

# openCRX Installation Guide for Tomcat 6

Version 2.4



[www.opencrx.org](http://www.opencrx.org)

---

## **License**

The contents of this file are subject to a BSD license (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.opencrx.org/license.htm>

**Copyright** 2009 © CRIXP Corp. All rights reserved.



## Table of Contents

<b>1</b>	<b>About this Book.....</b>	<b>4</b>
1.1	Who this book is for.....	4
1.2	What do you need to understand this book.....	4
1.3	Tips, Warnings, etc.....	4
<b>2</b>	<b>Prerequisites.....</b>	<b>5</b>
2.1	JDK 5.0.....	5
2.2	Ant 1.7.0.....	6
2.3	openMDX SDK.....	6
2.4	openCRX SDK.....	6
2.5	Apache Tomcat.....	7
2.6	Database.....	8
<b>3</b>	<b>Installing openCRX for Tomcat.....</b>	<b>9</b>
<b>4</b>	<b>Starting Tomcat.....</b>	<b>14</b>
4.1	Initial Setup of openCRX Application.....	15
4.1.1	First Login as "admin-Root".....	16
4.1.2	Create Data Segment "Standard".....	19
4.1.3	Import Codes and Data.....	20
4.1.4	Set Access Levels of Codes.....	21
4.1.5	Restarting Tomcat.....	21
4.2	Initial Setup of Segment "Standard".....	22
4.2.1	First Login as "admin-Standard".....	22
4.2.2	Configure Segment "Standard".....	23
4.2.3	Create a new Contact "guest".....	24
4.2.4	Create a new User "guest".....	27
4.2.5	Login as "guest".....	30
<b>5</b>	<b>Next Steps.....</b>	<b>31</b>

## List of Figures

Figure 1:	openCRX Login page.....	15
Figure 2:	First login with admin-Root / opencrx.....	16
Figure 3:	Start screen of admin-Root.....	18
Figure 4:	Create a new segment with Actions > Create Administrator.....	19
Figure 5:	Create a new segment named Standard.....	19
Figure 6:	Result of operation Create Administrator.....	19
Figure 7:	Load Code Tables and Data – Step 1.....	20
Figure 8:	Load Code Tables and Data – Step 2.....	20
Figure 9:	Execute Operation Security > Set Access Level.....	21
Figure 10:	Parameters of Operation Set Access Level.....	21

Figure 11: Result of the Operation Set Access Level.....	21
Figure 12: First login with admin-Standard / opencrx.....	22
Figure 13: As admin-Standard start the wizard Segment Setup.....	23
Figure 14: Wizard Segment Setup - properly configured Segment.....	23
Figure 15: Refresh homepage of user admin-Standard.....	24
Figure 16: Fly-in menu.....	24
Figure 17: New contact guest – step 1.....	25
Figure 18: New contact guest – step 2.....	26
Figure 19: New contact guest – step 3.....	26
Figure 20: New contact guest – step 4.....	26
Figure 21: New user guest – step 1.....	27
Figure 22: New user guest – step 2.....	27
Figure 23: New user guest – step 3.....	27
Figure 24: New user guest – step 4.....	28
Figure 25: New user guest – step 4.....	28
Figure 26: New user guest – step 5.....	28
Figure 27: New user guest – step 6.....	29

## List of Listings

Listing 1: The file openejb.xml.....	10
Listing 2: The file server.log.properties.....	11
Listing 3: The file run.bat.....	11
Listing 4: The file run.sh.....	12
Listing 5: Connector definition in server.xml.....	12
Listing 6: Access Log Valve in server.xml.....	13
Listing 7: The file tomcat-users.xml.....	13
Listing 8: Tomcat console output.....	14
Listing 9: Console Output – Initialization of the openCRX servlet.....	17
Listing 10: Console Output – Importing Codes and Data.....	20

# 1 About this Book

openCRX is the leading open source CRM enterprise suite. openCRX is based on the openMDX application framework, an open source application framework based on the OMG's model driven architecture (MDA) standards, delivering maximum openness, standards compliance and a state-of-the-art component-based architecture.

This book describes the manual installation of openCRX for deployment on Apache Tomcat.

## 1.1 Who this book is for

The intended audience are openCRX administrators and system administrators who either want to install openCRX manually (for whatever reason) or want to install openCRX on a platform not supported by our openCRX Server installer.

## 1.2 What do you need to understand this book

This book describes the manual installation of openCRX for deployment on Apache Tomcat. The book assumes that you are familiar with the Tomcat management and administration.

## 1.3 Tips, Warnings, etc.

We make use of the following pictograms:



Tip

Information provided as a "Tip" might be helpful for various reasons: time savings, risk reduction, etc.



Important

You should carefully read information marked with "Important". Ignoring such information is typically not a good idea.



Warning

Warnings should not be ignored (risk of data loss, etc.)

## 2 Prerequisites

As a first step select the openCRX version you want to install. Based on the published version compatibility information you can determine the appropriate versions of openMDX, Tomcat, and Java JDK:

<http://www.opencrx.org/faq.htm#versioncompatibility>



Write down the version numbers of the software packages you have chosen to install – this may be helpful in the future in case you require support or want to file a bug report:

openCRX v \_\_\_\_\_ (this guide is based on openCRX v2.4.1)  
openMDX v \_\_\_\_\_ (this guide is based on openCRX v2.4.1)  
Tomcat v \_\_\_\_\_ (this guide is based on Tomcat v6.0.18)  
JDK v \_\_\_\_\_ (this guide is based on Sun's JDK 1.5.17)



Throughout the installation of openCRX and related software it is a good idea to **avoid paths containing blanks** like the default installation directory ...\\Program Files\\... on Windows.

Whenever an installer proposes an installation path containing blanks, please change the path to avoid problems down the road.

### 2.1 JDK 5.0

Install the Sun Java JDK 5.0 available from the following site:

Sun JDK 5.0:

[http://java.sun.com/javase/downloads/index\\_jdk5.jsp](http://java.sun.com/javase/downloads/index_jdk5.jsp)



It is not sufficient to have a Java Runtime Environment (JRE) only. The full-blown JDK is required to run openCRX.



Don't forget to set the **environment variable** `JAVA_HOME`

## 2.2 Ant 1.7.0

Download Ant 1.7.0 (available from <http://ant.apache.org/>) for your platform and install it by expanding the downloaded file to a directory of your choice.



Ant v1.7.1 might work as well. However, we recommend Ant v1.7.0 (there are some combinations of JDK versions and Ant v1.7.1 that “don't like each other”...)



Don't forget to set the **environment variables** `ANT_HOME` and `ANT_OPTS` as follows:

- **ANT\_HOME** should point to the installation directory of Ant, e.g. `D:\apache-ant-1.7.0` on Windows or `/opt/apache-ant-1.7.0` on Linux.
- **ANT\_OPTS** should be set to `-Xmx512m` so that enough memory can be allocated to compile openCRX during the installation of the openCRX SDK.

## 2.3 openMDX SDK

Download and install the **openMDX SDK**: <http://www.openmdx.org/sdk.htm>  
Installation instructions are also available from the above website.



If your platform is not supported by the openMDX SDK Installer, you can install the openMDX SDK on one of the supported platforms and then copy the required files to your target platform.

## 2.4 openCRX SDK

Download and install the **openCRX SDK**: <http://www.opencrx.org/sdk.htm>  
Installation instructions are also available from the above website.



If your platform is not supported by the openCRX SDK Installer, you can install the openCRX SDK on one of the supported platforms and then copy the required files to your target platform.

## 2.5 Apache Tomcat

Download **Tomcat 6.0.18** from

<http://archive.apache.org/dist/tomcat/tomcat-6/v6.0.18/bin/>

This guide is based on Tomcat v6.0.18:

<http://archive.apache.org/dist/tomcat/tomcat-6/v6.0.18/bin/apache-tomcat-6.0.18.zip> (Windows)

<http://archive.apache.org/dist/tomcat/tomcat-6/v6.0.18/bin/apache-tomcat-6.0.18.tar.gz> (Linux)



If you want to deploy openCRX on a Tomcat version different from Tomcat v6.0.18, we recommend to build the files **tomcat-juli.jar** and **catalina.jar** matching to your Tomcat version as follows:

- download the desired Tomcat binary, e.g. **apache-tomcat-6.0.xx.zip** (or `apache-tomcat-6.0.xx.tar.gz` on Linux) and expand it to a directory of your choice, e.g. `D:\apache-tomcat-6.0.xx` on Windows or `/opt/apache-tomcat-6.0.xx` on Linux
- download and install the **openMDX SDK** (see <http://www.openmdx.org/sdk.htm>)
- **copy all the files** from the directory `TOMCAT_HOME\lib` to the openMDX SDK directory `<MDX_SDK_Install_Dir>\openmdx-2.4.1\apache\jre-1.5\tomcat\lib` (overwrite existing files)
- copy the file **tomcat-juli.jar** from the directory `TOMCAT_HOME\bin` to the openMDX SDK directory `<MDX_SDK_Install_Dir>\openmdx-2.4.1\apache\jre-1.5\tomcat\lib` (overwrite the existing file)
- open a shell and change to the directory `<MDX_SDK_Install_Dir>\openmdx-2.4.1\core`
- run **ant clean** and then run **ant deliverables**
- the above will build various files in the directory `<MDX_SDK_Install_Dir>\openmdx-2.4.1\jre-1.5\core\lib`
- copy the file **tomcat-juli.jar** to `TOMCAT_HOME\bin`
- open a shell and change to the directory `<MDX_SDK_Install_Dir>\openmdx-2.4.1\openejb`
- run **ant clean** and then run **ant deliverables**
- the above will build various files in the directory `<MDX_SDK_Install_Dir>\openmdx-2.4.1\openejb-3\openejb\lib`
- copy the file **catalina.jar** to `TOMCAT_HOME\lib`

## 2.6 Database

Please note that you **must set up the openCRX database** as described in the respective openCRX database installation guide **before you continue**. For example, if you want to install openCRX for PostgreSQL, you must first install PostgreSQL and the matching openCRX database definitions.

Database installation guides for the supported database management systems are available from <http://www.opencrx.org/documents.htm>



Please remember that you will also need a **JDBC driver** appropriate for your database management system (please refer to the relevant DB installation guide for additional information).

Copy the JDBC driver into the directory `TOMCAT_HOME\lib`.



Please note that this guides assumes that you installed PostgreSQL for DB-specific instructions. If you decided for another DBMS, you might have to adapt some of the commands/instructions accordingly.

Once you have successfully installed the database you are ready to continue with the Tomcat setup.



