektív megítélésű fogalmakat reprezentáló csomópontok (pl. karbantarthatóság, változtathatóság stb.). A gráf csúcsai közötti élek függőség fennállását jelentik. A modell lehetőséget biztosít a gráf éleinek szakértői súlyozására, ezáltal a karbantarthatóság-számítás során a fogalmak szubjektivitásából adódó bizonytalanság kezelhető. Minden szakértő által adott súlyozás egy szavazatnak tekintendő, az egyes éleken pedig a leadott szavazatok valószínűségi eloszlása kerül rögzítésre, amely így minden szakértő véleményét kifejezi.

5.2.1 Minőségmodell megtekintése

A Modell oldalon megjelenő minőségmodellek bármelyikét ábrázoló információs dobozra való kattintással annak részletei megtekinthetők. A modellek megtekintése egy, a szerkesztéshez hasonló felületen történik, ám a modell egyes tulajdonságai, adatai nem módosíthatók. A modell alapadatai mellett (pl.: név, leírás, használt benchmark, pontok száma) maga a minősítő modell gráfja is megtekinthető. A modell csomópontjaira duplán kattintva kiegészítő információ kapható az adott csomóponttól (név, leírás, milyen típusú forráskód elemekre van értelmezve, stb.). Ez mellett a gráf

élein megjelenő  ikonra kattintva az adott függőségre leadott szavazatok eloszlása és egyéb statisztikai információk jeleníthetők meg. A minőségmodell elemei tetszőlegesen elrendezhetők a jobb áttekinthetőség érdekében.

5.2.2 Minőségmodell létrehozása

Új minőségmodell létrehozására a **Modell** oldalon az **Minőségmodell létrehozása** gomb segítségével van lehetőség. Új modell esetén a modell nevének, leírásának megadását követően szükséges kiválasztani a használni kívánt benchmarkot (minden modell pontosan egy benchmarkot használ). Ezen adatok megadása után a **Modell összeállítás** >>> gombra kattintva a grafikus modell szerkesztő felület segítségével felépíthető az új modell.

Új alacsony szintű jellemző a jobb oldalon található Alacsony szintű jellemzők fülön található adatok megadásával lehetséges. A forráskód-elem típusának megadását követően, kiválasztható az adott típusra értelmezett forráskód-metrika, amely a modell gráf egy levél csúcsának fog megfelelni. Ezek megadásakor a név és leírás mezők automatikusan kitöltődnek a kiválasztott metrika megfelelő adataival, ám ezek a mezők tetszőlegesen módosíthatók. Új magas szintű jellemző létrehozása a Magas szintű jellemzők fülön található név illetve leírás megadását követően lehetséges. A létrehozott modell csúcsok a bal oldalon található felületen jelennek meg, amelyen az egér segítségével szabadon elrendezhetők, illetve közöttük függőségek létesíthetők.


A modell gráfnak az alábbi feltételeket kell teljesítenie:


- A modell gráfnak összefüggőnek kell lennie
- A modell gráfnak pontosan egy gyökere kell, hogy legyen

- A modell gráfban nem lehet kör
- Minden magas szintű jellemzőnek legalább két, de legfeljebb három bemenő éllel kell rendelkeznie

A modell létrehozása során ezen feltételek ellenőrzése automatikusan megtörténik, amennyiben valamelyik nem teljesül, nincs lehetőség a létrehozott modell elmentésére.


Mentés után az újonnan létrehozott modellt ábrázoló információs doboz megjelenik a modell felületen. Ahhoz viszont, hogy a modell alkalmas legyen a minősítésre, szükséges még az egyes minőségi mutatók közötti függőségek súlyozása is. Ezzel lehetséges finomhangolni azt, hogy melyik mutató milyen mértékben befolyásolja a végeredményt. Egy modellre többen is szavazhatnak, de legalább egy szavazatnak léteznie kell. A modell súlyozásához a most létrehozott modellhez tartozó

információs dobozon a  ikonra kell kattintani, amely a modell megjelenítő felületére visz. Itt meg kell adni a szavazóra vonatkozó alapadatokat (név, tapasztalat, stb.), majd minden magas szintű jellemzőre (csomópontra) szavazni kell. Szavazás során megadható, hogy az adott csomópont értékebe milyen arányban számítsanak bele az egyes alatta lévő mutatók.


