**KoKu testiautomaatio**

**Ympäristön asennusohjeet**

Luottamuksellinen

02.03.2012

B.Alden

Ixonos_logo.emf

© Ixonos Plc 2012

Asiakirjan muutoshistoria

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Versio | Päiväys | Tekijä | Tarkastaja | Sisällön kuvaus |
| 0.1 | 29.02.2012 | Björn Alden |  | Dokumentin ensimmäinen versio |
| 0.2 | 02.03.2012 | Björn Alden |  | Sisältöä muutettu Java package nimeämismuutoksen mukaan |
| 0.3 | 02.03.2012 | Björn Alden |  | Sisältöä päivitetty vastaamaan luokkarakenteen refaktorointia |

Sisällysluettelo Sivu

Otsikko

[1 Johdanto 4](#_Toc318470337)

[1.1 Tämän dokumentin sisältö 4](#_Toc318470338)

[1.2 Muut taustatiedot 4](#_Toc318470339)

[2 Muutama sana ratkaisun kehitysvaiheesta 4](#_Toc318470340)

[3 Testiautomaatioratkaisun asennus 5](#_Toc318470341)

[3.1 Esivaatimukset ennen asennusta 5](#_Toc318470342)

[3.2 Asennusohjeet 5](#_Toc318470343)

[3.2.1 Ratkaisun lähdekoodin noutaminen versionhallintajärjestelmästä 5](#_Toc318470344)

[3.2.2 Eclipse projektin luonti Maven työkalulla 5](#_Toc318470345)

[3.2.3 Projektin tuonti Eclipse kehitysympäristöön 6](#_Toc318470346)

[3.2.4 M2\_REPO classpath muuttujan määrittely Eclipse IDE:ssä 6](#_Toc318470347)

[3.2.5 Lisäkomponenttien asennus Eclipse kehitysympäristöön 6](#_Toc318470348)

[4 Testiautomaatioratkaisun käyttö 12](#_Toc318470349)

[4.1 Kohdeympäristön konfigurointi 12](#_Toc318470350)

[4.2 Testiajon suoritus Maven työkalulla 12](#_Toc318470351)

[4.2.1 Testiajon konfigurointi – ui\resources\testng.xml 12](#_Toc318470352)

[4.2.2 Testiajon käynnistys 12](#_Toc318470353)

[4.2.3 Testituloksien raportointi (Maven testiajon yhteydessä) 12](#_Toc318470354)

[4.3 Testiajon suoritus Eclipse kehitysympäristössä 12](#_Toc318470355)

[4.3.1 Testiluokan ajo - *Run As -> TestNG Test* 12](#_Toc318470356)

[5 Java toteutuksen rakenne 14](#_Toc318470357)

[6 Nykytoteutuksen puutteet 14](#_Toc318470358)

[7 Vinkkejä jatkokehitykseen 14](#_Toc318470359)

# Johdanto

## Tämän dokumentin sisältö

Tämä dokumentti on tehty Kohti Kumpanuutta (KoKu) projektin käyttöön. Dokumentissa kuvataan, miten KoKu käyttöliittymän testaukseen tarkoitettu testiautomaatioratkaisu asennetaan.

## Muut taustatiedot

Testiautomaatioratkaisu on kehitetty nopealla aikataululla. Kyseinen ratkaisu voidaan käyttää pohjaratkaisuna KoKu käyttöliittymän testiautomaation jatkokehityksessä.

