# Administrator's Guide 1.11.0

# Table of Contents

# Open Hospital

# 1 Abstract

OH - Open Hospital (https://www.open-hospital.org/) is a free and open-source Electronic Health Record (EHR) software application. Open Hospital is deployed as a desktop application that can be used in a standalone, single-user mode (PORTABLE mode) or in a client/server network configuration (CLIENT mode), where multiple clients and users connect to the same database server.

Open Hospital is developed in Java and it is based on open-source tools and libraries; it runs on any computer, requires low resources, and is designed to work without an internet connection.

Open Hospital is the first of a set of software applications that ISF [1: Informatici Senza Frontiere - https://www.informaticisenzafrontiere.org] has developed to support the information management and the activities of hospitals and health centers in the simplest manner possible, by providing tools for the hospital administrative operations (like registering patients, manage laboratory analysis and pharmaceutical stocks) and to produce detailed statistics and reports. It was first deployed in 2006 at the St. Luke Hospital in Angal (Uganda) and it is now used in dozens of different locations around the world.

# 2 Introduction

Open Hospital features include:

- Pharmacy management
- Laboratory management
- OPD management
- Patient admission and discharge management
- Pregnancy management
- Malnutrition control management
- Vaccines database
- Patient billing support
- Therapy management
- Appointment scheduling
- Internal communication
- Statistics and printing

In the following chapters, all the information needed to install, configure, run and maintain an Open Hospital installation are presented. The information included also allows enabling and disabling functionality as well as managing users and groups. All the information needed to use the application software is not included; please refer to the *User's Guide*, supplied with the application software.

# 3 Installation

## 3.1 Open Hospital

Open Hospital is a client application that can be used on a single computer (PORTABLE mode) or in a client/server configuration (CLIENT mode). The server part utilizes a DBMS [2: DataBase Management System] to archive, share, and present the information. The free and open-source MariaDB [3: See https://www.mariadb.org] is the database server of choice.

The typical client/server, networked (LAN) configuration is shown in the following figure:



created with Gliffy
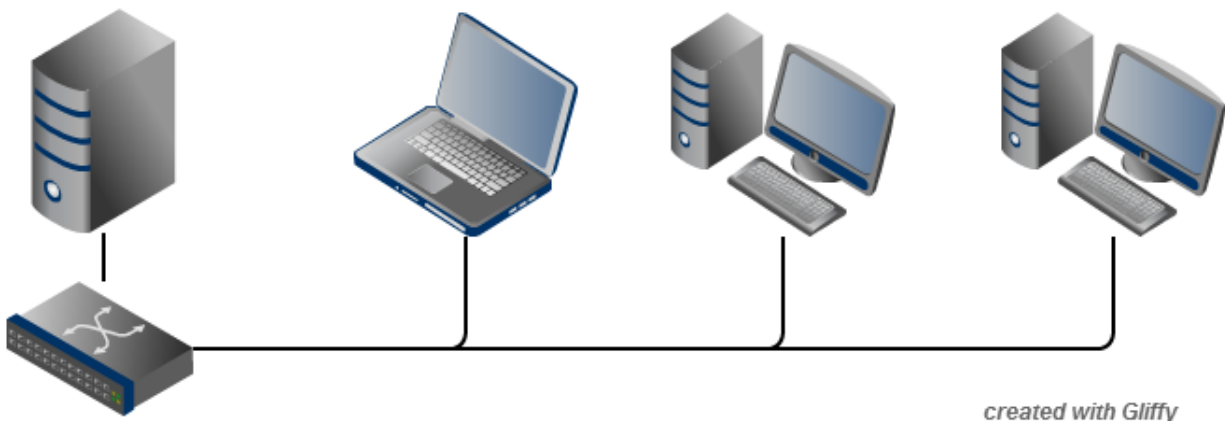
The Administrator/Installer tasks are hereby listed:

- Install the JRE - Java Runtime Environment (see compatibility matrix for the right version) on every client
- Install the MariaDB/MySQL server on a computer that will act as the OH server
- Create the OH database through the execution of a script in the folder sql/ within the .zip file folders
- Copy the Open Hospital application on every client (it may be installed on the server also)
- Configure each Open Hospital instance to point to the OH server (using the hostname or the IP address)

For the Java JRE installation refer to the next chapter Java Runtime Environment and the online documentation.

For the MariaDB/MySQL database server installation and configuration refer to the chapter Database Server and the online documentation.

For the OH database creation refer to the chapter Open Hospital - Database creation and the online documentation.

For the Open Hospital installation on the clients, it is sufficient to copy and extract the zip package

in the desired folder.