Végezetül, a szavazatok mentése után a modellt kalibrálni szükséges, amelyet a  ikonnal lehet elindítani. A kalibrálás során a minősítéshez szükséges előzetes statisztikai számítások elvégzése történik. Az egyes modellek információs dobozain azonnal látható, hogy egy modell kalibrálva van-e vagy sem.

FIGYELEM! EZ A FOLYAMAT A BENCHMARKBAN LÉVŐ RENDSZEREK SZÁMÁTÓL, AZOK MÉRETÉTŐL, A MODELL SZERKEZETÉTŐL ÉS KOMPLEXITÁSÁTÓL FÜGGŐEN, AKÁR IDŐIGÉNYES FOLYAMAT IS LEHET (EZT MODELLENKÉNT CSAK EGYSZER KELL ELVÉGEZNI).

5.2.3 Minőségmodell szerkesztése


Már létező modell szerkesztése a Modell oldalon, a szerkeszteni kívánt modell információs dobozán lévő  ikonra kattintva lehetséges. Ezután a modell létrehozásához is használt szerkesztő felületen a modell minden adata módosítható. Nem szerkeszthetők azonban azok a modellek, amelyekre már történt legalább egy szavazás.

5.2.4 Minőségmodell törlése

Minőségmodell törléséhez a Modell felületen lévő modelleket ábrázoló információs doboz jobb alsó sarkában található  ikonra kell kattintani. Ekkor a modell törlődik, így azzal többé már nem lehet minősítést végezni. Nem törölhető azonban olyan minőségmodell, amelyhez létezik minősítés.

5.3 Rendszer-minősítések kezelése

A **Minősítés** oldalon tekinthetők meg a QualitySoftware-ben nyilvántartott rendszerek különböző modellek szerinti minősítései. Minden egyes információs doboz egy rendszer egy adott modell szerinti minősítését jelképezi. Az információs dobozon a rendszer és a minősítéshez használt modell neve mellett szerepel az utolsó kiértékelés időpontja illetve az ehhez az időponthoz tartozó forráskód-verzió

minősítésének értéke és annak szöveges értékelése. A minősítés értéke tízes skálán mozog (0 - legrosszabb, 10 - legjobb), és a minősítéshez használt modellben megadott mérőszámok illetve a modell által használt benchmarkban szereplő rendszerek minőségének viszonylatában fejezi ki az adott rendszer minőségét. Az információs dobozon lévő  gomb megnyomásával az utolsó kiértékelt forráskód-verzióra vonatkozó legfontosabb rendszer szintű jellemzők tekinthetők meg (rendszermetrikák, szabálysértések száma).

5.3.1 Minősítés részleteinek megtekintése

A minősítés felületen a megfelelő információs doboz kiválasztásával megtekinthetők a rendszer adott modell szerinti minősítés eredményei egy összefoglaló felületen. Ezen a felületen látható a rendszer minőségi értékét jelző csillagsor, mellette a minősített rendszer neve, értéke - ha történt változás, akkor a változás értéke -, a minőség értéke zöld illetve piros színnel attól függően, hogy az az előre meghatározott határérték alatt vagy felett van. Az előző verzióhoz viszonyított minőségváltozást egy felfelé mutató zöld nyíl illetve egy lefele mutató piros nyíl jelzi.


Az adatok alatt egy idővonal mutatja a rendszer minősítésének alakulását. Ez mellett pókháló diagramok ábrázolják az egyes magas- illetve alacsony szintű modell-pontok értékeit az utolsó kiértékelt verzióban.

A minőséget ábrázoló idővonal színe piros illetve zöld lehet. A zöld szín azt jelenti, hogy a rendszer minőségi értéke meghalad egy előre definiált küszöbértéket, ami egy 0 és 10 közötti számérték. Ezzel az értékkel állítható be, hogy egy adott rendszertől milyen minősítési érték az elvárt. Az aktuális rendszer minősítését ábrázoló információs doboz színe is ettől függ: ha a legutolsó kiértékelt verzió minősége meghaladja ezt a küszöböt, akkor az információs doboz zöld színű, egyébként pedig piros.

Minden minősítés alapértelmezett küszöbértéke 5. A küszöbérték beállítása a  ikonnal lehetséges.