Lisätiedot:

* Björn Alden

# Muutama sana ratkaisun kehitysvaiheesta

Ratkaisu on kehitetty vaiheittain seuraavasti:

* Ratkaisun suunnittelu ja toteutustavan valinta
  + Selenium WebDriver pohjainen ratkaisu
* Ympäristön alustus Maven työkalulla
  + Pohjaratkaisu generoitu Selenium ohjeiden mukaisella pom.xml tiedostolla
  + Eclipse projektin luonti Maven työkalulla – Tämän jälkeen projekti on tuotu Eclipse kehitysympäristöön
* Ratkaisun kehitys Eclipse kehitysympäristössä
  + Muutaman KoKu testitapauksen toteutus jatkokehityksen pohjaksi
* Maven lisäkomponenttien integrointi
  + TestNG testiajuri (korvaa JUnit testiajurin)
  + ReportNG raportointi (TestNG raportoinnin sijaan)
* Eclipse plug-in lisäkomponenttien integrointi
  + Maven/TestNG testiajojen suorituksen mahdollistaminen Eclipse kehitysympäristössä
* Java luokkahierarkian optimointi/refaktorointi ylläpidon helpottamiseksi (testaajan näkemyksen mukaan)
* Hyödyllisten apufunktioiden toteutus ja integrointi luokkahierarkiaan

# Testiautomaatioratkaisun asennus

## Esivaatimukset ennen asennusta

Tarvittavat komponentit:

* Windows työasema
  + Java ympäristö
  + Internet Explorer
  + Firefox
* Eclipse IDE kehitysympäristö asennettu
* Maven työkalu asennettu

Testiautomaatioratkaisun kehityksen aikana käytetyt versiot on esitetty alla olevassa taulukossa.

|  |  |
| --- | --- |
| *Komponentti* | *Versiotiedot* |
| Työaseman käyttöjärjestelmä | Windows 7 Enterprise, Service Pack 1 |
| Java ajoympäristö | 1.6.0\_30 |
| Internet Explorer | 8.0 |
| Firefox | 10.0.2 |

*Taulukko-1*: Käytettyjen komponenttien versiotiedot

## Asennusohjeet

### Ratkaisun lähdekoodin noutaminen versionhallintajärjestelmästä

Ratkaisu on tallennettu KokU projektin SVN versionhallintajärjestelmään.

Nouto onnistuu seuraavan osoitteen avulla:

* <https://172.16.17.23/kohtikumppanuutta/koku-ta/trunk/ui>

Lisätiedot SVN käytöstä:

* Marko Asplund

### Eclipse projektin luonti Maven työkalulla

Eclipse projekti luodaan seuraavalla komennolla:

* mvn eclipse:eclipse

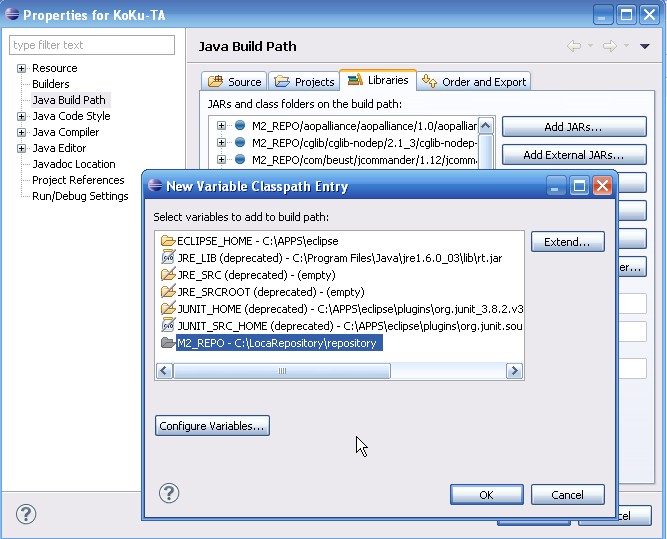
### Projektin tuonti Eclipse kehitysympäristöön

Projekti tuodaan Eclipse ympäristöön seuraavan valikon kautta:

* *File -> Import... -> Existing Projects into Workspace*

### M2\_REPO classpath muuttujan määrittely Eclipse IDE:ssä

Eclipse ympäristössä on syytä tarkistaa, että M2\_REPO classpath muuttuja on määritelty. Tarpeen mukaan määritellään kyseinen muuttuja osoittamaan omaan paikallisen maven hakemistoon.