For the Open Hospital configuration refer to the chapter Configuration Files.

### 3.1.1 Prerequisites - Software compatibility matrix

Before starting, check the following software compatibility matrix, to verify the external software versions needed to run Open Hospital and the current available extended features for every version, architecture, and operating system.

| Open Hospital compatibility matrix | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **OH version** | **MariaDB / MySQL** (community version) | Java JRE version (*) | O.S. | O.S. arch | JAVA/JRE arch | Special Features | | |
| | | | | | | DICOM (*) | SMS (gsm) | Webcam |
| **1.11.1** | **10.2.40**/5.7.35 | **1.8 - zulu8.56.0.21** | Ubuntu 21.04 | x64 | 64bit | **v** | **x** | **v** |
| | | | Windows 7/10/11 (**) | x64 | 64bit | **x** | **v** | **v** |
| | | | Windows 7/10/11 (**) | x64 | 32bit | **v** | **v** | **v** |
| | | | Windows 7/10 (**) | i386 | 32bit | **v** | **v** | **v** |
| **1.11.0** | **10.2.40**/5.7.35 | **1.8 - zulu8.56.0.21** | Ubuntu 21.04 | x64 | 64bit | **v** | **x** | **v** |
| | | | Windows 7/10/11 (**) | x64 | 64bit | **x** | **v** | **v** |
| | | | Windows 7/10/11 (**) | x64 | 32bit | **v** | **v** | **v** |
| | | | Windows 7/10 (**) | i386 | 32bit | **v** | **v** | **v** |
| | | | Ubuntu 18.04 | i386 | 32bit | **v** | ? | ? |

| Open Hospital compatibility matrix | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **1.10.0** | 5.1.x | 1.6 | Ubuntu 20.04 | x64 | 64bit | v | ? | v |
| | | | Windows 7/10 | x64 | 64bit | x | v | v |
| | | | Windows 7/10 | x64 | 32bit | v | v | v |
| | | | Windows 7/10 | i386 | 32bit | v | v | v |
| | | | Ubuntu 18.04 | i386 | 32bit | v | x | ? |
| **1.9.1** | 5.0.x | 1.6 | Ubuntu 20.04 | x64 | 64bit | v | ? | v |
| | | | Windows 7/10 | x64 | 64bit | x | v | v |
| | | | Windows 7/10 | x64 | 32bit | v | v | v |
| | | | Windows 7/10 | i386 | 32bit | v | v | v |
| | | | Ubuntu 18.04 | i386 | 32bit | v | ? | ? |

| | |
|---|---|
| (*) | With Java 9+ loading a JPG/JPEG as DICOM image may not work |
| (**) | Windows Powershell 5.1 is required |
| **v** | *working* |
| **x** | *not working* |
| **?** | *not tested* |
| n/a | *not applicable* |
| *last updated* | *2021.09.16* |

### 3.1.2 Java Runtime Environment (JRE)

Depending on the operating system, there are different ways to install the Java Runtime Environment (JRE). The oh.sh / oh.bat scripts can be used to download and install the latest JRE available; it is also possible to install it manually, following the specific instructions for the operating system being used.

See Software Compatibility Matrix to identify the correct Java version.

### 3.1.3 Database Server

Depending on the operating system, there are several software tools to install and manage a MariaDB/MySQL database server. In most cases just a standard installation package of the software is needed, paying attention to configuration options during the installation process.

> **i**  See Software Compatibility Matrix to identify the correct MariaDB / MySQL version.

> **i**  MariaDB is the preferred alternative for the DBMS.

#### 3.1.3.1 MariaDB/MySQL - Server installation

Please referer to the latest MariaDB/MySQL Server online documentation for download [4: https://mariadb.org/download/] and installation [5: https://mariadb.org/documentation/] information while using this document.

**on a Linux machine**: during the installation, a "root" password must be defined; it is very important to choose it carefully and to keep it safe. If the password is not set during installation an empty password will be automatically set for the root user.

**on a Windows machine**: launch the installation and pay attention to the following steps:

- Custom Setup
- Install all components except Development Components
- Select Modify password for database user 'root', choose the root password, and keep it safe
- Select Enable access from remote machines for 'root' user
- Select UTF8 as the default charset for Best Support For Multilingualism
- Select Install as service
- Select Enable networking and write down the Port Number in an installation document

> **i**  MariaDB installer does not include the bin directory in the Windows PATH environment variable; it must be added manually to the Windows system variable PATH.

