RefereeManager

Developer Documentation

12. November 2017

Ekkart Kleinod

✉ ekleinod@edgesoft.de

Inhalt

1 Preface

This documentation focuses on the development aspect of the RefereeManager.

1.1 Used Language

The developer's documentation is in English. As I am a German native speaker, you can ask German questions every time.

Die Entwicklerdokumentation ist auf Englisch. Da ich Deutsch als Muttersprache spreche, können Sie natürlich jederzeit auch deutsche Fragen stellen.

2 Structure and Git repositories

The RefereeManager is written in Java 8+, using FXML and Java features introduced in Java 8.

2.1 Structure

The repository is structured as follows:

folder	content
build	Build files for releases.
documentation	User and developer documentation.
files	Released files (will change in the future).
refereemanager	Eclipse project with sources, ressources and tests.
submodules	Needed submodules (edgeutils for Ant and JAXB commons).

2.2 Git-Repository/github

The git repository is maintained using github at:

– https://github.com/tt-schiri/refereemanager/

Github is used for code maintenance and issue tracking.

Short information about the structure of the git repository:

The branches are constructed regarding the git branching model of http://nvie.com/posts/a-successful-git-branching-model/

This means, there are always at least three branches:

1. master - contains released versions
2. develop - main synchronisation branch for feature, release, and hotfix branches
3. feature/work - main working branch for development

Additionally, the following branches may occur:

– feature/* - writing a special feature
– release/* - synchronizing release versions between develop and master
– hotfix/* - fast bugfixes

3 Required software

In order to develop the RefereeManager, you will need at least:

– Git
– SVN
– Ant (Java)
– Apache (incl. mysql module) and PHP
– MySQL

Please install the applications for your platform. I tested development with XU-
buntu so far.

3.1 Git

You need *git* for handling the developed code.

There are no special installation or configuration steps required in order to deve-
lop the RefereeManager.

3.2 SVN

You need *svn* for accessing the *PHPExcel* code. If you prefer to use downloaded
Archives of *PHPExcel* you won't need *svn*.

There are no special installation or configuration steps required in order to deve-
lop the RefereeManager.

3.3 Ant

You need *ant* for execution of the setup and maintenance scripts. *Ant* requires
Java. You can choose not to use *ant*, in this case you have to execute the required
scripts by hand.

There are no special installation or configuration steps required in order to deve-
lop the RefereeManager.

3.4 Apache and PHP

Install *apache* according to your platform. Make sure you install the *mysql* module.

Install *PHP* as well. Preferably combine the installation steps in order to ensure your *apache* server executes *PHP* correctly. In order to use CakePHP scripts *PHP* has to be executable by command line too.

After each change restart apache with

```
$ sudo apachectl -k restart
```

Execute the following configuration steps.

3.4.1 Access restrictions

In standard configuration, all sites in *apache2* have access restrictions. You have to allow access explicitely.

Handle this situation with care if you are on a live system. In my test system I can allow access to all sites withour having to consider security issues.

After the changes, check if http://localhost/ (still) works.

Change the following files:

– apache2.conf
– httpd.conf
– default

as follows:

apache2.conf   For some reason in newer *apache* installations the file apache2.conf does not link to httpd.conf anymore. I don't know if this is a bug, a feature, a security thing or else. In that case, add the following line to apache2.conf

```
# Include all the user configurations:
Include httpd.conf
```

Add the line before

```
Include ports.conf
```

The file is provided in the RefereeManager git repository, check if a link is needed or the changes are made faster by hand. The linking solution:

```
$ cd /etc/apache2/
$ sudo ln -s <...>/git/refereemanager/configuration/apache2/ports.conf
```

httpd.conf   Add the file `httpd.conf` to the directory `/etc/apache2` or link the given `httpd.conf` file from the RefereeManager git repository. Check if the file exists first, merge existing files if you have to.

Important changes content of `httpd.conf` in section `Directory /`:

```
Options FollowSymLinks
AllowOverride All
```

At the end of the file:

```
LoadModule rewrite_module /usr/lib/apache2/modules/mod_rewrite.so
```

The linking solution:

```
$ cd /etc/apache2/
$ sudo ln -s <...>/git/refereemanager/configuration/apache2/httpd.conf
```

In both cases it is important to allow the following of symbolic links and the override.

default   Change `default` in `/etc/apache2/sites-available/` (to be tested if this change is needed). Here too allow override and following of symbolic links in sections `Directory /` and `Directory /var/www`.

You can link to the git-stored file as well:

```
$ cd /etc/apache2/sites-available/
$ sudo ln -s <...>/git/refereemanager/configuration/apache2/sites-available/defaul
```

3.5 Known problems

3.5.1 Apache did not start

If you choose to link the needed files, there may be problems in case you encrypt your home. In that case, the links can only be resolved after you log in. You have to restart *apache* every time you log in (see Apache and PHP (Abschnitt 3.4)).