### 3 Installing openCRX for Tomcat

In a first step, install Apache Tomcat by expanding the downloaded file **apache-tomcat-6.0.18.zip** (or **apache-tomcat-6.0.18.tar.gz** respectively) to a directory of your choice, e.g. `D:\apache-tomcat-6.0.18` on Windows or `/opt/apache-tomcat-6.0.18` on Linux. For the remainder of this document we assume that `TOMCAT_HOME` points to the Tomcat installation directory, e.g. **`TOMCAT_HOME=D:\apache-tomcat-6.0.18`**



Tip

On Windows, avoid paths that contain blanks (e.g. the famous `C:\Program File\...`) as there are all kinds of pitfalls down the road...



Important

Make sure that you added `JAVA_HOME` to your system environment variables as Tomcat needs the JDK to compile JSPs, e.g. `JAVA_HOME=D:\java\jdk1.5.0` on Windows or `JAVA_HOME=/opt/java/jdk1.5.0` on Linux.

Now we **add openMDX and Apache OpenEJB** to Tomcat by copying several files from the openCRX SDK to the Tomcat directory structure:

- copy the file **tomcat-juli.jar** from the openMDX SDK installation directory `<MDX_SDK_Install_Dir>\openmdx-2.4.1\jre-1.5\core\lib` to the Tomcat directory `TOMCAT_HOME\bin` (overwrite the existing file)
- copy the file **catalina.jar** from the openMDX SDK installation directory `<MDX_SDK_Install_Dir>\openmdx-2.4.1\openejb-3\openejb\lib` to the Tomcat directory `TOMCAT_HOME\lib` (overwrite the existing file)



Important

Please note that the files `tomcat-juli.jar` and `catalina.jar` are version-specific, i.e. if you want to run openCRX on a Tomcat version different from Tomcat v6.0.18 you need to go back to chapter 2.5 Apache Tomcat and build them.

- copy the two files **openmdx-base.jar** and **openmdx-base-ext.jar** from the openMDX SDK installation directory `<MDX_SDK_Install_Dir>\openmdx-2.4.1\jre-1.5\core\lib` to the Tomcat directory `TOMCAT_HOME\lib`
- open the ZIP file `<MDX_SDK_Install_Dir>\openmdx-2.4.1\repository\distribution\openmdx-2.4.1-openejb.openejb-3.zip` and copy the file **openmdx-openejb.war** - located in the folder `openmdx-2.4.1/openejb-3/openejb/deployment-unit` - to the Tomcat directory `TOMCAT_HOME\webapps` and then rename it to **0-openmdx-openejb.war** (Tomcat must load it as the first WebApp)

Next we install the **openCRX enterprise archive**:

- create the directory `TOMCAT_HOME\apps`
- copy the file **opencrx-core-CRX.ear** from the directory `<SDK_Install_Dir>\opencrx-2.4.1\jre-1.5\core\deployment-unit` to to the Tomcat directory `TOMCAT_HOME\apps`

Next we configure the database connection (and optionally a mail resource):

- create a new file `TOMCAT_HOME\conf\openejb.xml` and open it with an editor (Notepad, kate, etc.); paste the following text into the file:

*Listing 1: The file openejb.xml*

```
<?xml version="1.0" encoding="UTF-8"?>
<openejb>

<Container id="My Stateless Container" type="STATELESS">
  Timeout 0
  PoolSize 10
  StrictPooling true
</Container>

<!-- openCRX with HSQLDB, DB: CRX_CRX -->
<Resource id="jdbc_opencrx_CRX" type="DataSource">
  JdbcDriver org.hsqldb.jdbcDriver
  JdbcUrl jdbc:hsqldb:hsq://127.0.0.1:9001/CRX_CRX
  UserName sa
  Password manager99
  JtaManaged true
</Resource>

<Resource id="mail/provider/CRX" type="javax.mail.Session">
  mail.debug true
  mail.transport.protocol smtp
  mail.smtp.user myusername
  mail.smtp.password mypassword
  mail.smtp.starttls.enable true
  mail.smtp.auth true
  mail.smtp.host myhost
  mail.smtp.port 25
  mail.smtp.user myuseruname
  mail.from mymailfrom
  mail.store.protocol pop3s
  mail.pop3s.host myhost
  mail.pop3s.port 995
  mail.pop3s.auth true
  mail.pop3s.user myusername
  mail.pop3s.password mypassword
</Resource>

<Deployments dir="apps/" />

</openejb>
```

See <http://openejb.apache.org/configuration.html> for additional details.

- adapt the values of **UserName**, **Password** and the **JdbcUrl** to your own environment and then save the file (see also table below)

The following table might be helpful to configure your database connection:

<b>HSQLDB 1</b>	org.hsqldb.jdbcDriver jdbc:hsqldb:hsq://127.0.0.1:9001/CRX_CRX
<b>MySQL 5</b>	com.mysql.jdbc.Driver jdbc:mysql://127.0.0.1:3306/CRX_CRX
<b>PostgreSQL 8</b>	org.postgresql.Driver jdbc:postgresql://127.0.0.1:5432/CRX_CRX
<b>MS SQL 2005</b>	com.microsoft.sqlserver.jdbc.SQLServerDriver jdbc:sqlserver://127.0.0.1:1433;databaseName=CRX_CRX;selectMethod=cursor
<b>DB/2 9</b>	com.ibm.db2.jcc.DB2Driver jdbc:db2://127.0.0.1:50000/CRX_CRX
<b>Oracle 10</b>	oracle.jdbc.driver.OracleDriver jdbc:oracle:thin:@127.0.0.1:1521:XE



Verify all the relevant information for correctness, e.g. ensure that the **JdbcUrl** – in particular the **database name** – the **UserName**, and the **Password** match with your installation.

A simple copy/paste of the above sample file will typically not work, i.e. it is expected that you will have to adapt some of the parameters.

We also need to configure **openCRX logging**:

- copy the file **slf4j-openmdx1.jar** from the openMDX SDK installation directory `<MDX_SDK_Install_Dir>\openmdx-2.4.1\jre-1.5\log\lib` to the Tomcat directory `TOMCAT_HOME\lib`
- create the file **server.log.properties** with the following content in the Tomcat installation directory `TOMCAT_HOME`:

*Listing 2: The file server.log.properties*

```
ApplicationId = openCRX
LogFileExtension = log
LogFilePath = D:/apache-tomcat-6.0.18/logs/
LogLevel = warning
java.LoggingMechanism = SharedDatedFileLoggingMechanism
LogFormat = ${logger}|${time}|${level}|${logsource}|${host}|${thread}|${class}|${method}|${line}|${summary}|${detail}
```



Adapt `D:/apache-tomcat-6.0.18` to your environment.

Notice the slashes ("`/`") in the path – they are forward slashes on all platforms!

Next we create a **batch file to start Apache Tomcat** with all the appropriate JVM options:

- **on Windows**: create the file **run.bat** with the following content in the Tomcat directory `TOMCAT_HOME\bin`:

*Listing 3: The file run.bat*

```
RMDIR /S /Q D:\apache-tomcat-6.0.18\temp
MKDIR D:\apache-tomcat-6.0.18\temp
RMDIR /S /Q D:\apache-tomcat-6.0.18\work
MKDIR D:\apache-tomcat-6.0.18\work
RMDIR /S /Q D:\apache-tomcat-6.0.18\maildir

set JAVA_HOME=D:\Java\jdk1.5.0

set JAVA_OPTS=-Xmx800M
set JAVA_OPTS=%JAVA_OPTS% -Djava.protocol.handler.pkgs=org.openmdx.kernel.url.protocol
set JAVA_OPTS=%JAVA_OPTS% -Dorg.openmdx.log.config.filename=D:\apache-tomcat-6.0.18\server.log.properties
set JAVA_OPTS=%JAVA_OPTS% -Dorg.opencrx.maildir=D:\apache-tomcat-6.0.18\maildir

set JAVA_OPTS=%JAVA_OPTS% -Dopencrx.CRX.jdbc.driverName="org.postgresql.Driver"
set JAVA_OPTS=%JAVA_OPTS% -Dopencrx.CRX.jdbc.url="jdbc:postgresql://localhost:5432/CRX_CRX"
set JAVA_OPTS=%JAVA_OPTS% -Dopencrx.CRX.jdbc.userName="sa"
set JAVA_OPTS=%JAVA_OPTS% -Dopencrx.CRX.jdbc.password="manager99"
REM JAVA_OPTS=%JAVA_OPTS% -Dorg.openmdx.persistence.jdbc.useLikeForOidMatching=false

catalina.bat run
```



- adapt `D:\apache-tomcat-6.0.18` to your environment
- adapt `D:\Java\jdk1.5.0` to your environment

- adapt the options related to the database connection to your environment (`driverName`, `url`, `userName`, and `password`)
- if you are using PostgreSQL and if you set up the database as proposed by us, you can also uncomment the JVM option **`useLikeForOidMatching=false`** for enhanced performance.
- **on Linux**: create the file `run.sh` with the following content in the Tomcat directory `TOMCAT_HOME\bin`:

*Listing 4: The file run.sh*

```
#!/bin/sh
export CATALINA_HOME=/opt/apache-tomcat-6.0.18
export JAVA_HOME=/opt/Java/jdk1.5.0
export JAVA_OPTS="-Xmx800M -XX:MaxPermSize=128m -Dtomcat.server.port=8005"
export JAVA_OPTS="$JAVA_OPTS -Djava.protocol.handler.pkgs=org.openmdx.kernel.url.protocol"
export JAVA_OPTS="$JAVA_OPTS -Dorg.openmdx.log.config.filename=$CATALINA_HOME/server.log.properties"
export JAVA_OPTS="$JAVA_OPTS -Dorg.opencrx.maildir=$CATALINA_HOME/maildir"
export JAVA_OPTS="$JAVA_OPTS -Dopencrx.CRX.jdbc.driverName="org.postgresql.Driver"
export JAVA_OPTS="$JAVA_OPTS -Dopencrx.CRX.jdbc.url="jdbc:postgresql://localhost:5432/CRX_CRX"
export JAVA_OPTS="$JAVA_OPTS -Dopencrx.CRX.jdbc.userName="sa"
export JAVA_OPTS="$JAVA_OPTS -Dopencrx.CRX.jdbc.password="manager99"
cd $CATALINA_HOME
rm -Rf temp
mkdir temp
rm -Rf work
./bin/catalina.sh run
```



- adapt `/opt/apache-tomcat-6.0.18` to your environment
- adapt `/opt/Java/jdk1.5.0` to your environment
- adapt the options related to the database connection to your environment (`driverName`, `url`, `userName`, and `password`)
- if you are using PostgreSQL and if you set up the database as proposed by us, you can also add the following JVM option for enhanced performance:
  - Dorg.openmdx.persistence.jdbc.useLikeForOidMatching=false

Next, we have to make a few changes to Tomcat's **server.xml**:

- open the file `TOMCAT_HOME\conf\server.xml`
- look for the connector definition `<Connector port="8080" protocol="HTTP/1.1"` and add the option `URIEncoding="UTF-8"` so that the connector definition looks as shown below:

*Listing 5: Connector definition in server.xml*

```
<Connector port="8080" protocol="HTTP/1.1"
  connectionTimeout="20000"
  redirectPort="8443"
  URIEncoding="UTF-8"
 />
```



Tomcat can create Apache-style access log files as follows: in the file `TOMCAT_HOME\conf\server.xml`, look for the engine definition `<Engine name="Catalina" defaultHost="localhost">` and then add the following valve section just below:

*Listing 6: Access Log Valve in server.xml*

```
...
<Engine name="Catalina" defaultHost="localhost">

<Valve className="org.apache.catalina.valves.AccessLogValve"
  directory="logs" prefix="combined_access_log."
  suffix=".txt" pattern="combined" resolveHosts="false" />
...
```

As a final step, you must activate security for the openCRX application.