A minőséget ábrázoló idővonal a rendszer minősítésének alakulását mutatja, ám lehetőség van tetszőleges alacsony és magas szintű minőségi jellemző alakulását is megjeleníteni. Ehhez az idővonal mellett megjelenő pókháló diagramokon ábrázolt pontok valamelyikére kell kattintani, ami után a megfelelő jellemző idővonala közvetlenül a minősítési idővonala alá töltődik be. Ily módon könnyen össze lehet hasonlítani ezen két érték alakulását az időben.

Az idővonalon található minden egyes mérési pont kijelölhető, és részletes információ kérhető a forráskód-verzióval kapcsolatban. Megtekinthető, hogy mely forráskód jellemzők változása okozta a minőség változását (zöld szín jelzi a minőségi javulást, piros pedig a romlást), és hogy az adott metrikaérték rendszer szintű átlaga százalékosan mennyit változott. Ezek mellett különböző riportok készítésére is van lehetőség az adott verzióra vonatkozóan, lásd **Rendszer-minősítő riport generálása** fejezet.


A minőségváltozás mellett a  ikon kiválasztásával a rendszer relatív karbantartási költsége is megjeleníthető, amely a rendszerbe épített költségmodell alapján kerül kiszámításra. A rendszer karbantartási költségének mértéke százalékosan van kifejezve, amely azt ábrázolja, hogy a rendszer karbantartása mennyibe kerül egy átlagos rendszerhez képest (a 100% egy 5-ös minőségű rendszer karbantartási költségét jelenti).

5.3.2 Rendszer-minősítés végrehajtása


A minősítések automatikus, ütemezett indítását a korábban ismertetett Jenkins elemzésvezérlő végzi a QualitySoftware felületről függetlenül (lásd **Forráskód-elemzés vezérlése** fejezet). A QualitySoftware segítségével egy már feltöltött rendszer minősítése kézzel is elindítható a **Minősítések** oldalon a **Minősítés indítása** gombra kattintva. Először a minősíteni kívánt rendszer kiválasztása szükséges. Itt csak azon rendszerek jelennek meg, amelyekre lehetséges minősítést indítani (nem lettek még leminősítve a rendszerben található kalibrált modellek mindegyikével). Ezután szükséges megadni a használandó modellt (a legördülő listában csak azok a modellek jelennek meg, amelyekkel még nem lett minősítve a fentebb kiválasztott rendszer). A modell kiválasztása után megadható, hogy a rendszer mely verziója kerüljön minősítésre.

Lehetőség van a rendszer legutolsó verziójának, vagy egy megadott verzió utáni összes verzióknak (beleértve a kiválasztottat is) a minősítésére. Az utóbbi lehetőség esetében egy legördülő listából ki kell választani a kívánt verziót, amelytől a minősítés kezdődjön. A minősítést a **Minősítés indítása** gomb megnyomásával lehet elindítani, amely sikeres lefutása után a minősítés részletei felületre jutunk. Az így létrehozott minősítések küszöbértéke alapértelmezetten 5 lesz.

5.3.3 Rendszer-minősítés törlése


Egy adott rendszer valamely minősítésének törlése a **Minősítés** oldalon található megfelelő információs dobozon szereplő  ikon segítségével lehetséges. A minősítés törlésével az adott szoftver forráskód-jellemzői nem kerülnek törlésre (ezen információk törlését a Jenkins elemzésvezérlő felületen lehet végrehajtani, lásd **Job-ok törlése** fejezet), így a minősítés a **Minősítés indítása QualitySoftware felületről** fejezetben leírtak szerint újra végrehajtható, illetve ugyanezen rendszer egyéb modellek szerinti minősítése is előállítható.

5.3.4 Rendszer-minősítő riport generálása

Döntéshozók és vezetők számára egy adott rendszer minőségéről vezetői jellegű PDF riport generálására van lehetőség. Ezt a riportot a rendszer minősítéseit ábrázoló információs dobozon, vagy a minősítés részletei felületen megjelenő minőségi értékek idővonalán szereplő mérési pontokat kijelölve megjelenő  ikon segítségével lehet előállítani. Az így módon előálló riport tartalmazza a minősítés tárgyát képező szoftver forráskódjának verzió azonosítóját, a minősítés értékét, a magas szintű jellemzők számszerű alakulását, a minősítéshez használt modell leírását, a legkritikusabb forráskód-elemek és szabálysértések listáját, illetve a modellben szereplő alacsony- és magas szintű jellemzők szerinti bontást.

