

Install WAR on Tomcat

The eHour WAR distribution is intended for deployment into an existing application server.

The way of deployment changed with version 1.4.2. These instructions only apply to version 1.4.2, for older version please use [these steps](#).

1. Install Java

The Java runtime environment is not bundled with eHour. If Java is not installed, download and install the latest JRE from [Oracle's site](#). Java 7 or higher is required.

2. Download eHour

If you haven't done so already, download the WAR distribution of eHour from <http://ehour.nl/download/download.phtml>

Extract it and remember the path, the location where you unpacked the zip file is the eHour home directory


2. Apache Tomcat installation

Apache Tomcat is the container server which hosts the application. If you don't have Apache Tomcat installed, download it from <http://tomcat.apache.org/download-80.cgi> and follow [their instructions](#). Apache Tomcat 7 is the minimum required version.

3. Configure Apache Tomcat

Set the location of your [eHour Home directory](#) by creating an environment variable EHOURL_HOME with the absolute path of the eHour directory.

Windows


 Locate the bin directory of your Tomcat install. It contains the catalina.bat and other scripts which startup Tomcat. In this directory create a file **setenv.bat** or edit it when it already exists.

Assuming that your [eHour Home directory](#) is at c:\ehour, add to the contents of this file the line

```
set EHOURL_HOME="c:\ehour"
```

Of course change the path to where your EHOURL_HOME dir is actually located

Linux

 Locate the bin directory in your Tomcat's install directory. Among other files it contains catalina.sh and startup.sh. In this directory create or edit the file **setenv.sh**. Make sure that it's executable with


```
chmod +x setenv.sh
```

Add the following line to setenv.sh. Change /opt/ehour to point to the [eHour Home directory](#):

```
export EHOURL_HOME="/opt/ehour"
```

4. Configure database

4a. Create initial datamodel

 Skip this step when you're performing an upgrade.

MySQL

Follow the MySQL installation guide ([Linux](#), [Setup MySQL on Windows](#)) to install and setup your database.

PostgreSQL

Follow the [PostgreSQL install guide](#) to install and setup your database.

4b. Configure eHour to connect to the database

Configure to which database eHour connects by changing the ehour.properties. The instructions are different per database, follow [these steps](#).

4c. Add database drivers to Tomcat

MySQL

Download the Connector driver from <https://dev.mysql.com/downloads/connector/j/>. Choose the platform independent option to download the tarred or zipped driver. Copy the `mysql-connector-java-8.xx.jar` to Tomcat's lib directory.

PostgreSQL

Download the JDBC driver from <https://jdbc.postgresql.org/download/postgresql-42.2.5.jar>. Copy the downloaded jar file to Tomcat's lib directory.

5. Install eHour in Tomcat

Copy the WAR file from eHour's app directory to Tomcat's webapp directory. That's it.

The filename of WAR file dictates the URL. When the WAR file that you copy to Tomcat's webapp directory is named "ehour-1.4.2.war", the URL where eHour is available will be <http://localhost:8080/ehour-1.4.2/>. When you want the URL to be <http://localhost:8080/ehour>, make sure to rename the WAR to `ehour.war`.

6. Start Apache Tomcat

Make sure you have started your database first.

Start Tomcat by running `startup.sh` in the `bin` directory of the Tomcat installation directory. After a couple of seconds Tomcat and eHour will be initialized and you can access the application at <http://localhost:8080/ehour>