

Annex 21: Technical documentation_IT

HazChemAtWork

Logged in as post@claudia-berg.de | Log out | [HazChem Administration](#)

SEARCH

Substance Name or CAS No. / EC No.

Exact name
 Only aggregated data

OR

Country

OR

Year

Methodology for use of data in HazChem@Work database

EXAMPLES OF ADDITIONAL FUNCTIONS

OR

FILTER

OR

ESTIMATE THE NUMBER OF EXPOSED WORKERS

OR

PROVIDE FEEDBACK

Overview of data provided for HazChemAtWork:

[Substances](#) [Data Providers](#) [Measurement Methods](#)

Funded by the European Union

HazChem@Work Test Database

Technical Documentation
November 2016

Table of contents

1. Purpose and use of this document	3
2. Technical requirements	3
2.1. Web server	3
2.2. MySQL version	3
3. System overview	4
3.1. Database	4
3.2. PHP files and structure	5
3.2.1. Front-end	5
3.2.2. Back-end	6
4. Installation.....	7
5. Import data	8

1. Purpose and use of this document

This document provides guidance and is intended to assist in the basic understanding of the test database's technical design and functionality.

Both technical and system overview are given in order to set up the test database on another web server.

2. Technical requirements

The test database can only be installed either on a web server or on a personal computer running a LAMP/WAMP (open-source components Linux (Windows), Apache, MySQL, and PHP).

2.1. Web server

The test database needs a web server (Apache recommended as most robust server) with PHP 5.4.x or higher (PHP 7 is currently not supported), MySQL and session support enabled.

2.2. MySQL version

The used MySQL version should be 5.1.x or higher.

3. System overview

General information

The HazChem@Work test database is a web based application written in HTML/CSS and using the script language PHP for connecting to a MySQL database and displaying content.

PHP is a script language for creating dynamic websites and web-based applications. It is used by over 81% of websites as a server side programming language. PHP allows you to connect to and manipulate databases.

MySQL is an open-source relational database management system (RDBMS) - its source code available under the terms of the GNU General Public License.

3.1. Database

The database consists of different tables linked to each other.

Here is an overview of all tables and their purposes:

Tables containing user information/logging

- benutzerdaten -> user data incl. log-in data and encrypted pwd
- logdaten -> log files (log-in, log-out, registration)

Tables containing provided data

- general_data -> general information, IDs, data type
- country_data -> country ID, name and No. of employees (future use)
- data_source_data -> database name, description and owner
- substance_data -> substance data
- exposure_data -> exposure and contextual data
- disease_data -> disease data and adverse effects
- sector_data -> sector and occupation data
- production_data -> data on production and use of chemicals (future use)

Tables containing additional information

- oels -> occupational exposure limits
- oel_footnotes
- subst_500_20 -> list of 500 substances (coarse and fine scoring)

3.2. PHP files and structure

The test database consists of a front-end (accessible for registered front-end users) and a protected back-end (only accessible for users, that are marked as administrators).

3.2.1. Front-end

The front-end provides search and filter functionalities as well as lists, guidances, dummy functionality and the possibility to give feedback.

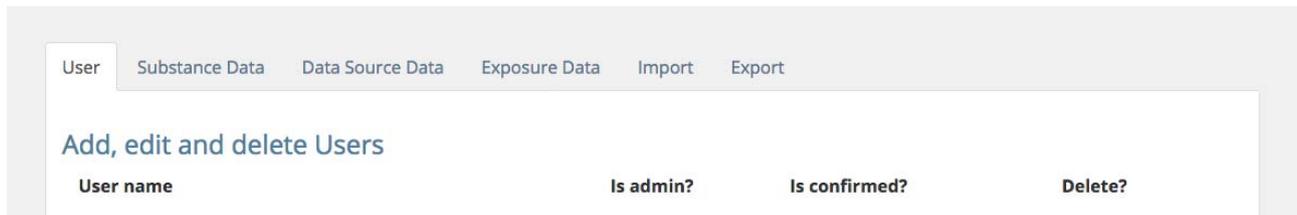
To provide a user-friendly and responsive interface, the framewok 'bootstrap' is used.