To check if the server is active and running, open a terminal window (command prompt) and type the following command:

**# mysql –u root -p**

**Enter password:**

The terminal should reply with the MySQL client command line prompt:

**mysql>**

The prompt means that a successful connection to the database server instance that was just

installed has been established.

> If the terminal does not reply as above most probably during installation the **Include Bin Directory in Windows PATH** option was not selected and therefore Windows can not find the mysql command. If this is the case add the path manually by searching "system variables" in the Control Panel or append the full path to the "mysql" command (e.g., "C:\Program Files\MySQL\MySQL Server 5.0\bin\mysql" [with the quotes]). Another option is to uninstall and reinstall MySQL with the **Include** option selected.

### 3.1.3.2 MariaDB/MySQL - Configuration

Please use the following configuration values for MariaDB/MySQL server in my.cnf config file. Values can be adjusted and adapted to the hardware resources available. Config file can be my.ini or my.cnf; Default for Windows MariaDB is my.ini located at "C:\Program Files\MariaDB-[VERSION]\data\"

```
#
# Configuration to be inserted below last row of [mysqld] section
#
sql_mode=STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,N
O_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION
max_allowed_packet = 4M # must be adjusted with the value used on clients for
DICOM_SIZE
skip-external-locking
key_buffer_size = 16M
thread_cache_size = 64
lower_case_table_names = 1
table_open_cache = 64
tmp_table_size = 16M
read_buffer_size = 256K
read_rnd_buffer_size = 512K
join_buffer_size = 2M
sort_buffer_size = 2M
myisam_sort_buffer_size = 8M

[mysqldump]
quick
max_allowed_packet = 16M

[mysql]
no-auto-rehash

[isamchk]
key_buffer = 16M
sort_buffer_size = 16M
read_buffer = 2M
write_buffer = 2M

[myisamchk]
key_buffer = 16M
sort_buffer_size = 16M
read_buffer = 2M
write_buffer = 2M
```

### 3.1.3.3 MariaDB/MySQL - Networking

If deployed in a client-server, networked configuration, the database server must be configured to listen on the network interface connected to the local LAN; this can be achieved by editing the MariaDB/MySQL's config file, "my.cnf", and setting the "bind-address" parameter. For example, if you want the database server to listen on all the available network interfaces / IP addresses:

Change:

```
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address = 127.0.0.1
```

To:

```
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address = 0.0.0.0
```

In this way, the database server listens on any IP address (0.0.0.0) configured on the server and can serve the connecting clients.

In is also possible to specify a specific IP address, e.g:

```
bind-address = 192.168.1.100
```

## 3.1.4 Open Hospital - Database creation

To create the Open Hospital database, a MySQL client and a set of SQL script files placed under the sql/ folder are needed.

On a Microsoft Windows® system, MySQL Workbench® can be used as a client to connect to the database server and perform the needed tasks, while on Linux based systems the MySQL Administrator and MySQL Query Browser tools can be used.

The suggested approach is to use the command-line MariaDB/MySQL client which uses the same commands and syntax for every platform.

Open a terminal and navigate to the sql/ folder (e.g.):

```
C:\WINDOWS> cd C:\Users\OH\OpenHospital-v1.11.0\sql

C:\Users\OH\OpenHospital-v1.11.0\sql
```

and run the following command to connect to the database server (e.g.):

```
C:\Users\OH\OpenHospital-v1.11.0\sql> mysql -u root -p [-h hostname/ip address]
Enter password: *****
```

using the 'root' password chosen during the installation process. The terminal should reply with the MySQL client command line prompt:

```
mysql>
```

now we are ready to create the database user (isf) with password, the main OH database and grant the necessary permissions with the following SQL commands:

```
mysql>  CREATE DATABASE oh; CREATE USER 'isf'@'localhost' IDENTIFIED BY 'isf123';
        CREATE USER 'isf'@'%' IDENTIFIED BY 'isf123'; GRANT ALL PRIVILEGES ON oh.* TO
'isf'@'localhost'; \
        GRANT ALL PRIVILEGES ON oh.* TO 'isf'@'%';
mysql>  FLUSH PRIVILEGES;
```

To check if the database has been correctly created, enter the following command:

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| oh                 |
| performance_schema |
+--------------------+
```

The next step is to launch the main SQL database creation script with the command:

```
mysql> source create_all_en.sql
```

The data can be installed in different languages by using the related "create_all_xx.sql" file or with demo data using the "create_all_demo.sql" file (English only).

If the error message "ERROR 1148 (42000): The used command is not allowed with this MySQL version" appears the "local infile" command must be enabled on the client by logging in again and specifying the "--local-infile=1" parameter after the "-p"; thus "mysql -u root -p --local-infile=1".

The SQL script creates the OH database structure and populates it with default data.

Starting with version 1.11, it is also possibile to use the provided scripts (oh.sh/oh.bat) to automatically perform all the database creation tasks.

The 'isf' user, different from the 'root' one, is the user that the Open Hospital software uses to connect to the database and its password must be changed immediately with the command:

```
mysql>SET PASSWORD FOR isf'@'%' = PASSWORD('new-password-here');

Query OK, 1 row affected (0.00 sec)

Rows matched: 1 Changed: 1 Warnings: 0
```

And then:

```
mysql>FLUSH PRIVILEGES;

Query OK, 0 rows affected (0.00 sec)
```

```
mysql>quit;
```

> Remember to set the updated database password also in the Open Hospital configuration files and startup scripts.

### 3.1.5 Open Hospital - Configuration

Open Hospital comes with several configuration files called "properties files". To set up the database connection for Open Hospital, modify the *database.properties* and *log4j.properties* files, located in the "resource" folder rsc/.

Specify:

- the **IP address** of the machine where the database server was installed and containing the Open Hospital database. Replace "localhost" with the server IP address (e.g., 192.168.1.100)
- the "isf" user (or the user specified in the installation phase) **password**
- the MariaDB/MySQL Server **Port Number**, if changed from "3306"

> These settings are required for every Open Hospital installation/client.

More information on these files can be found in the sections database.properties and log4j.properties.

> Starting with version 1.11, it is possibile to use the provided scripts (oh.sh/oh.bat) to generate config files by choosing the option "-g".

### 3.1.6 Open Hospital - Startup and run

The Open Hospital software is now ready to be run by executing the proper launcher command:

- on a Linux machine: **oh.sh -C** (CLIENT mode)
- on a Windows machine: **oh.bat** (Select CLIENT mode from the interactive menu)

If everything is configured correctly, the Open Hospital main splash screen is presented:



And then the main Menu:



It is possible to create a shortcut (link) to the executable script on the desktop, in the Programs Menu or wherever is useful by using the provided icon "oh.ico" that is found in the package.



> If when running Open Hospital the application appears small-scaled or blurry then the issue is probably using a high-dpi display. In such cases, there are some solutions (click here) to apply only on the machines using this type of monitor.

## 3.2 Open Hospital - PORTABLE mode

Open Hospital is also distributed in a **PORTABLE** mode.

Since Open Hospital is a Java software program, it is portable by definition. The "PORTABLE" mode here refers to the possibility of having a self-contained copy of the OH software that includes Java, MariaDB/MySQL Server, and the database. This package can be copied and run everywhere, carrying together all the saved data.

