

## 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	<a href="https://www.java.com/de/download/manual.jsp">https://www.java.com/de/download/manual.jsp</a>
PostgreSQL 9.4	Postgresql-9.4	<a href="http://www.postgresql.org/download/">http://www.postgresql.org/download/</a>
Glassfish 4.1	-	<a href="http://download.java.net/glassfish/4.1/release/glassfish-4.1.zip">http://download.java.net/glassfish/4.1/release/glassfish-4.1.zip</a>
Wkhtmltopdf	wkhtmltopdf	<a href="http://wkhtmltopdf.org/downloads.html">http://wkhtmltopdf.org/downloads.html</a>
R	r-base	<a href="https://cran.rstudio.com">https://cran.rstudio.com</a>

### 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://github.com/george-kakarontzas/HReady/blob/master/comprofits.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.

### Insert admin user for application:

Connect to the 'comprofits' database and use the following command to create an admin user with standard credentials (admin/admin) for the application:

```
insert into employee (idemployee, identity_card_number, social_security_number,  
first_name, last_name, gender, province, address, postal_code, city, country, date_of_birth,  
phone_private, phone_mobile, email, photo_path, username, password, marital_status,
```

```
number_of_children, department_iddepartment, current_in_company_employment_id,  
role, is_active, division_iddivision, cv_path) VALUES (1, 123456789, 123456789, 'John', 'Doe',  
1, 'Rhineland-Palatinate', 'Hoelderlinstr. 3', 57076, 'Siegen', 'DE', '1900-01-01', 123456789,  
123456789, 'webmaster@comprofits.eu', NULL, 'admin',  
'8c6976e5b5410415bde908bd4dee15dfb167a9c873fc4bb8a81f6f2ab448a918', 'M', NULL,  
NULL, NULL, 'administrator', 't', NULL, NULL);
```

### **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

Change the Datasource Classname to 'org.postgresql.ds.PGSimpleDataSource' (enter it in the field below the selection box).

For the tab "Additional Properties" scroll down to 'Additional Properties' and delete all values but the following ones:

Property	Value
PortNumber	5432
DatabaseName	comprofits
User	comprofits
Password	db_pass
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 and other files such as the users' CVs, in the folder /var/webapp/images. You can create it with

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

If you are installing the application on a windows server then create the folder in c:\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
```

### **Download and deploy the application:**

Download the application from GitHub into a folder of your choice and deploy it to Glassfish via

```
/opt/glassfish4/bin/asadmin --port adminport deploy war_file
```

**Note:** The admin port specifies under which domain the app is deployed (4848 for domain1).

The application is now available at <http://hostname:instanceport/comprofitsapp>

**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.