The file structure is as follows:

- alldetails.php -> view: 'all details'
- alldetails_disease.php ->view: 'all details' for disease data
- bootstrap -> bootstrap files
- compare_form.php -> form for comparison of expos. data
- compare.php -> comparison of exposure data
- css -> bootstrap files and style sheets
- detail.php -> detail view
- detaildisease.php -> detail view for disease data
- fonts -> bootstrap web fonts
- getsubstancehint.php -> auto-complete helper
- hcawadmin -> back-end
- images -> images used in front- & back-end
- inc -> included files and methods/functions
- index.php
- index2.php
- js -> javascript files
- methoden.php
- methodology_providing_data.php
- methodology_use.php
- overview_data_provided.php -> overview of provided data
- populate.php -> auto-populate helper (filter function)
- subst_500.php -> list of substances relevant f. work places
- uploads -> uploaded files (back-end)

3.2.2. Back-end

The back-end itself provides functionality for administrators and allows them to manipulate



data in these sections:

- User: add, edit and delete users
- Substance data: edit substance data
- Data source data: edit data source data
- Exposure data: edit exposure data
- Import: import data provided as described in chapter 5 ('Import data')
- Export: export all data or filtered parts of it

All files needed for the back-end are stored in a single folder ('hcawadmin').

Its structure is as follows:

- addons
- images
- inc
- index.php
- jsscripts
- popups
- scripts
- styles

Most folders (except 'inc') contain files and scripts for the used WYSIWYG editor.

The 'inc' folder contains files, that are included in 'index.php':

- benutzer.inc.php -> edit user data
- datenbank.inc.php -> connection to database
- dbdaten.inc.php -> edit data source data
- exposuredaten.inc.php -> edit exposure data
- funktionen.inc.php -> methods for
- substanzdaten.inc.php -> edit substance data
- import.inc.php -> import data
- export.inc.php -> export data

4. Installation

The database's source files are provided in a zip file.

To install the database on a web server or a computer running a LAMP/WAMP (s. a.), follow these steps:

- **Extract the zip file** on your computer: It contains two folders; one for the PHP files and one for the SQL files.
- **Upload the PHP files** to a directory on your web server using an FTP client.
- **Log in to MySQL** and create a database for the HazChem@Work test database.
- **Create a user** for this database.
- **Open the file 'inc/datenbank.inc.php'** and **insert the required information** (database server, database user, database password and database):

```
<?php
$server="localhost"; // db server
$user="XXX"; // db user
$password="XXX"; // db password
$datenbank="hazchematwork"; // db
MYSQL_CONNECT($server, $user, $password) or die ("<H3>Database unreachable</H3>");
MYSQL_SELECT_DB($datenbank) or die ("<H3>Database does not exist</H3>");
?>
```

- **Create an administrator account:**
INSERT INTO benutzerverwaltung VALUES (

```
NULL,
[USER EMAIL],
[PASSWORD encrypted],
[FIRST NAME],
[LAST NAME],
[COMPANY],
'1',
'1',
[current date in format 0000-00-00],
[current date in format 0000-00-00]
)";
```

- **Log in into the database.**
- Further users can use the registration form. Registrants have to be marked as "confirmed" in the protected back-end (only accessible for administrators).

5. Import data

Data that should be imported automatically (back-end import functionality) needs to be in CSV format.

The fields/field names below show, what kind of data can be automatically imported at the moment. The functionality has not been extended due to missing data.

CSV specifications

The encoding has to be UTF-8, fields must be separated by semicolon and enclosed within double-quote characters. The first line must contain the field names as specified below.

Field names

The field names are based on the data collection format (annex 10), but with underscores instead of blanks and in some cases abbreviated:

substance_name
cas_number
ec_number
index_number
is_inhactive
is_dermal
method_name
method_is_standardised
method_is_specified measurand
nomeasurements
nocompanies
personal_static
measurement_conditions
peak_value
peak_value_unit
min_value
min_value_unit
50percentil
50percentil_unit
95percentil
95percentil_unit
additional_work_process
additional_work_task
workplace_name_description
additional_technology
od_sector
od_occupation
nace_code
sector_isco
data_source_id
data_year
country