Since the JRE and MariaDB/MySQL are not platform-independent and come in different versions and architectures, there exists:

- **Open Hospital - portable for Linux** platforms (embedding Java JRE and MariaDB for Linux, 32 or 64bit)

- **Open Hospital - portable for Windows** platforms (embedding Java JRE and MariaDB for Windows, 32 or 64bit)

Once the required version is downloaded, the application can be started by launching the script inside the package:

- **oh.sh** in the Linux version

- **oh.bat** in the Windows version

No other configuration is needed. Additional options can be shown by executing:

- on Linux: **oh.sh -h**

- on Windows: **oh.bat -h**

> To use Open Hospital in PORTABLE mode for Linux from a USB key it is required that the file permissions in the archive remain unchanged once extracted, otherwise the launcher cannot launch or use the application in an appropriate manner.

> Important: the PORTABLE mode is meant to try or test the software or to be used in a facility (like a dispensary) where only a single computer is available. If the facility is large and more clients/operators need to work on the same data, the full installation (client/server) in a networked architecture is recommended.

Unresolved directive in AdminManual.adoc - include::https://raw.githubusercontent.com/informatici/openhospital/master/oh-bundle/OH-README.md[lines=21..218,leveloffset=+2]

## 3.3 Backup & Restore

Backup of the Open Hospital database can be performed in several ways.

Starting with version 1.11, the easiest way is to use the provided scripts (oh.sh/oh.bat) by choosing the option "-s".

Restore of the Open Hospital database can be performed also in several ways.

Starting with version 1.11, the easiest way is to use the provided scripts (oh.sh/oh.bat) by choosing the option "-r".

### 3.3.1 Backup of database (alternative with MySQL Workbench®)

There are several ways to back up and restore **data** with MySQL. The recommended method is to use a common MySQL client such as MySQL Workbench®:

From https://stackoverflow.com/a/18379189:

1. Click on the Management tab on the left side of the Navigator Panel
2. Click on Data Export Option
3. Select Schema
4. Select Tables
5. Select the required option from the dropdown below the tables list as required
6. Select Include Create schema checkbox
7. Click on the Advance option
8. Select Complete insert checkbox in Inserts Panel
9. Start Export
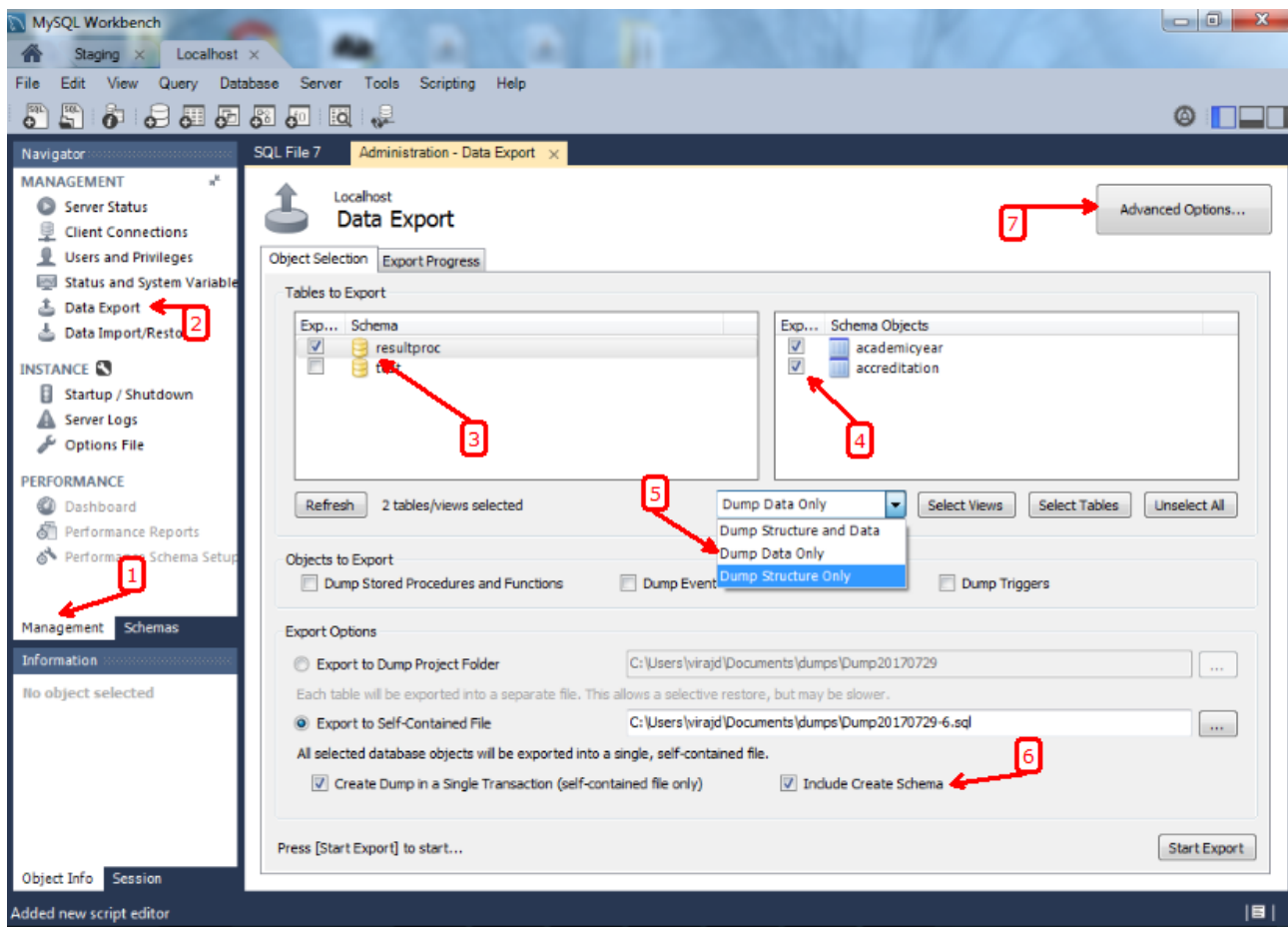
## 3.3.2 Restore of database (alternative with MySQL Workbench®)

To restore data with MySQL Workbench®:

1. Under **Server Administration** on the Home window select the server instance onto which the data will be restored (Create **New Server Instance** if doing it the first time).

2. Click on **Manage Import/Export**

3. Click on **Data Import/Restore** on the left side of the screen.

4. Select **Import from Self-Contained File** radio button (right side of the screen)

5. Select the path and the name of the restore sql file.

6. Click the **Start Import** button at the right bottom corner of the window.

## 3.3.3 Backup of client (data, code and settings)

Backup of the software **code and settings** are needed only if changes have been made to the "official version" to correctly perform any future Update process (see Update Open Hospital).

Backup of the client **data and configuration files**, especially for "PORTABLE" mode, can be performed by saving the **"data"** folder (including all subfolders) placed under the main OH installation folder.