3.5.2 The mysql extension is missing

Make sure you installed the *mysql* extension correctly **and** restarted *apache* (you figured *that* out by yourselves)

If you use `localhost` you have to deactivate local file access with `LOAD DATA` statements In order to do that, edit `php.ini`, usually located in `/etc/php5/apache2/`. Uncomment the lines

```
mysql.allow_local_infile = On
mysqli.allow_local_infile = On
```

as follows

```
;mysql.allow_local_infile = On
;mysqli.allow_local_infile = On
```

3.6 MySQL

You need *MySQL* as database server.

Make sure you install the database with the default settings and without database password. If you choose otherwise, note the values in order to put them in the configuration files.

4 Setup your working environment

Now, it is time to set up your working environment. In order to do this, follow these steps:

1. check out the git sources resp. download it
2. create and link your apache target directories
3. alter the configuration files
4. copy phpMyAdmin
5. copy CakePHP
6. copy PHPExcel
7. copy PHPExcelHelper
8. copy TCPDF
9. copy RefereeManager
10. setup database
11. start development

You should use the provided *ant* scripts for some of these tasks. Simply call

```
$ cd <...>/git/refereemanager/scripts/
$ ant
```

For a short help message, stating the available targets and a short description. If you are not familiar with *ant*, refer to the Ant Manual[1].

Important: All scripts work without security warnings.

4.1 Check out the git sources resp. download it

Clone the code of the *RefereeManager*.

```
$ git clone https://github.com/ekleinod/refereemanager.git <...>/git/refereemanage
```

Clone the code of *CakePHP*, alternatively download and unzip a version.

```
$ git clone git://github.com/cakephp/cakephp.git <...>/git/cakephp/
```

Clone the code of *phpMyAdmin*, alternatively download and unzip a version. Due to the size of the sources I use a download here.

---

[1]   http://ant.apache.org/manual/

```
$ git clone git://github.com/phpmyadmin/phpmyadmin.git <...>/git/phpmyadm
```

Checkout the code of *PHPExcel*.

```
$ svn checkout https://phpexcel.svn.codeplex.com/svn/trunk/ <...>/git/php
```

Clone the code of *PHPExcelHelper*, alternatively download and unzip a version.

```
$ git clone git://github.com/ekleinod/PHPExcelHelper.git <...>/git/phpexc
```

Clone the code of *TCPDF*, alternatively download and unzip a version.

```
$ git clone http://git.code.sf.net/p/tcpdf/code <...>/git/tcpdf/
```

4.2 Create and link your apache target directories

Check if the directories exist. If yes: delete them (check if they contain valueable files first):

```
$ rm -rf <...>/apache/phpmyadmin/
$ rm -rf <...>/apache/refereemanager/
```

Now create the directories:

```
$ mkdir <...>/apache/phpmyadmin/
$ mkdir <...>/apache/refereemanager/
```

Now link the directories to your apache home directory:

```
$ cd /var/www/
/var/www/$ sudo ln -s <...>/apache/phpmyadmin/
/var/www/$ sudo ln -s <...>/apache/refereemanager/
```

All directories that are referenced by apache have to be given read rights for apache. Make sure, the whole path is readable for apache, don't forget to check the home folder.

4.3 Alter the configuration files

4.3.1 RefereeManager scripts

You have to tell the scripts where to find your directories. Default structure is:

```
../
    apache/
        phpmyadmin/
        refereemanager/
    git/
        cakephp/
        phpexcel/
        phpexcelhelper/
        phpmyadmin/
        refereemanager/
```

If your directory structure is the same, you do not have to change the configuration file.

Otherwise, edit the file

```
<...>/git/refereemanager/scripts/configuration.xml
```

Change the file according to your needs. You can check your settings with

```
$ cd <...>/git/refereemanager/scripts/
$ ant echoproperties
```

4.3.2 phpMyAdmin configuration

Change the *phpMyAdmin* configuration according to your installation. You find a configuration for the installation defaults in the RefereeManager configuration directory:

```
<...>/git/refereemanager/configuration/phpmyadmin/config.inc.php
```

If you leave your altered file at this place, it is copied by the scripts to and from the apache directory.

4.3.3 CakePHP database configuration

Note: Set your server settings, the database name (default: refereemanager), and the database table prefix (default: rfrmgr_).

Change the *CakePHP* database configuration according to your installation. You find a configuration for the installation defaults in the RefereeManager repository:

```
<...>/git/refereemanager/cakephp/app/Config/database.php
```

If you leave your altered file at this place, it is copied by the scripts to and from the apache directory.

4.3.4 TCPDF configuration

Note: Set your preferred font, sont fize, creator text etc.

Change the *TCPDF* configuration according to your preferences. You find a configuration with reasonable defaults in the RefereeManager repository:

```
<...>/git/refereemanager/cakephp/app/Vendor/TCPDF/config/tcpdf_config.php
```

If you leave your altered file at this place, it is copied by the scripts to and from the apache directory.

