

Comprofits Installation Guide

Software requirements:

This installation tutorial is based on a Debian Linux 8.1 installation. You can use the package name to install them via the apt-get command. If you want to deploy the application on a Windows machine or if there is no Linux package, you have to use the manual download links and consider the notes at the end of this document.

Software	Linux package	Manual Download
Java JDK 7 or higher	Openjdk-7-jdk	https://www.java.com/de/download/manual.jsp
PostgreSQL 9.4	Postgresql-9.4	http://www.postgresql.org/download/
Glassfish 4.1	-	http://download.java.net/glassfish/4.1/release/glassfish-4.1.zip
Wkhtmltopdf	wkhtmltopdf	http://wkhtmltopdf.org/downloads.html
R	r-base	https://cran.rstudio.com

Creating the Database User:

Use the command line or the pgAdmin-Tool (Windows) to connect to the PostgreSQL server and create the database user for ComProFITS.

```
create user comprofits with password 'db_pass';
```

Note: The database user's name **must** be 'comprofits'. Otherwise the schema import will fail.

Setting up the tables:

Download the latest ComProFITS schema

https://bitbucket.org/comprofits/comprofitsapp/downloads/comprofits_current_schema.sql

and import it into the database.

```
sudo -u postgres psql < schema_location
```

The schema file will automatically create a new database with the name 'comprofits' and set the ownership to the 'comprofits' user. It will also create a user named 'admin' with password 'admin' with the administrator role for the application. Please remember to change this password.

Install and set up the Glassfish server:

Download the Glassfish 4.1 server and unzip the file to a directory of your choice (using '/opt' in this tutorial). The Glassfish server has a preconfigured domain called 'domain1' which uses the ports 4848 for the admin console and 8080 for the application instance. You can create a new domain with different ports by entering the following command (optional):

```
/opt/glassfish4/bin/asadmin create-domain --adminport port1 --instanceport port2  
domain_name
```

Before you can enter the admin console, you need to set a password for the admin user (it is empty by default) and enable secure access for the domain. Otherwise, remote connections will be rejected by Glassfish.

First, start the domain.

```
/opt/glassfish4/bin/asadmin start-domain domain_name
```

Then, change the admin password.

```
/opt/glassfish4/bin/asadmin change-admin-password
```

Enable the secure access.

```
/opt/glassfish4/bin/asadmin enable-secure-admin
```

And finally restart the domain, so the changes take effect.

```
/opt/glassfish4/bin/asadmin restart-domain domain_name
```

Set up JDBC Connection and Security realm:

Open the admin console in your browser. It is located at <https://hostname:adminport>

In the menu, go to Resources -> JDBC -> JDBC Connection Pools -> New and enter/select the following values:

Property	Value
Pool Name	comprofitsPgDbPool
Resource Type	javax.sql.ConnectionPoolDataSource
Database Driver Vendor	Postgresql

On the next page, change the Datasource Classname to 'org.postgresql.ds.PGSimpleDataSource' (enter it in the field below the selection box).

Scroll down to 'Additional Properties' and delete all values but the following ones:

Property	Value
PortNumber	5432
DatabaseName	comprofits
User	comprofits
Password	<i>db_pass</i>
ServerName	Localhost

Click 'Finish' to create the Connection Pool.

Go to Resources -> JDBC -> JDBC Resources -> New and enter/select the following values:

Property	Value
JNDI Name	jdbc/comprofitspgdb
Pool Name	comprofitsPgDbPool

Click 'Ok' to create the JDBC Resource.

Go to Configurations -> server-config -> Security -> Realms -> New and enter/select the following values:

Property	Value
Name	comprofits_jdbc_realm
Class Name	com.sun.enterprise.security.auth.realm.jdbc.JDBCRealm
JAAS Context	jdbcRealm
JNDI	jdbc/comprofitspgdb
User Table	all_users
User Name Column	uname
Password Column	pwd
Group Table	all_users
Group Table User Name Column	uname
Group Name Column	rol
Password Encryption Algorithm	SHA-256

Click 'Ok' to create the Security Realm. You can now exit the admin console.

Create Directory for Images:

The application stores the profile images in the folder /var/webapp/images. You can create it with

```
mkdir -p /var/webapp/images
```

Install additional R packages:

The application uses several packages of R that are not part of the default installation. To install them, you have to start the R executable via command line (just enter 'R'). After that, use the following commands to download and activate the packages:

```
install.packages("ggplot2")
install.packages("ScottKnott")
library(ggplot2)
library(ScottKnott)
```

Now exit R with the following command:

```
q()
```

Note: If you get a message, that the package is not available for the R version you're using, just exit and start R again and try downloading from a different mirror (Mirror 40 worked for both packages).

Create script for wkhtmltopdf:

For wkhtmltopdf to work properly under Linux, you have to install the Xvfb package and create a wrapper script for the wkhtmltopdf executable.

First, install the Xvfb package.

```
apt-get install xvfb
```

Then, rename the original wkhtmltopdf executable.

```
mv /usr/bin/wkhtmltopdf /usr/bin/wkhtmltopdf_2
```

Now, create a new file with the name 'wkhtmltopdf' in the /usr/bin folder with the following content:

```
xvfb-run -a -s "-screen 0 640x480x16" wkhtmltopdf_2 --quiet $*
```

Finally, make the new script executable for all users.

```
chmod 755 /usr/bin/wkhtmltopdf
```

Access the source code and license:

The source code of the project is provided as Open Source Software under the Apache License 2. For more details about the license please consult: <http://www.apache.org/licenses/LICENSE-2.0>

You can access the source code in <https://bitbucket.org/comprofits/comprofitsapp>

The project is a standard maven-based java enterprise edition application. Most of the libraries will be downloaded from the development environment, except RCaller which is available in the files

directory "comprofitsapp /files/runner_jar/". We used the NetBeans IDE in the development of this application,

Notes for Windows Users:

The application is configured for deployment on Linux by default. If you use Windows on the server, you need to change the paths of the file upload folder, wkhtmltopdf and R executables **after** the deployment of the application.

Open the web.xml file in the folder

Glassfish-folder\glassfish\domains\domain_name\applications\comprofitsapp\WEB-INF

with an editor and change the param-values for R_EXE and WKHTMLTOPDF_EXE and FILE_UPLOAD_DIR. After that, you can restart the instance and it should work.