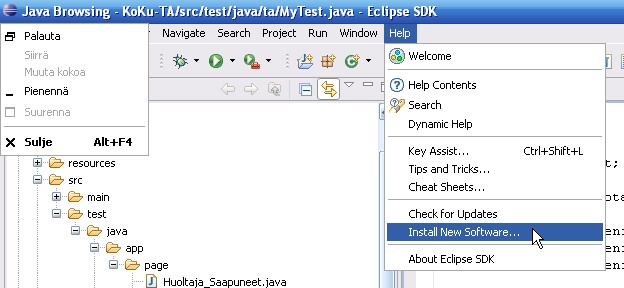
*Kuva-1:* M2\_REPO muuttujan määrittely Eclipse ympäristössä

### Lisäkomponenttien asennus Eclipse kehitysympäristöön

Eclipse kehitysympäristöön asennetaan seuraavat lisäkomponentit:

* m2eclipse
* TestNG Eclipse

Asennus tapahtuu *Help* -valikon kautta (katso kuva alla).



*Kuva-2:* Tarvittavan lisäkomponentin asennus Eclipse ympäristössä (*Help -> Install New Software...*)

#### m2eclipse plug-in asennus

Kyseinen komponentti mahdollistaa Maven työkalun käytön Eclipse kehitys­ympäristössä.

*Lisätiedot:*

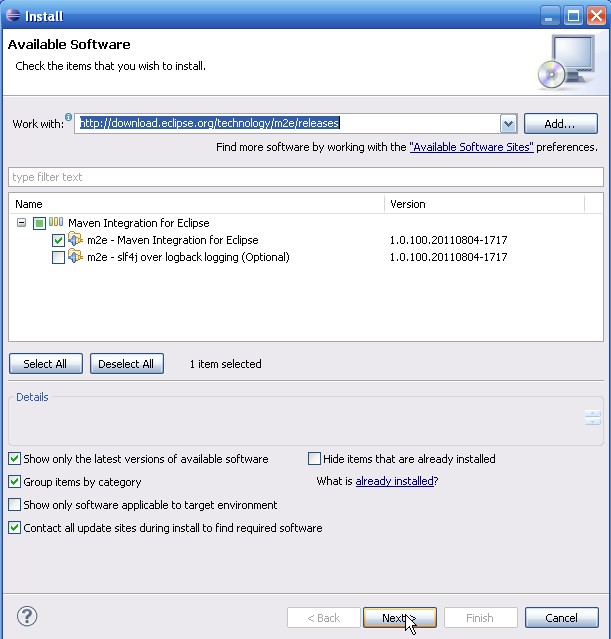
* <http://www.eclipse.org/m2e>

*Eclipse asennuksessa tarvittava osoite:*

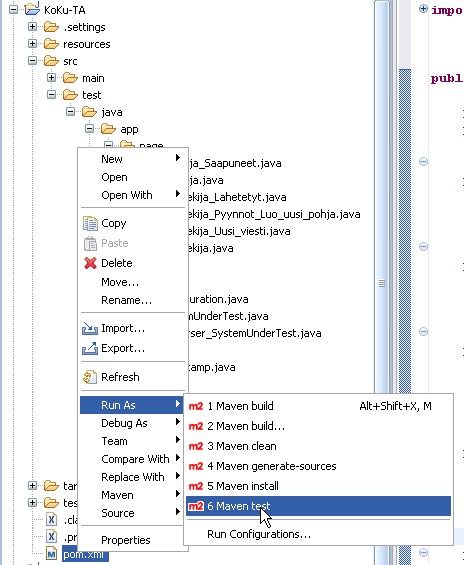
* <http://download.eclipse.org/technology/m2e/releases>

Lisäohjeet – Katso kuvat alla

* [m2eclipse plug-in asennus](#m2eclipse_asennus)
* [maven käyttöesimerkki Eclipse kehitysympäristössä](#maven_Eclipse_kaytto)



*Kuva-3:* m2eclipse plug-in asennus asennus Eclipse ympäristössä



*Kuva-4:* maven työkalun käyttöesimerkki Eclipse ympäristössä (m2eclipse plug-in asennuksen jälkeen)

#### TestNG Eclipse plug-in asennus

Kyseinen komponentti mahdollistaa TestNG testien suorittamisen Eclipse kehitys­ympäristössä.