5.3.5 Szakértői riport generálása

Szoftverfejlesztők és projektvezetők számára, egy adott rendszer minősítésének alapján XLS formátumú riport generálható, amely áttekintést ad a rendszer adott verziójának állapotáról metrikák tekintetében, illetve az egyes forráskódelemek minőségi mutatóit illetően.


Ezen riport alapján a forráskód minősége és karbantarthatósága javítható. A szakértői riport a rendszer minősítéseit ábrázoló információs dobozon, vagy a minősítés részletei felületen megjelenő minőségi értékek idővonalán szereplő mérési pontokat kijelölve megjelenő  ikon segítségével állítható elő.

A szakértői riportban az alábbi információk találhatóak meg:

- Minősített rendszer főbb adatai, minősítés eredményeinek áttekintése
- Metrikaérték-változások az előző verzióhoz képest
- Új kódolási szabálysértések az előző verzióhoz képest
- Az adott verzió kritikus forráskódelemei és metrikái
- Az adott verzióban fellelhető összes szabálysértések listája
- Az adott verzióban fellelhető kódmásolatok listája

5.3.5 Widget-kód generálása

Widget kód generálásával egy olyan forráskód-minőséget igazoló HTML kód előállítására van lehetőség, amelyet egy weboldalba beágyazva valós időben nyomon követhető az adott rendszer forráskódjának minősítése egy esztétikus pecsétet ábrázoló embléma formájában. A pecséten megjelenik a minősített rendszer neve, az utolsó minősítés dátuma valamint az aktuális minőségi érték. Az így előállított embléma a rendszert fejlesztők számára marketing és PR célokra használható, a felhasználók számára pedig a rendszer minőségére vonatkozóan nyújt biztosítékot.

A beágyazható HTML kódot a Minősítés oldalon, a megfelelő rendszer minősítéseit reprezentáló információs dobozon található  ikonnal lehet legenerálni.

6. Támogatás

A QualitySoftware rendszer telepítése és használata során felmerülő hibák bejelentésére és támogatás kérésére a XYZ Kft. egy erre a célra kialakított felületet biztosít ügyfelei számára, amely a következő címen érhető el:

<https://support.XYZ.com>

A megrendeléskor automatikusan létrejön egy projekt, valamint egy felhasználói fiók, amelyről az érintettek e-mail értesítést kapnak. A rendszer segítségével nyomon követhető a bejelentett hibák életciklusa a bejelentéstől egészen a lezárásig, illetve a javítócsomag kiadásáig.

Új hibajegy felvétele esetén a hiba és a hibához vezető eseménysorozat pontos leírása szükséges, valamint a jelenséget bemutató képernyőkép. Lehetőség szerint mellékelni kell tömörített formában a releváns napló állományokat, amelyek a következő útvonalakon érhetőek el:

- `$glassfish-rootdirectory/domains/$qg-domain/logs`
- `$glassfish-rootdirectory/domains/$jenkins-domain/logs`

Ahol a `$glassfish-rootdirectory` a telepítéskor megadott Glassfish szerver gyökér könyvtára, illetve a `$qg-domain` és `$jenkins-domain` pedig a telepítéskor megadott QualitySoftware és Jenkins domainek nevei.

A hiba bejelentése után a XYZ Kft. support csapatának a feladata felmérni a hiba súlyosságát, értesíteni az illetékeseket, illetve visszajelezni a hibát bejelentő ügyfélnek a várható javítási időről. A hibabejelentési folyamat lezárásaként előálló javítócsomag kiadásáról az ügyfél értesítést kap, amelyet ugyanezen rendszerből lehet majd letölteni.

A QualitySoftware rendszer használatával és üzemeltetésével kapcsolatosan felmerülő kérdéseket a support@XYZ.com e-mail címre várjuk. A kérdések megválaszolása a XYZ Kft. support csapatának a feladata, akik minden információt biztosítanak a rendszer gördülékeny üzemeltetéséhez.

Függelék

Forráskód-metrikák leírása

Kódolási szabálysértések listája

Kódmásolat-metrikák leírása