# 4 Folders and directory structure

Open Hospital has a folder tree hierarchy which is shown here, where *<version>* is replaced with the current Open Hospital version, **<operating_system> with windows or linux and \*<arch>** with the OS architecture (32 or 64 bit):

- **OpenHospital-<version>-<operating_system>_<arch>-<client|portable>/**

- **bundle** – contains the **language** files

- **etc** – contains the DBMS configuration files

- **doc** – contains the Open Hospital **documentation**

- **lib** – contains the Java **libraries** needed to run the software

- **oh** – contains the Open Hospital application

- **oh/bin** – contains the Open Hospital **binaries** (compiled software)

- **oh/rpt** – contains the JasperReports® **reports** used in Open Hospital

- **oh/rpt/PDF** – contains the reports generated by Open Hospital in PDF format

- **oh/rsc** – contains the **resources** of Open Hospital, like configuration files and other related files

- **oh/rsc/icons** – contains the **icons** used in Open Hospital for windows and buttons

- **oh/rsc/images** – contains the **images** used in Open Hospital

- **data** – contains all the OH data - created after the first startup

- **data/db** – contains the database files

- **data/logs** – contains all the logs

- **data/dicom** – contains the dicom files

- **data/dump** – contains the database backups

- **sql** – contains the database creation scripts

- **tmp** – temporary folder

These folders may be organized differently depending on the version of the software or the architecture of the operating system.

Depending on the choosen operating system and architecture, the needed external applications and libraries (like MySQL/MariaDB and Java JRE) will be present also:

- *jdk-11.0.9+11-jre/* or *zulu11.50.19-ca-jre11.0.12-<operating_system>*<arch>/ - Java Virtual Machine

- *_mariadb-<version>-<arch>/* or *_mysql-<version>-<arch>-<build>/* - MySQL Server or MariaDB Server

# 5 Configuration Files

Customization of Open Hospital depends on settings set in text files called "properties files". Open Hospital comes with 8 main properties files:

- **oh/rsc/**
    - **database.properties** – for Open Hospital database connection
    - **dicom.properties** – for the DICOM viewer module
    - **examination.properties** – contains the settings for the 'examination' module
    - **log4j.properties** – for the logging system
    - **settings.properties** – for configuring Open Hospital options
    - **sms.properties** – for the SMS Manager module
    - **txtPrinter.properties**– for the text printing system
    - **xmpp.properties** – for the Xmpp Server

Furthermore, additional configurations files (not meant to be modified by the user) are:

- **resolutions.xml** – stores locally the information about webcams
- **version.properties** – contains the current Open Hospital version

Every configuration file works only <u>locally</u> and is meant to manage <u>only a single instance</u> of Open Hospital application.

This means that clients running Open Hospital may run differently according to their configuration files.

> ℹ  Any change to these files requires an application restart to apply the settings.

The following chapters will describe these properties files and their features in detail.

## 5.1 settings.properties

The default settings.properties file is:

```
# This file contains Open Hospital settings
LANGUAGE=en
SINGLEUSER=no
AUTOMATICLOT_IN=no
AUTOMATICLOT_OUT=no
AUTOMATICLOTWARD_TOWARD=no
LOTWITHCOST=yes
VISITSHEET=WardVisits
PATIENTSHEET=patient_clinical_sheet_ver3
EXAMINATIONCHART=patient_examination
OPDCHART=patient_opd_chart
ADMCHART=patient_adm_chart
DISCHART=patient_dis_chart
PATIENTBILL=PatientBill
BILLSREPORT=BillsReport
BILLSREPORTPENDING=BillsReportPending
BILLSREPORTMONTHLY=BillsReportMonthly
PHARMACEUTICALORDER=PharmaceuticalOrder
PHARMACEUTICALSTOCK=PharmaceuticalStock_ver4
PHARMACEUTICALSTOCKLOT=PharmaceuticalStock_ver5
PHARMACEUTICALAMC=PharmaceuticalAMC
PATIENTEXTENDED=yes
OPDEXTENDED=yes
MATERNITYRESTARTINJUNE=no
LABEXTENDED=yes
LABMULTIPLEINSERT=yes
INTERNALPHARMACIES=yes
MERGEFUNCTION=yes
SMSENABLED=no
INTERNALVIEWER=yes
MAINMENUALWAYSONTOP=no
RECEIPTPRINTER=yes
VIDEOMODULEENABLED=yes
PATIENTVACCINEEXTENDED=yes
ENHANCEDSEARCH=no
XMPPMODULEENABLED=no
DICOMMODULEENABLED=yes
DICOMTHUMBNAILS=yes
ALLOWPRINTOPENEDBILL=yes
ALLOWMULTIPLEOPENEDBILL=yes
PATIENTBILLGROUPED=PatientBillGrouped
PATIENTBILLSTATEMENT=PatientBillStatement
DEBUG=no
```

Every line is composed of a key (capitalized) and a value of the form:

key=value

Furthermore, values can be:

- **Boolean**: yes | no or true | false
- **String**: usually a filename or a country code (ISO 3166-1)

In following chapters will describe the meaning of each of the key/value pairs.

## 5.1.1 LANGUAGE

The following table shows the default value and the allowed ones:

| Key | Default Value | Allowed Values |
| --- | --- | --- |
| LANGUAGE | en | ar, en, it, fr, es, de, sw |

Open Hospital ships with nine translations, identified by the international country code:

- ar – Arabic
- de - German
- en – English
- es – Spanish
- fr – French
- it – Italian
- pt – Portuguese
- sw – Swahili
- zh_CN - simplified Chinese

To change the language used in the application simply edit *settings.properties* and change the value of this key. If an unknown value is set, the local computer language is applied.

> **i** An application restart is required to apply the setting.

## 5.1.2 SINGLEUSER
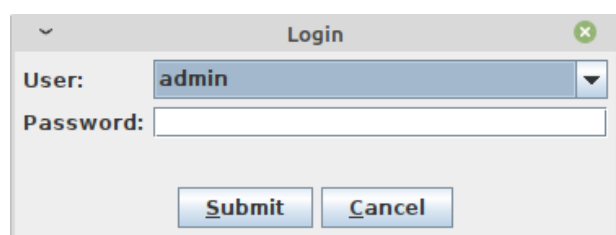
The following table shows the default value and the allowed ones:

| Key | Default Value | Allowed Values |
| --- | --- | --- |
| SINGLEUSER | yes | yes, no |

Open Hospital is designed to support multi-user usage. This means that is possible to define different users, arrange them by group and set different rights in application usage; for instance, a user "receptionist" may be able to register a new patient but not to prescribe a therapy.

Open Hospital has a predefined user called "admin" that has all rights to access all the functionality of the application. When SINGLEUSER is set to YES (default) the program starts with this user and all the program functionality is available.

If SINGLEUSER is set to NO the following login window appears when the program is started:



The default password for the "admin" user is "admin". After typing this password in the white field and pressing **Submit**, the main *Menu* with the functions are enabled.

Once logged in as the "admin" user defining additional users and/or groups is possible (see the Users & Groups chapter in the User's Guide), or just continue with a simple form of data protection.

> The setting is client-side specific, this means that the login can be disabled on a specific client, but logons will still be required on other clients, with the same defined user and group.

> Data security must never be left solely to the application but it must include proper network architecture and a rigid configuration of the clients.

### 5.1.3 AUTOMATICLOT

The following table shows the default value and the alternatives for lot management:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| AUTOMATICLOT_IN | no | yes, no |
| AUTOMATICLOT_OUT | no | yes, no |
| AUTOMATICLOTWARD_TOWARD | no | yes, no |

Open Hospital allows for automatic management of lots in the main pharmacy. This means that is possible to work in the pharmacy without taking care of lot definitions in loading the store and just specify preparation and expiring date for each charging movement (see 6.2.2.2 Insert stock charging movement in the User's Guide).

For discharging movement, the lot is automatically calculated according to the nearest expiring date of lots (FEFO - First Expiring First Out). If the quantity to discharge is larger than the quantity of the first selected lot, more than one discharging operation can be generated.

*AUTOMATIC LOTS MANAGEMENT* is split into two different settings for charging and discharging the main pharmacy and to manage lots also in the wards. In this way, more control is given to the user that may want to provide all the lots' details in *charging operations* but ask the application to automatically *discharge* the nearest expiring ones (FEFO). Similarly, in wards (see 6.3 Pharmaceuticals Stock Ward in the User's Guide) the user may decide to manage lots manually or to ask the application to do it automatically.