*Lisätiedot:*

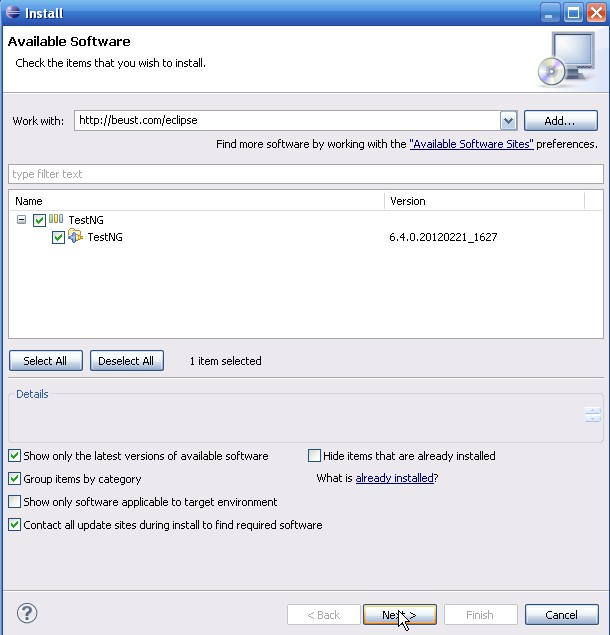
* <http://testng.org/doc/eclipse.html>

*Eclipse asennuksessa tarvittava osoite (Oletus: käytössä 3.4. tai uudempi Eclipse IDE versio):*

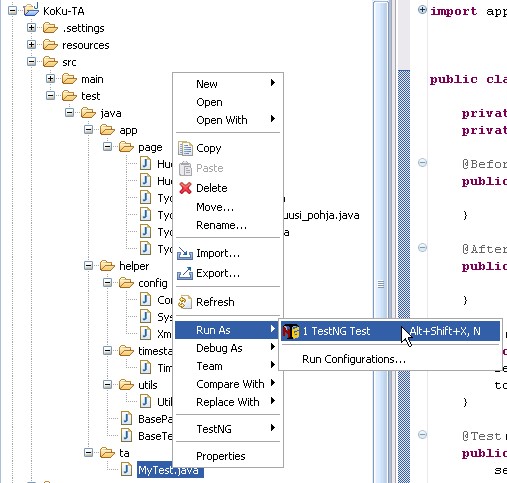
* <http://beust.com/eclipse>

*Lisäohjeet – Katso kuva alla:*

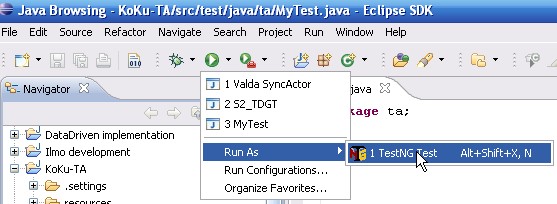
* [TestNG Eclipse plug-in asennus](#TestNG_Eclipse_asennus)
* [TestNG testiajon suoritus Eclipse ympäristössä (käyttöesimerkki 1)](#TestNG_testiajon_esim_1)
* [TestNG testiajon suoritus Eclipse ympäristössä (käyttöesimerkki 2)](#TestNG_testiajon_esim_2)



*Kuva-5:* TestNG Eclipse plug-in asennus Eclipse ympäristössä



*Kuva-6:* TestNG testiajon suoritus Eclipse ympäristössä (käyttöesimerkki 1)