- open the file `TOMCAT_HOME\conf\tomcat-users.xml` and add users and roles as follows:

*Listing 7: The file tomcat-users.xml*

```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <role rolename="OpenCrxAdministrator"/>
  <role rolename="OpenCrxUser"/>
  <role rolename="OpenCrxRoot"/>
  <role rolename="manager"/>
  <user username="admin-Root" password="opencrx" roles="OpenCrxRoot"/>
  <user username="admin-Standard" password="opencrx" roles="OpenCrxAdministrator,OpenCrxUser"/>
  <user username="system" password="manager99" roles="manager"/>
  <user username="guest" password="opencrx" roles="OpenCrxUser"/>
</tomcat-users>
```



- adapt the `passwords` to your liking



We recommend that you stay with the file-based authentication for all users until you have finished installing openCRX. You avoid situations where you have to trouble-shoot multiple issues at the same time...



It is recommended that you stay with the file-based authentication for the user **admin-Root**. This simplifies the openCRX bootstrapping.



Just a quick reminder before you move on to starting Tomcat. Please remember to copy the **JDBC driver** appropriate for your database management system to the Tomcat directory `TOMCAT_HOME\lib` (refer to the relevant DB installation guide for additional information).

## 4 Starting Tomcat

You are now ready to start Tomcat. Open a command shell and start run.bat (or run.sh on Linux). You should verify whether the start options match the ones described earlier:

### Listing 8: Tomcat console output

```
[12:11:07.72]C:\apache-tomcat-6.0.xx\bin>run
[12:11:13.04]C:\apache-tomcat-6.0.xx\bin>RMDIR /S /Q C:\apache-tomcat-6.0.xx\temp
[12:11:13.08]C:\apache-tomcat-6.0.xx\bin>MKDIR C:\apache-tomcat-6.0.xx\temp
[12:11:13.09]C:\apache-tomcat-6.0.xx\bin>RMDIR /S /Q C:\apache-tomcat-6.0.xx\work
[12:11:13.23]C:\apache-tomcat-6.0.xx\bin>MKDIR C:\apache-tomcat-6.0.xx\work
[12:11:13.23]C:\apache-tomcat-6.0.xx\bin>RMDIR /S /Q C:\apache-tomcat-6.0.xx\maildir
[12:11:13.23]C:\apache-tomcat-6.0.xx\bin>set JAVA_HOME=D:\Java\jdk1.5.0
[12:11:13.23]C:\apache-tomcat-6.0.xx\bin>set JAVA_OPTS=-Xmx800M
-Djava.protocol.handler.pkgs=org.openmdx.kernel.url.protocol
-Dorg.openmdx.log.config.filename=C:\apache-tomcat-6.0.xx\server.log.properties
-Dorg.openmdx.maildir=C:\apache-tomcat-6.0.xx\maildir
-Dopencrx.CRX.jdbc.driverName="org.postgresql.Driver"
-Dopencrx.CRX.jdbc.url="jdbc:postgresql://localhost:5432/CRX_CRX"
-Dopencrx.CRX.jdbc.userName="system"
-Dopencrx.CRX.jdbc.password="manager99"
-Dorg.openmdx.persistence.jdbc.useLikeForOidMatching=false
[12:11:13.23]C:\apache-tomcat-6.0.xx\bin>catalina.bat run
Using CATALINA_BASE:   C:\apache-tomcat-6.0.xx
Using CATALINA_HOME:   C:\apache-tomcat-6.0.xx
Using CATALINA_TMPDIR: C:\apache-tomcat-6.0.xx\temp
Using JRE_HOME:        D:\Java\jdk1.5.0
Mar 16, 2009 12:11:16 PM org.apache.catalina.core.AprLifecycleListener init
INFO: The APR based Apache Tomcat Native library which allows optimal performance in production environments was not found
Mar 16, 2009 12:11:16 PM org.apache.coyote.http11.Http11Protocol init
INFO: Initializing Coyote HTTP/1.1 on http-8080
Mar 16, 2009 12:11:16 PM org.apache.catalina.startup.Catalina load
INFO: Initialization processed in 1173 ms
Mar 16, 2009 12:11:16 PM org.apache.catalina.core.StandardService start
INFO: Starting service Catalina
Mar 16, 2009 12:11:16 PM org.apache.catalina.core.StandardEngine start
INFO: Starting Servlet Engine: Apache Tomcat/6.0.18
OpenEJB init-params:
Apache OpenEJB 3.1      build: 20081009-03:31
http://openejb.apache.org/
DEBUG: JavaMail version 1.4.1
DEBUG: not loading file: D:\Java\jdk1.5.0\jre\lib\javamail.providers
DEBUG: java.io.FileNotFoundException: D:\Java\jdk1.5.0\jre\lib\javamail.providers (The system cannot find the file specified)
DEBUG: !anyLoaded
DEBUG: not loading resource: /META-INF/javamail.providers
DEBUG: successfully loaded resource: /META-INF/javamail.default.providers
DEBUG: Tables of loaded providers
DEBUG: Providers Listed By Class Name: {com.sun.mail.smtp.SMTPSSLTransport=javax.mail.Provider[TRANSPORT,...]}
DEBUG: Providers Listed By Protocol: {imaps=javax.mail.Provider[STORE,imaps,com.sun.mail.imap.IMAPSSLStore,Sun Microsystems, Inc]...}
DEBUG: successfully loaded resource: /META-INF/javamail.default.address.map
DEBUG: !anyLoaded
DEBUG: not loading resource: /META-INF/javamail.address.map
DEBUG: not loading file: D:\Java\jdk1.5.0\jre\lib\javamail.address.map
DEBUG: java.io.FileNotFoundException: D:\Java\jdk1.5.0\jre\lib\javamail.address.map (The system cannot find the file specified)
context path = /0-openmdx-openejb
context path = /docs
context path = /examples
context path = /host-manager
context path = /manager
context path =
Mar 16, 2009 12:12:03 PM org.apache.coyote.http11.Http11Protocol start
INFO: Starting Coyote HTTP/1.1 on http-8080
Mar 16, 2009 12:12:03 PM org.apache.jk.common.ChannelSocket init
INFO: JK: ajp13 listening on /0.0.0.0:8009
Mar 16, 2009 12:12:03 PM org.apache.jk.server.JkMain start
INFO: Jk running ID=0 time=0/31 config=null
Mar 16, 2009 12:12:03 PM org.apache.catalina.startup.Catalina start
INFO: Server startup in 46797 ms
```



Please note that as long as Tomcat reports errors/exceptions during this startup phase you must investigate and fix them before you continue. You might want to inspect the log files in the Tomcat directory `TOMCAT_HOME\logs` to find out what went wrong.

Once Tomcat starts without errors you are ready to continue with the initial setup of the openCRX application.

## 4.1 Initial Setup of openCRX Application

Connect to the login page of openCRX. The exact URL depends on your deployment details. For a standard openCRX deployment on Tomcat running on your local machine and listening at port 8080 the URL is

**<http://localhost:8080/opencrx-core-CRX/Login>**

You should see the openCRX Login page as follows:

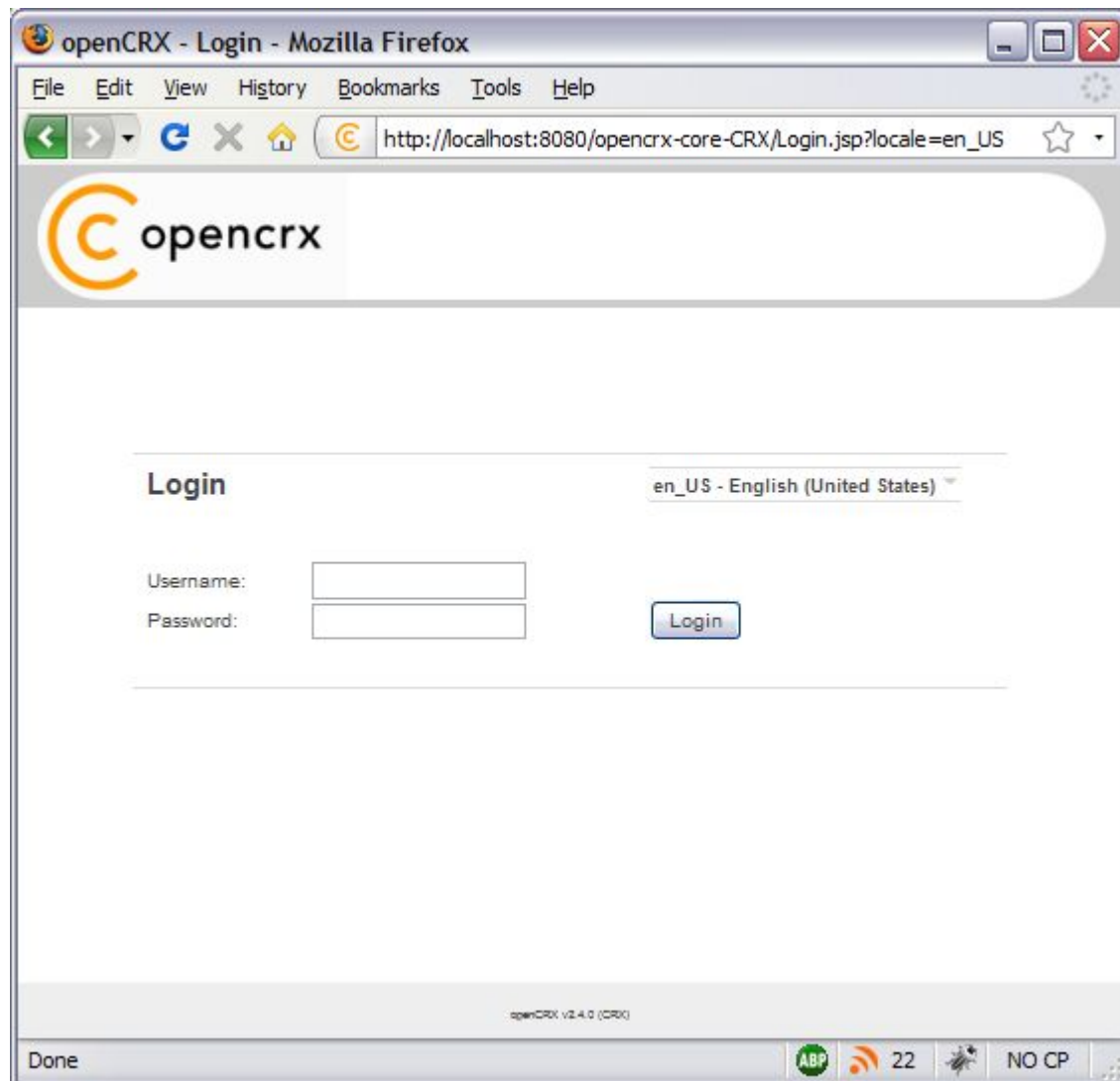


Figure 1: openCRX Login page



openCRX tries to determine the locale from your environment. You can change the locale at any time with the drop down on the login page.