4.4 Copy phpMyAdmin

Copy the *phpMyAdmin* files to your phpmyadmin apache directory.

Ant target
    phpmyadmin2apache
    call: ant phpmyadmin2apache
Source directory
    <...>/git/phpmyadmin
Target directory
    <...>/apache/phpmyadmin

The script executes the following steps:

1. check if target directory exists – failure otherwise
2. delete all files and directories in the target directory

3. copy all files and directories from source directory to target directory

Important: Your *phpMyAdmin* configuration file is deleted too, restore it in the step Copy RefereeManager (Abschnitt 4.9).

4.5 Copy CakePHP

Copy the *CakePHP* files to your refereemanager apache directory.

Ant target
    `cakephp2apache`
    call: `ant cakephp2apache`
Source directory
    $<...>$`/git/cakephp`
Target directory
    $<...>$`/apache/refereemanager`

The script executes the following steps:

1. check if target directory exists – failure otherwise
2. delete all files and directories in the target directory
3. copy all files and directories from source directory to target directory
4. change access rights for $<...>$`/apache/refereemanager/app/tmp/` and it's files and subdirectories to read/write/execute for all users (so *apache* can change there)

Important: Your *RefereeManager* files and configurations are deleted too, restore them in the step Copy RefereeManager (Abschnitt 4.9). Be sure you don't have unsaved changes before executing this step.

4.6 Copy PHPExcel

Copy the *PHPExcel* files to your refereemanager apache directory.

Ant target
    `phpexcel2apache`
    call: `ant phpexcel2apache`
Source directory
    $<...>$`/git/phpexcel`
Target directory
    $<...>$`/apache/refereemanager`

The script executes the following steps:

1. check if target directory exists – failure otherwise
2. copy the helper file from source directory to target directory

Important: Your *PHPExcel* files will be overwritten.

4.7 Copy PHPExcelHelper

Copy the *PHPExcelHelper* files to your refereemanager apache directory.

Ant target
    `phpexcelhelper2apache`
    call: `ant phpexcelhelper2apache`
Source directory
    `<...>/git/phpexcelhelper`
Target directory
    `<...>/apache/refereemanager`

The script executes the following steps:

1. check if target directory exists – failure otherwise
2. copy the helper file from source directory to target directory

Important: Your *PHPExcelHelper* files will be overwritten.

4.8 Copy TCPDF

Copy the *TCPDF* files to your refereemanager apache directory.

Ant target
    `tcpdf2apache`
    call: `ant tcpdf2apache`
Source directory
    `<...>/git/tcpdf`
Target directory
    `<...>/apache/refereemanager`

The script executes the following steps:

1. check if target directory exists – failure otherwise
2. copy the helper file from source directory to target directory

Important: Your *TCPDF* files will be overwritten.

4.9 Copy RefereeManager

Copy the *RefereeManager* files in your refereemanager apache directory.

Ant target
    `refman2apache`
    call: `ant refman2apache`
Source directory
    `<...>/git/refereemanager`
Target directory
    `<...>/apache/refereemanager`

The script executes the following steps:

1. check if target directory exists – failure otherwise
2. copy all relevant files and directories from source directory to target directory

Important: Your *RefereeManager* files and configurations are overwritten.

4.10 Setup database

Set up the database. You should be able to access your database with:

`https://localhost/phpmyadmin/`

If not, check:

– the apache www server
– the link of `phpmyadmin` in the apache www root (see Create and link your apache target directories (Abschnitt 4.2))
– your phpMyAdmin configuration (see phpMyAdmin configuration (Abschnitt 4.3.2))

Create a database called `refereemanager`, use `utf8_general_ci` collation. If you choose another name, alter your CakePHP configuration file accordingly (see CakePHP database configuration (Abschnitt 4.3.3)).

Create the database tables using the code of:

`<...>/refereemanager-code/database/refereemanager.sql`

Fill the database with the default configuration data of your language (skip if you want to use a database dump that contains the default values):

```
<...>/refereemanager-code/database/<lang>/defaults.sql
```

Fill the database with your data if you have some.

Important: If you fill your database with a dump, you will most likely overwrite the loaded defaults.

4.11 Start development

Everything is ready for development.

5 Ideas

- apache.conf and config.inc.php in own folder
- link apache.conf in phpmyadmin to /etc/apache2/conf-available/phpmyadmin.conf
- link conf in avaliable to conf-enabled
- link mods-available/rewrite.load to mods-enabled
- ensure directory permissions through complete path

6 Versions

6.1 v0.2

Released: 2014–05–04

– database for referees working
– output of referee lists: HTML, PDF, Excel

6.2 v0.1

Released: 2013–10–13

– no special functionality
– saving the code before introducing the new branching model [2]

---

[2] http://nvie.com/posts/a-successful-git-branching-model/