*Kuva-7:* TestNG testiajon suoritus Eclipse ympäristössä (käyttöesimerkki 2)

# Testiautomaatioratkaisun käyttö

## Kohdeympäristön konfigurointi

Kohdeympäristön osoitteet on määritelty seuraavassa tiedostossa:

* ui\resources\config\_SystemUnderTest.xml

## Testiajon suoritus Maven työkalulla

Ajettavat testit määritellään seuraavalla TestNG konfigurointitiedostolla:

* ui\resources\testng.xml

*Lisätiedot:*

* [http://testng.org](http://testng.org/)

### Testiajon konfigurointi – ui\resources\testng.xml

### Testiajon käynnistys

Suoritusohjeet:

* Avataan konsoli (cmd.exe)
* Siirrytään paikalliseen ui –hakemistoon (cd komennolla)
* Annetaan seuraava Maven komento:
  + mvn clean test

### Testituloksien raportointi (Maven testiajon yhteydessä)

Testiajon tulokset generoidaan html -muodossa seuraavaan hakemistoon

* ui\target\surefire-reports\html

*Lisäohjeet:*

* Siirrytään raporttihakemistoon ja avataan seuraava tiedosto:
  + index.html

## Testiajon suoritus Eclipse kehitysympäristössä

### Testiluokan ajo - *Run As -> TestNG Test*

Suoritusohjeet:

* Ajettava esimerkki testiluokka
  + koku-ta-ui/src/test/java/ta/MyTest.java
* Java testiluokka voidaan ajaa seuraavasti:
  + Run as -> TestNG Test

# Java toteutuksen rakenne

Toteutus löytyy seuraavan polun alla:

* src/test/java
* Ratkaisuun sisältyvät *Java package* kansiot on esitetty alla olevassa taulukossa

|  |  |
| --- | --- |
| Java package | Sisältö |
| fi.koku.ta | MyTest luokka sisältää esimerkki testitapaukset |
| fi.koku.ta.app.page | Tyontekija ja Huoltaja luokat (sovelluksen käyttäjäluokat) |
| fi.koku.ta.app.page.viestit | Sisältää Java luokat eri tyyppisille viesteille (esim. lähetys, saapuneet ja uusi viesti) |
| fi.koku.ta.app.page.pyynnot | Sisältää Java luokat pyynnöille (esim. Luo uusi pyyntö) |
| fi.koku.ta.helper | BasePage ja BaseTest luokat (geneeriset luokat, joiden ominaisuuksia laajennetaan omilla/uusilla luokilla |
| fi.koku.ta.helper.config | Konfigurointirajapinta (hyvin alkeellinen tässä vaiheessa) |
| fi.koku.ta.helper.timestamp | Apufunktiot, joilla generoidaan aikaleimoja eri tarkoituksiin (esim kuvankaappauksen tiedostonimen osaksi) |
| fi.koku.ta.helper.utils | Utils luokka sisältää hyödyllisiä apufunktioita, jotka ovat suoraan käytettävissä BasePage ja BaseTest luokkien kautta. |

# Nykytoteutuksen puutteet

Esimerkkitestitapaukset eivät testiautomaatioratkaisun kehitysvaiheessa ole toimineet kovinkaan luotettavasti. Luotettavuutta on paranneltu työn edetessä, mutta toteutus kaipaa todennäköisesti jatkokehitystä.

Luotettavuutta parannetaan tarkistuksia lisäämällä sekä käyttöliittymäkomponenttien tunnistustapaa (esim xpath lausekkeita) kehittämällä.

# Vinkkejä jatkokehitykseen

Käyttöliittymä komponenttien käsittelyyn (navigointi, tiedon syöttö, toiminnon valinta,...) liittyvät asiat kannattaa toteuttaa keskitetysti (sisällytetään fi.koku.ta.app.page pakettihierarkiaan).

Lisätarkistukset (assert kutsut) kuuluu toteuttaa testiluokkaan.