### 4.1.1 First Login as “admin-Root”

Enter **admin-Root** into the field *Username* and then enter **opencrx** into the field *Password* (obviously, if you chose a different password when you created the file `tomcat-users.xml` you have to enter that password...):

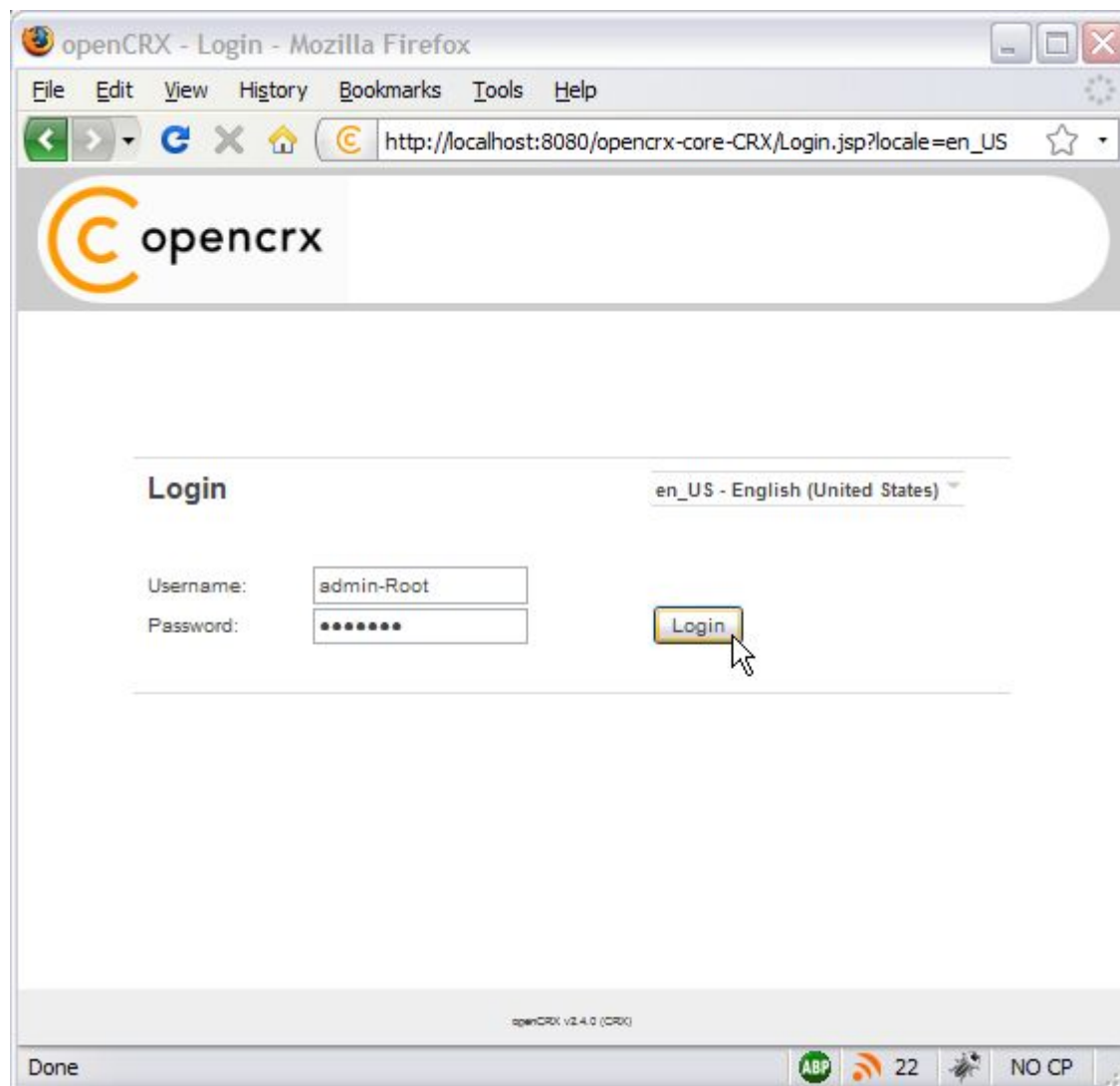


Figure 2: First login with admin-Root / opencrx

Click the button  to start the initialization process.



If you can't get past the login screen (with the correct Username and Password, of course) and the Warning “**Browser must accept cookies**” keeps showing up need to **verify your browser settings related to cookies**. If you have installed a **personal firewall** (e.g. ZoneAlarm, Comodo, etc.) you also need to verify the cookie settings of your firewall. **Tomcat must be able to create/set a session cookie.**



The openCRX servlet is initialized during this first login, i.e. don't stop/kill the browser if it takes a while. The application console output will look similar to the following listing:

### Listing 9: Console Output – Initialization of the openCRX servlet

```

Mon Mar 16 12:27:34 CET 2009: Login: requestURL=http://localhost:8080/openctx-core-CRX/Login.jsp;
  isRequestedSessionIdFromCookie=false; servletPath=/Login.jsp; remoteUser=null
Mon Mar 16 12:27:34 CET 2009: Login: Redirecting...
Mon Mar 16 12:27:34 CET 2009: Login: requestURL=http://localhost:8080/openctx-core-CRX/Login.jsp;
  isRequestedSessionIdFromCookie=true; servletPath=/Login.jsp; remoteUser=null
Mon Mar 16 12:27:35 CET 2009: Login: requestURL=http://localhost:8080/openctx-core-CRX/Login.jsp;
  isRequestedSessionIdFromCookie=true; servletPath=/Login.jsp; remoteUser=null
Inspecting /WEB-INF/config/ui/1-Easy/en_US
Inspecting /WEB-INF/config/ui/1-Easy/de_CH
Inspecting /WEB-INF/config/ui/1-Easy/es_MX
Inspecting /WEB-INF/config/ui/1-Easy/zh_CN
Inspecting /WEB-INF/config/ui/1-Easy/sv_SE
Inspecting /WEB-INF/config/ui/1-Easy/tr_TR
Inspecting /WEB-INF/config/ui/1-Easy/fa_IR
Inspecting /WEB-INF/config/ui/1-Easy/fr_FR
...
...
Creating org::openmdx::security::realm1/provider/CRX/segment/Root
Creating org::openmdx::security::realm1/provider/CRX/segment/Root/realm/Default
Creating org::openmdx::security::realm1/provider/CRX/segment/Root/realm/Default/principal/admin-Root
Creating org::openmdx::security::realm1/provider/CRX/segment/Root/realm/Default/principal/Administrators
Creating org::openmdx::security::realm1/provider/CRX/segment/Root/realm/Default/principal/Users
Creating org::openmdx::security::realm1/provider/CRX/segment/Root/realm/Root
Creating org::openmdx::security::realm1/provider/CRX/segment/Root/realm/Root/principal/Administrators
Creating org::openmdx::security::realm1/provider/CRX/segment/Root/realm/Root/principal/Users
Creating org::openmdx::security::realm1/provider/CRX/segment/Root/realm/Root/principal/admin-Root.User
Creating org::openmdx::security::realm1/provider/CRX/segment/Root/realm/Root/principal/admin-Root
Loading /WEB-INF/config/bootstrap/Root/103_security_authentication.xml
Storing 1 objects
Creating org::openmdx::security::authentication1/provider/CRX/segment/Root
Loading /WEB-INF/config/bootstrap/Root/200_code_segment.xml
Storing 1 objects
Creating org::openctx::kernel::code1/provider/CRX/segment/Root
Loading /WEB-INF/config/bootstrap/Root/300_admin_segment.xml
Storing 1 objects
Creating org::openctx::kernel::admin1/provider/CRX/segment/Root
Loading /WEB-INF/config/bootstrap/Standard/dummy.xml
Storing 0 objects
Done
Inspecting /WEB-INF/config/filters/
Loading /WEB-INF/config/filters/
Loaded filters #5229

```



If you see lots of error messages on the console and/or your log files grow very fast (several hundred KB if not more) then it is quite likely that your **DB connection** is not configured correctly. If your log file contains errors like **MEDIA\_ACCESS\_FAILURE** there is definitely something wrong with your DB connection (or your DB). Please note that there is absolutely no sense in continuing before you have fixed your DB connection. You might want to verify

- the spelling of the name of your database (including capitalization), e.g. `ctx-CRX` vs. `CRX_CRX`
- user name (e.g. `system`) and password (e.g. `manager99`)
- permissions of the user, etc.

After the successful startup of openCRX you see a screen similar to the one shown in the figure below. You should see the root objects Administration, Codes, Security Realm, Security Policies, and Security Subjects:

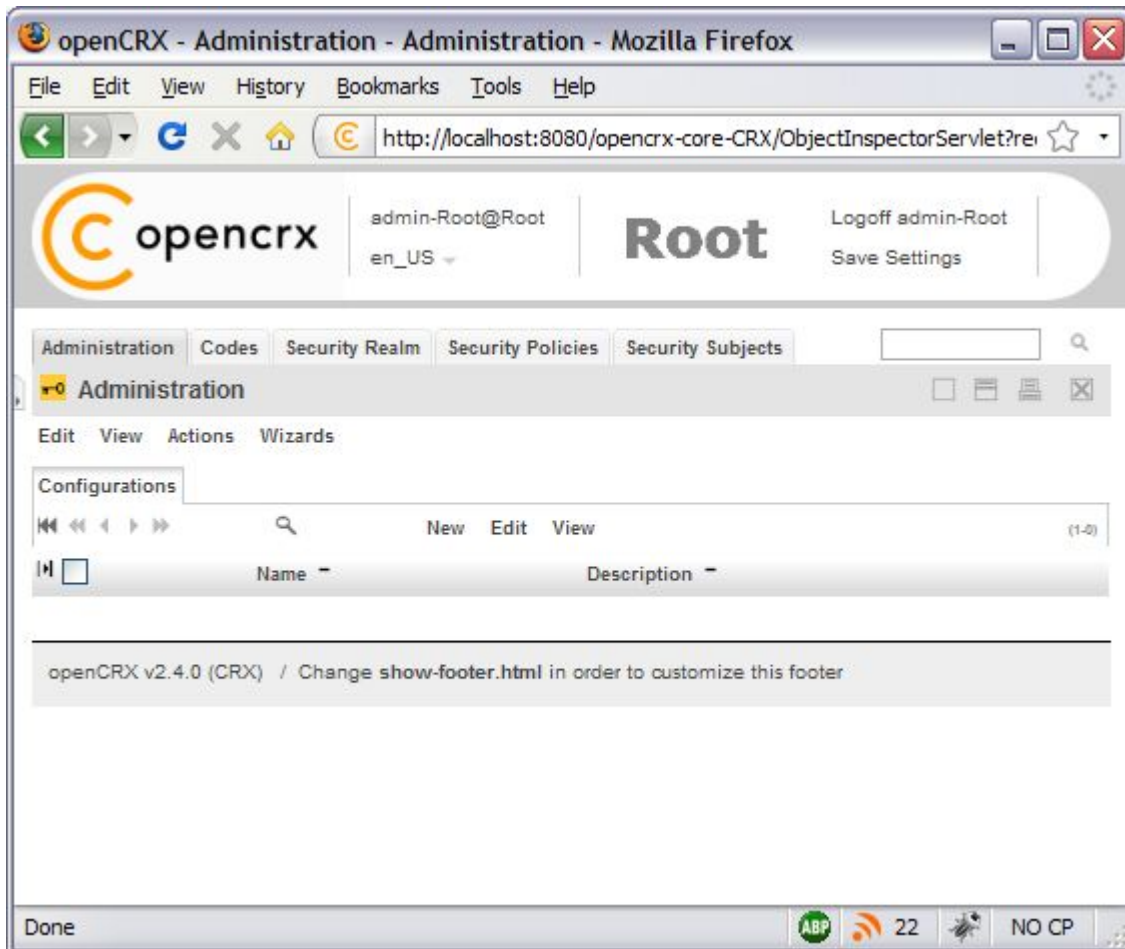


Figure 3: Start screen of admin-Root

## 4.1.2 Create Data Segment "Standard"

Execute the operation **Actions > Create Administrator** to create a new segment and selected default accounts (including an administrator's account which allows you to manage the newly created segment).

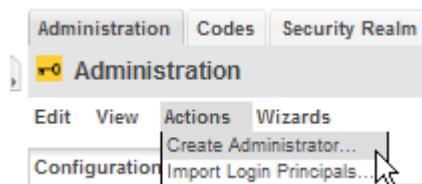


Figure 4: Create a new segment with Actions > Create Administrator

Set the field Segment name to **Standard**. Leave the field Admin principal name empty and set the fields Initial password and Password again to **opencrx** as shown in the figure below:

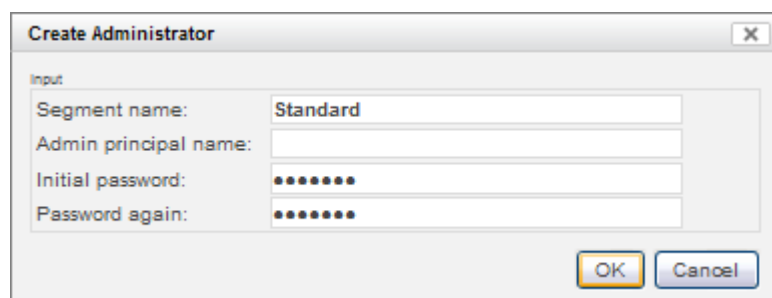
A screenshot of a 'Create Administrator' dialog box. It has a title bar with a close button. Below the title bar, there is an 'Input' section with four text fields: 'Segment name:' containing 'Standard', 'Admin principal name:' which is empty, 'Initial password:' containing seven dots, and 'Password again:' containing seven dots. At the bottom right of the dialog are 'OK' and 'Cancel' buttons.

Figure 5: Create a new segment named Standard



Please verify the spelling/capitalization of the segment name: **Standard** (with a capital "S").

Next you click **OK** to create the segment **Standard** and the segment administrator **admin-Standard**. The result of executing this operation should look as follows (Status 0 indicates that there were no errors):



Figure 6: Result of operation Create Administrator

### 4.1.3 Import Codes and Data

openCRX is distributed with many code tables and several data files (e.g. unit of measurement information, activity management process). Code tables and data files must be imported from the provided XML files to make them available to openCRX.

Execute the operation **View > Reload** and then click **OK** to start the import:

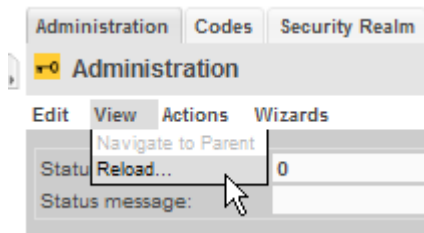


Figure 7: Load Code Tables and Data – Step 1

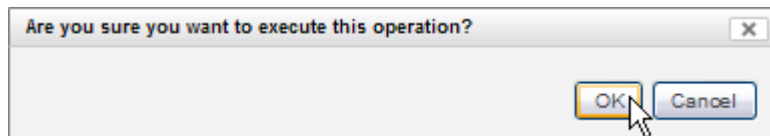


Figure 8: Load Code Tables and Data – Step 2

Please note that this operation takes some time to complete as thousands of objects are made persistent in your database during the import. The console output will look similar to the following listing:

#### Listing 10: Console Output – Importing Codes and Data

```

...
Loading codes
Loading /WEB-INF/config/code/Root/en_US/accesslevel.xml
Loading /WEB-INF/config/code/Root/en_US/accountcategory.xml
Loading /WEB-INF/config/code/Root/en_US/accountrole.xml
Loading /WEB-INF/config/code/Root/en_US/accountroleContract.xml
Loading /WEB-INF/config/code/Root/en_US/accountroleInventoryItem.xml
Loading /WEB-INF/config/code/Root/en_US/accountstate.xml
...
Loading /WEB-INF/config/code/Root/ro_RO/usageproductbaseprice.xml
Loading /WEB-INF/config/code/Root/ro_RO/utcoffset.xml
Storing 1626 code entries
Done
Loading data
Loading /WEB-INF/config/data/Root/100_uom_schedules.xml
Storing 4 objects
Loading /WEB-INF/config/data/Root/101_uoms.xml
Storing 39 objects
Done
Inspecting /WEB-INF/config/ui/1-Easy/en_US
Inspecting /WEB-INF/config/ui/1-Easy/de_CH
...
Loading 63 ui files for locale es_CO
Loading 67 ui files for locale sk_SK
Loading 68 ui files for locale ro_RO
Storing 3115 ui elements
Inspecting /WEB-INF/config/filters/
  
```

### 4.1.4 Set Access Levels of Codes

Navigate to the package Codes by clicking on the tab [**Codes**]. Then execute the operation **Security > Set Access Level** as shown below:

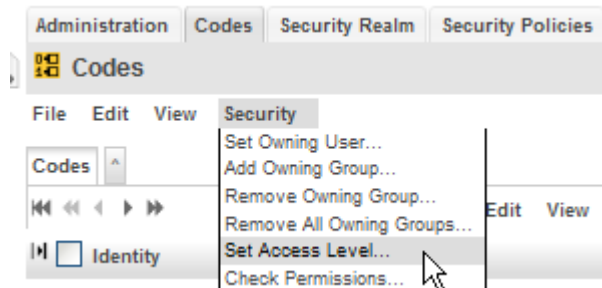


Figure 9: Execute Operation Security > Set Access Level

Set the parameters as shown below and then click **OK**:

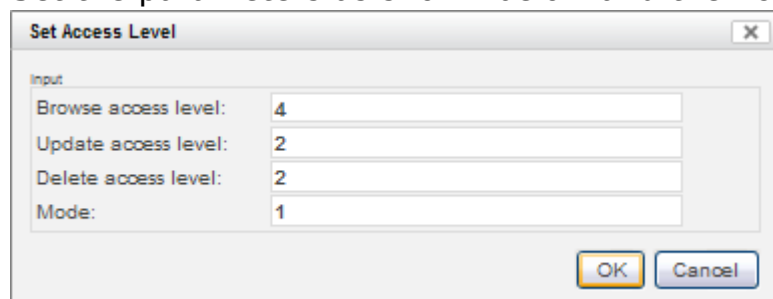


Figure 10: Parameters of Operation Set Access Level

Please note that this operation also takes a fair amount of time to complete:

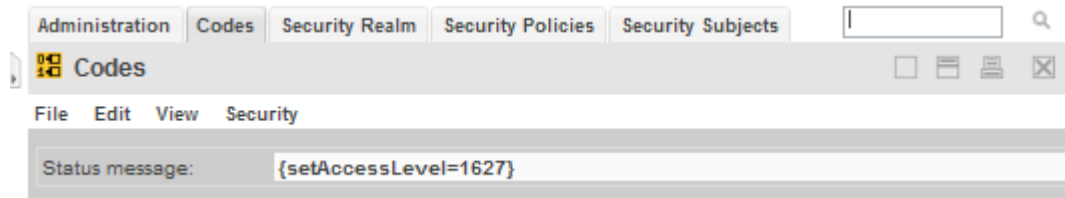


Figure 11: Result of the Operation Set Access Level

The number in the status messages indicates the number of code values processed (you might get a different number, which is not a problem).

### 4.1.5 Restarting Tomcat

Please stop Tomcat by typing <CTRL>-C into the command window.



In case you're starting Tomcat without the help of a batch file that cleans Tomcat's tmp and work directories, it is strongly advised that you clean them by hand before you start Tomcat again!

On Windows you can start Tomcat by executing `TOMCAT_HOME\bin\run.bat`, on Linux you execute `TOMCAT_HOME\bin\run.sh`.

## 4.2 Initial Setup of Segment "Standard"

Once Tomcat is up and running again, we configure the new segment and create a new user **guest**.

### 4.2.1 First Login as "admin-Standard"

Connect to the login page of openCRX. The exact URL depends on your deployment details. For a standard openCRX deployment to JBoss running on your local machine and listening at port 8080 the URL is

**http://localhost:8080/opencrx-core-CRX/Login**

You should see the openCRX Login page. Enter **admin-Standard** into the field *Username* and then enter **opencrx** into the field *Password* (you may have chosen a different password, i.e. enter the password you chose when you created the segment):

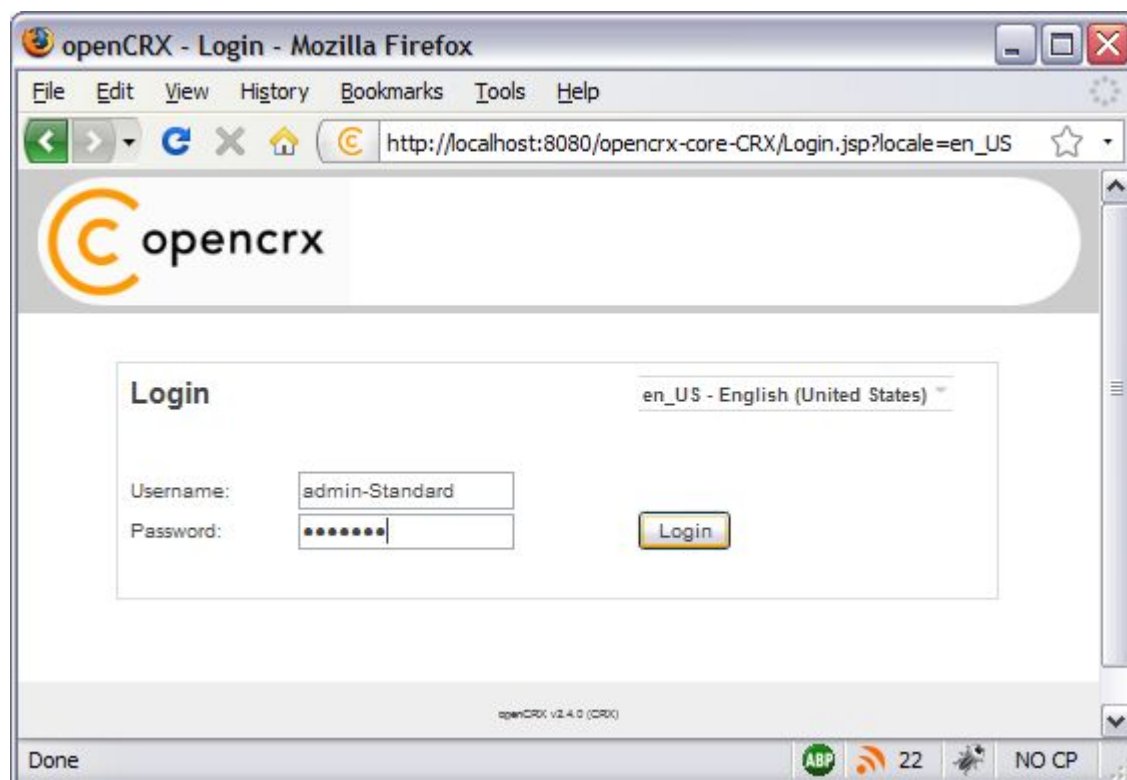


Figure 12: First login with admin-Standard / opencrx

Click the button **Login** to start the login process.

The ObjectInspectorServlet is initialized during this first login after restarting Tomcat, i.e. don't stop/kill the browser if it takes a while.



Please note that charts are not displayed when a users logs in for the first time. Activate charts by executing the operation **View > Recalculate and Refresh**.

## 4.2.2 Configure Segment "Standard"

Once the admin-Standard's homepage is displayed, start the wizard Segment Setup. Select the menu **Wizards > Segment Setup** as shown below:

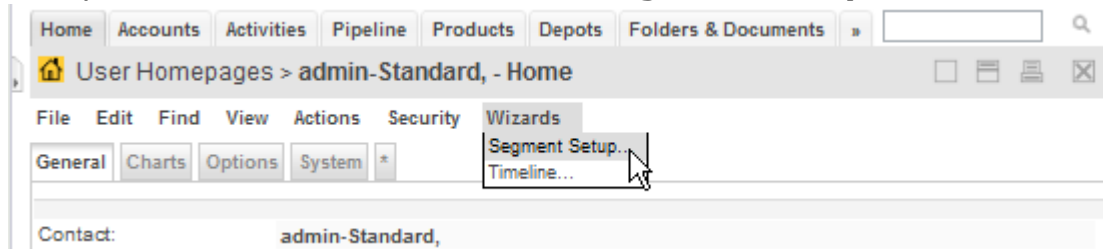


Figure 13: As admin-Standard start the wizard Segment Setup

This wizard verifies all the default objects of a segment. Running it on a newly created segments results in a lot of red crosses. Click the button **Setup** and the wizard will create all the missing objects (the red crosses will change to green check marks as shown below):

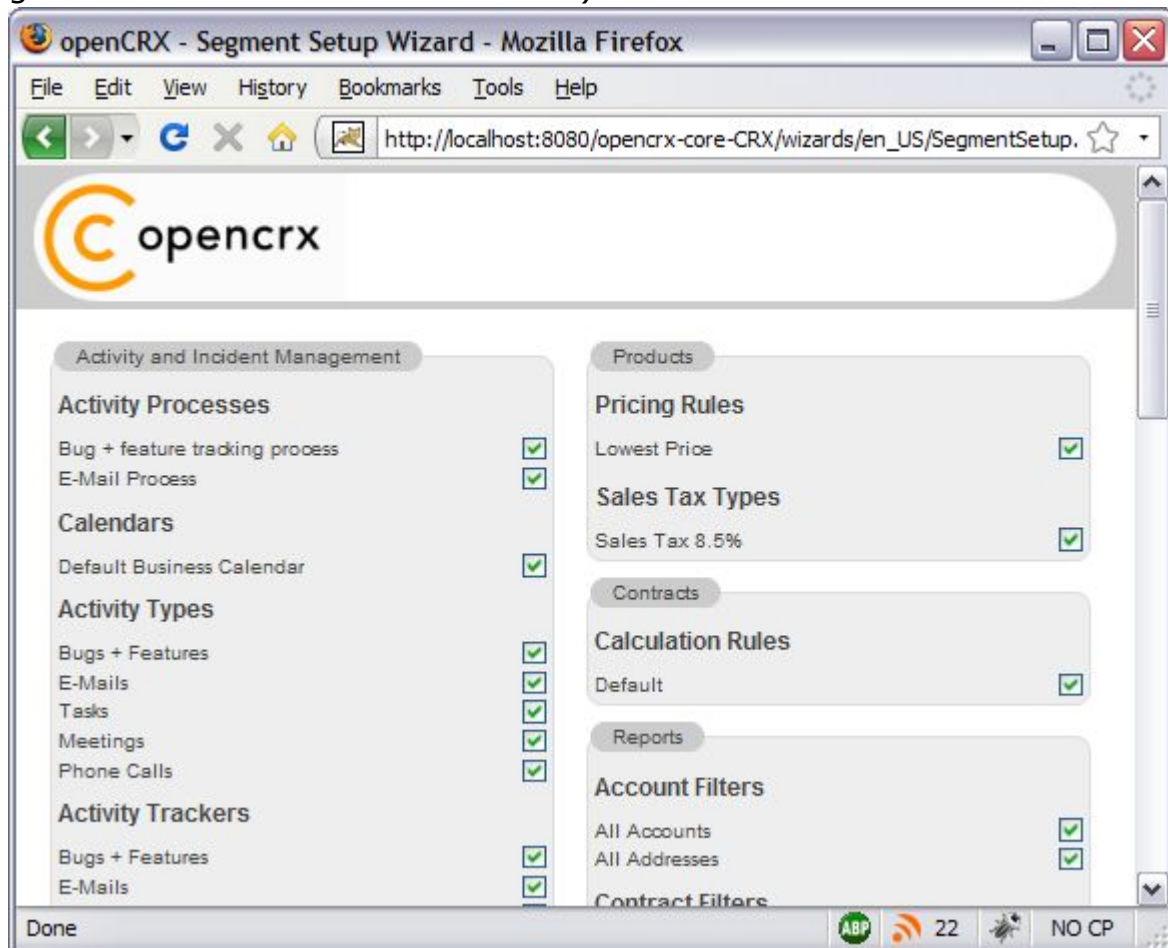


Figure 14: Wizard Segment Setup - properly configured Segment

Click on the button **Cancel** to leave the wizard and return to the homepage of the user admin-Standard.

### 4.2.3 Create a new Contact "guest"

Refresh the homepage of the user admin-Standard by clicking on the icon as show below:

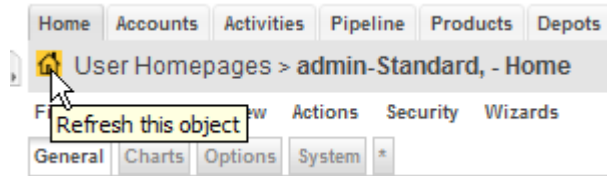


Figure 15: Refresh homepage of user admin-Standard

Hover your mouse over the fly-in tab to show the following menu:

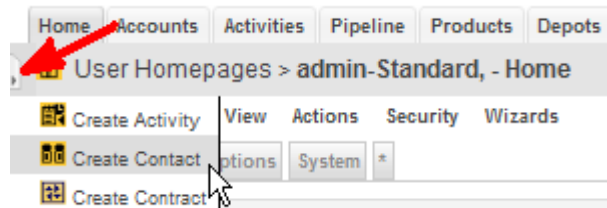


Figure 16: Fly-in menu

Choose the entry **Create Contact** and click on it.



You will see an empty form ready to be populated with the data of our new user guest. At a minimum you must provide a last name, e.g. **guest** as shown below:

openCRX - Accounts - Accounts - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://localhost:8080/opencrx-core-CRX/ObjectInspectorServlet?reque

admin-Standard@Standard en\_US Logoff admin-Standard Save Settings

Home Accounts Activities Pipeline Products Depots Folders & Documents

Accounts

File Edit View Actions Security Wizards Reports

Contact

Salutation code: N/A Department: Salutation: Do not phone: First Name: Phone business: Middle Name: Phone mobile: Last Name: guest Phone home: Alias name: E-mail business: Job Title: E-mail home: Job Role: Birth Date: Organization:

Addresses (Home / Business)

Address line(s): Address line(s): Street / PO Box: Street / PO Box: City: City: ZIP/Postal code: ZIP/Postal code: Country/Region: N/A Country/Region: N/A

Description

Description:

Search New Search Cancel

Done ABP RSS 22 NO CP

Figure 17: New contact guest – step 1

Next you click on the button [**Search**]. openCRX will verify that you don't already have a user with the same last name (but as we are working with a new/empty database this search doesn't bring up anything, of course). Now you can create the new contact by clicking on the button [**Create New**]:

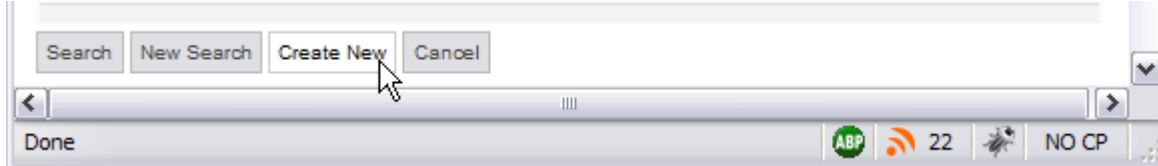


Figure 18: New contact guest – step 2

As we don't want to create activities, leads or any other objects for our new contact, we now click [Cancel] to leave the wizard:

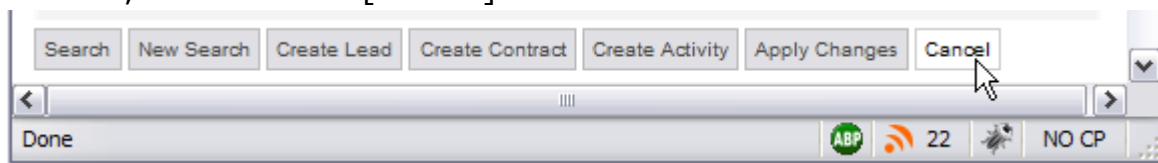


Figure 19: New contact guest – step 3

openCRX takes us back to the Accounts grid where we see the newly created contact guest:

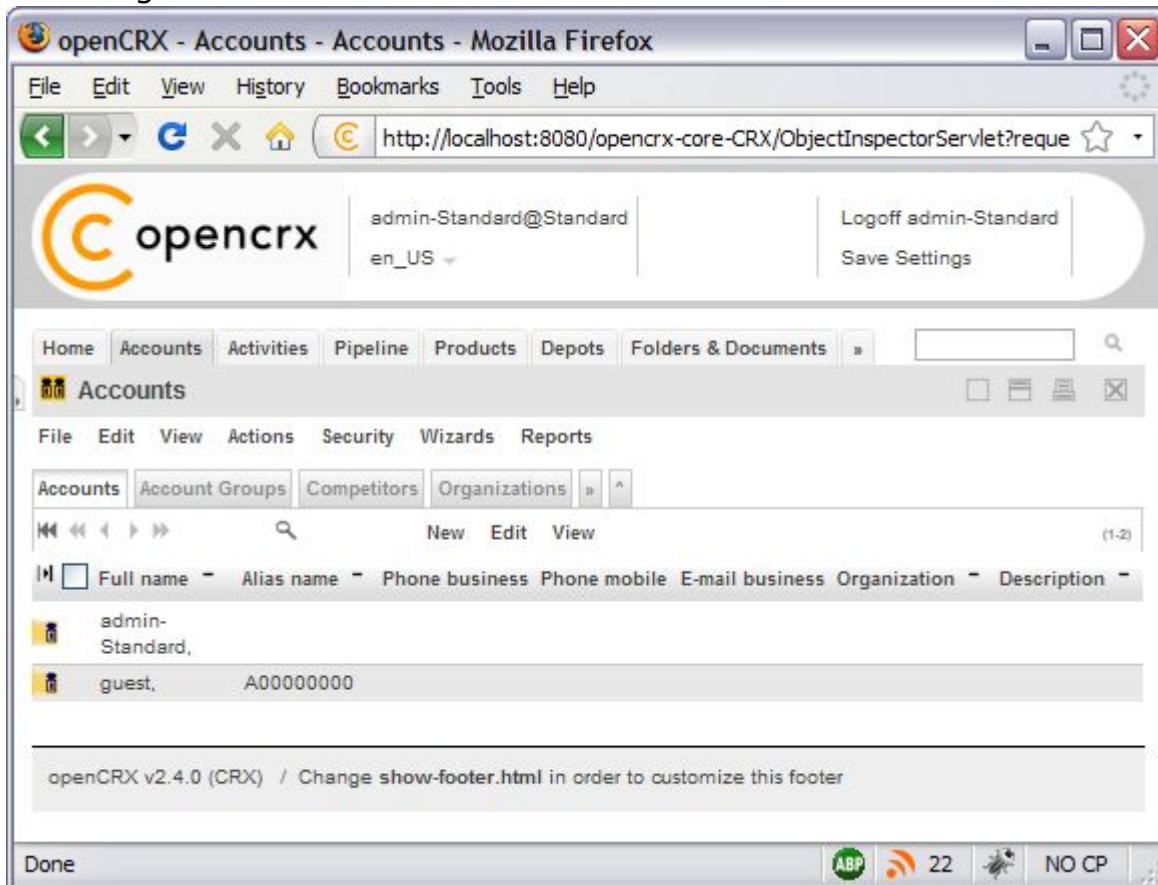


Figure 20: New contact guest – step 4

## 4.2.4 Create a new User "guest"

Click on the (hidden) tab [**User Homepages**]:

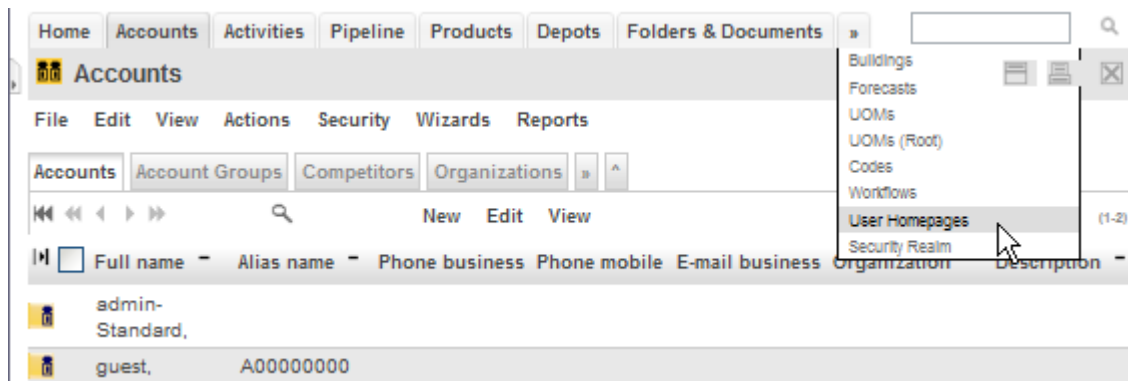


Figure 21: New user guest – step 1

Next you select the operation **Actions > Create User...** which allows you to create and initialize a new user:

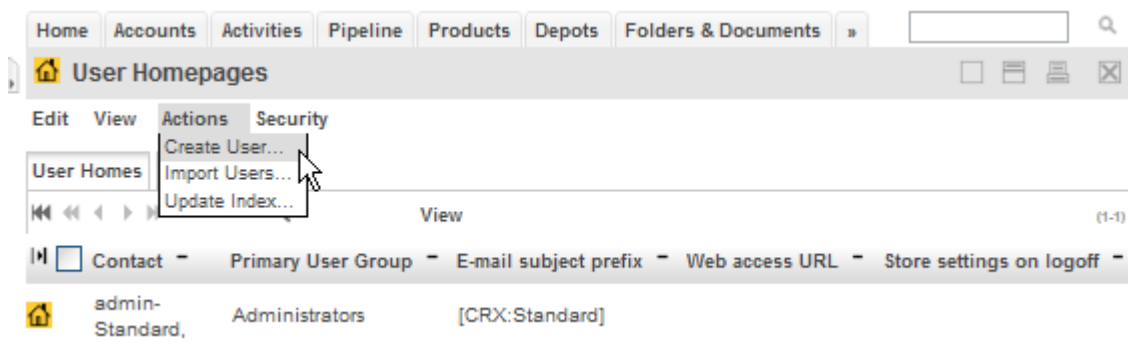


Figure 22: New user guest – step 2

Set the fields to the values as shown below – type **guest** into the field *Principal name*, use the Lookup Inspector or the auto-completer to fetch values for *Contact* and *Primary user group* (note that you must either fetch values with the Lookup Inspector or select entries from the auto-completer's drop down menu for the parameters *Contact* and *Primary user group*, i.e. it is not sufficient to just type some text into these fields), and then type a password (e.g. **opencrx**) into the fields *Initial password* and *Password again*:

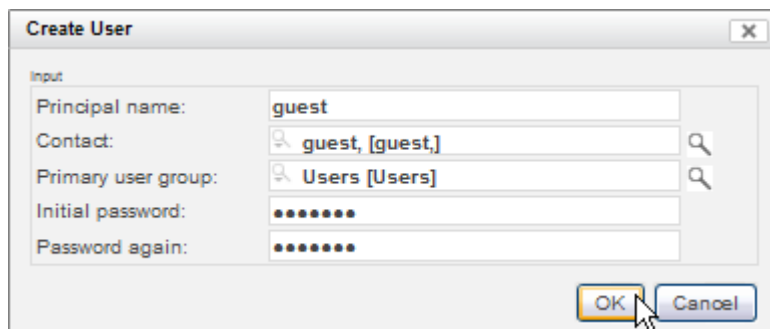


Figure 23: New user guest – step 3

Status 0 indicates that the user guest was created without errors:

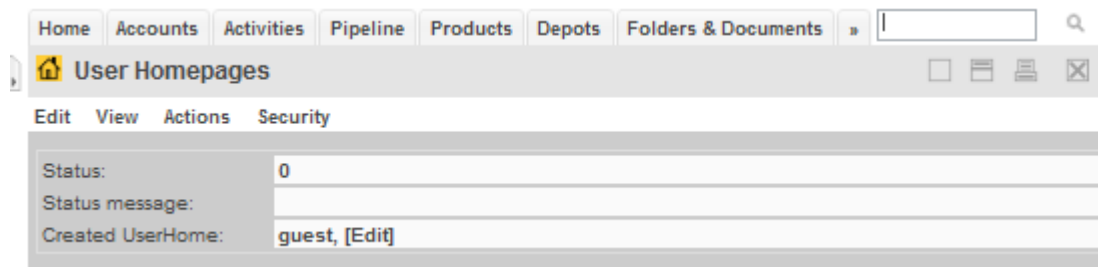


Figure 24: New user guest – step 4

Next we navigate to the homepage of the newly created user guest by clicking on the icon as show below:

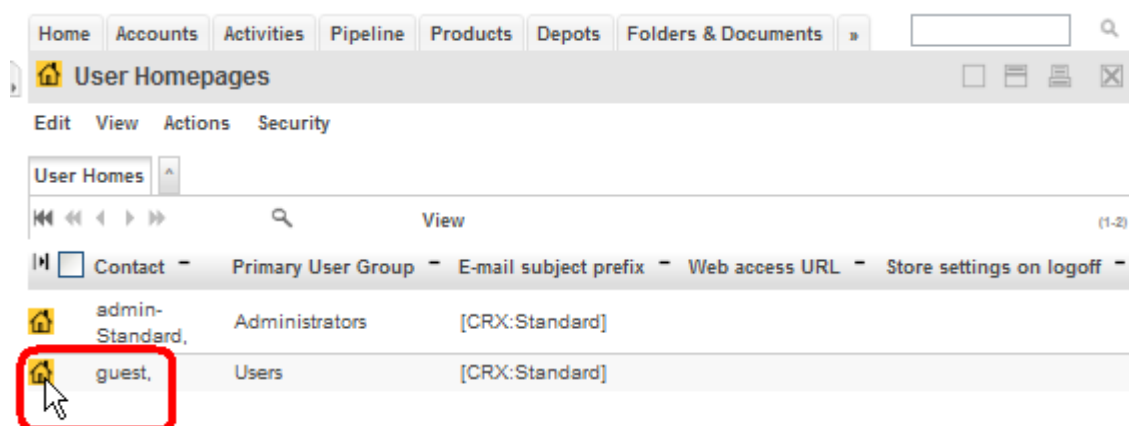


Figure 25: New user guest – step 4

Please note that we are still logged in as admin-Standard (as shown in the header of the application), but we are looking at the homepage of the user guest. Select the operation **Edit > User Settings** and click on it:

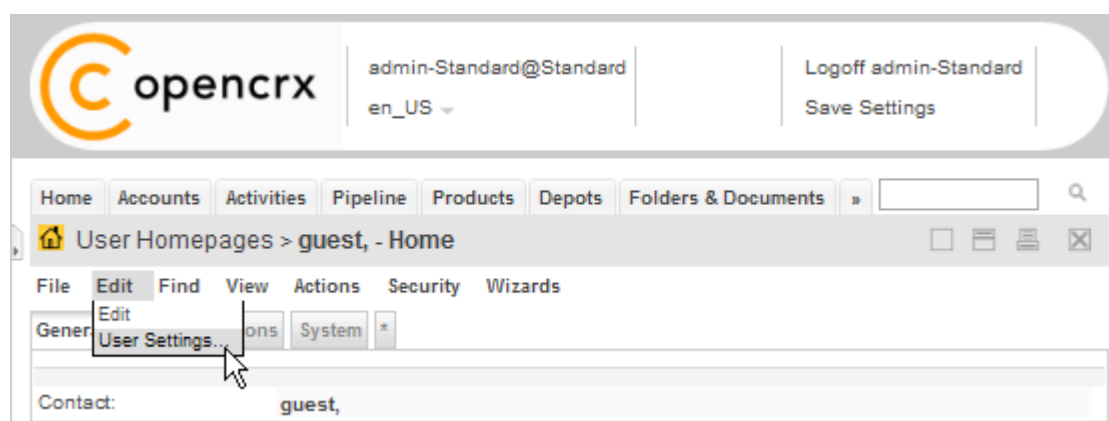


Figure 26: New user guest – step 5

This will start the wizard User Settings. You can configure various settings with this wizard. At a minimum you should probably set the **timezone** and enter the new user's **e-mail address**. Once you're done you can click the button **[Apply]**. The wizard will then create a bunch of objects and finalize the initialization of the user guest:

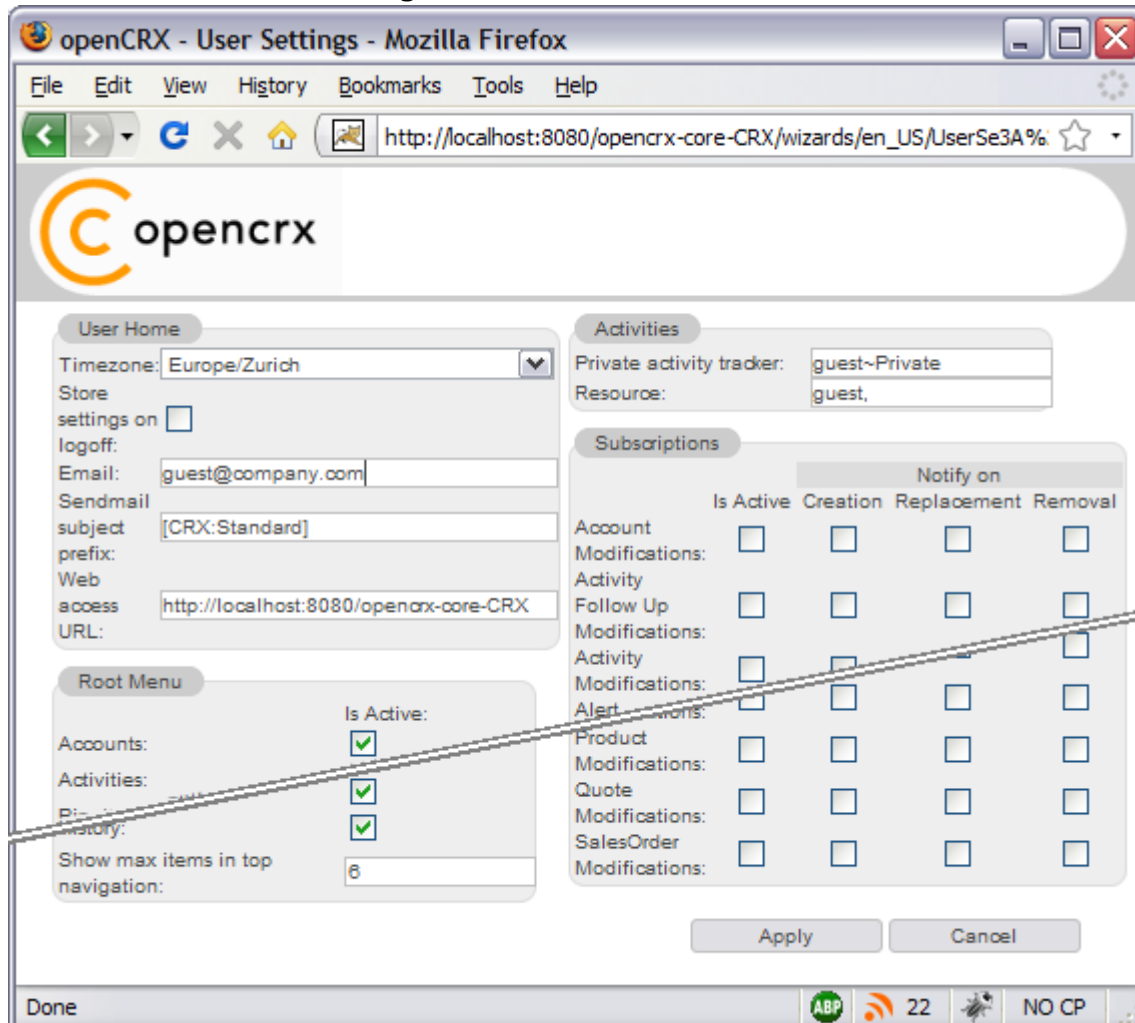


Figure 27: New user guest – step 6

Click **[Cancel]** to leave the wizard and then logoff as admin-Standard.



The wizard User Settings created a user group **<username>.Group**, in our case **guest.Group**. The primary user group of our user guest was automatically set to this new user group guest.Group.

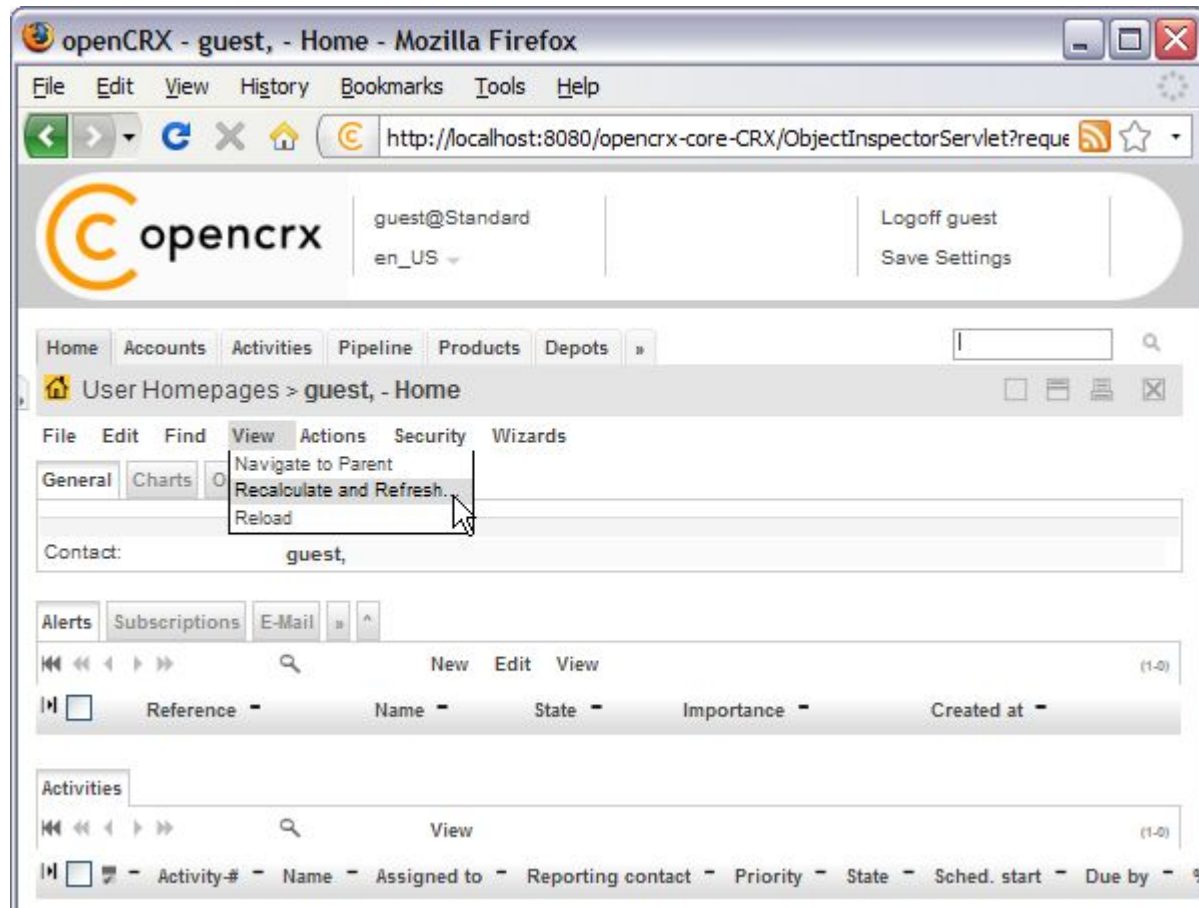
If you want to change the primary user group to anything else or if you ever must reset a user (lost password, etc.), you can re-execute the operation **Create User** as admin-Standard at any time.

If you want to reset a user without changing the user's password, you can simply leave the password fields empty when recreating the user.

## 4.2.5 Login as “guest”

Let's verify whether our new user guest can actually login. Enter **guest** into the field *username*, **opencrx** into the field *password*, and then click [**Login**].

You should now see the homepage of the newly created user guest. Initialize the charts by executing the operation **View > Recalculate and Refresh**:



That's it – you have successfully completed the installation and initialization of openCRX and you have also created a user guest.

## 5 Next Steps

Now that you have successfully installed and configured openCRX you might want to have a look at some of the additional documentation published at <http://www.opencrx.org/documents.htm>.