With **AUTOMATICLOT_IN** = **YES** the application generates a lot number automatically and only asks for an expiration date (mandatory). By default, AUTOMATICLOT_IN is set to NO. It is possible to change the value at any time.

With **AUTOMATICLOT_OUT** = **YES** the application discharges automatically the nearest expiring lot when required. By default, AUTOMATICLOT_OUT is set to NO. It is possible to change the value at any time.

With **AUTOMATICLOTWARD_TOWARD** = **YES**, the application discharges from a ward (to other wards) the nearest expiring lot automatically when required. By default, AUTOMATICLOTWARD_TOWARD is set to NO. It is possible to change the value at any time. For discharges to patients (drug giving) the lot is **never** asked.

> ⓘ An application restart is required to see the modification applied for all the above settings.

## 5.1.4 LOTWITHCOST

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|---|---|---|
| LOTWITHCOST | yes | yes, no |

Open Hospital allows for managing the cost of medicals in the main pharmacy (see 6.2.2.2 Insert stock charging movement in the User's Guide).

> ⓘ An application restart is required to see the modification applied.

## 5.1.5 PATIENTSHEET

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|---|---|---|
| PATIENTSHEET | patient_clinical_sheet | any kind of .jasper file name |

Open Hospital can produce a report about the clinical history of a patient (see 10.9 Clinical Sheet in the User's Guide).

By default, PATIENTSHEET is set to patient_clinical_sheet, that is, the filename of the related report to use for the Clinical Sheet functionality. It is possible to use a different report by installing it in the report folder (see Reports) and by changing this parameter.

> ⓘ An application restart is required to see the modification applied.

## 5.1.6 OPDCHART

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| OPDCHART | patient_opd_chart | any kind of .jasper file name |

Open Hospital can produce a report about the OPD chart of a patient (see 10.9 Clinical Sheet in the User's Guide).

By default, OPDCHART is set to patient_opd_chart which is the filename of the related report to use for the Clinical Sheet functionality **OPD Chart**. It is possible to use a different report by installing it in the report folder (see Reports) and by changing this parameter.

> ℹ️  An application restart is required to see the modification applied.

### 5.1.7 ADMCHART

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| ADMCHART | patient_adm_chart | any kind of .jasper file name |

Open Hospital can produce a report about the Admission of a patient (see 10.9 Clinical Sheet in the User's Guide).

By default, ADMCHART is set to patient_adm_chart, that is, the filename of the related report to use for the Clinical Sheet functionality **Admission Chart**. It is possible to use a different report by installing it in the report folder (see Reports) and by changing this parameter.

> ℹ️  An application restart is required to see the modification applied.

### 5.1.8 DISCHART

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| DISCHART | patient_dis_chart | any kind of .jasper file name |

Open Hospital can produce a report about the Discharge of a patient (see 10.9 Clinical Sheet in the User's Guide).

By default, DISCHART is set to patient_dis_chart which is the filename of the related report to use for the Clinical Sheet functionality **Discharge Chart**. It is possible to use a different report by installing it in the report folder (see Reports) and by changing this parameter.

> ℹ️  An application restart is required to see the modification applied.

### 5.1.9 PATIENTBILL

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| PATIENTBILL | PatientBill | any kind of .jasper file name |

Open Hospital can manage patient bills and produce an A4 format breakdown with his/her items and payments (see 8.2 Functions of Accounting in the User's Guide).

By default, PATIENTBILL is set to PatientBill, that is, the filename of the related report to use for the bill printing functionality. It is possible to use a different report by installing it in the report folder (see Reports) and by changing this parameter.

> **ℹ** An application restart is required to see the modification applied.

### 5.1.10 BILLSREPORT

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| BILLSREPORT | BillsReport | any kind of .jasper file name |

Open Hospital can manage patient bills and produce a report about all bills paid (or not paid) within a span of time (see **8.2 Functions of Accounting** in the *User's Guide*).

By default, BILLSREPORT is set to BillsReport, that is, the filename of the related report to use for the account printing functionality. It is possible to use a different report by installing it in the report folder (see Reports) and by changing this parameter.

> **ℹ** An application restart is required to see the modification applied.

### 5.1.11 BILLSREPORTMONTH

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| BILLSREPORTMONTH | BillsReportMonth | any kind of .jasper file name |

Open Hospital can manage patient bills and produce a report about all bills paid (or not paid) monthly (see 8.2 Functions of Accounting in the User's Guide).

By default, BILLSREPORTMONTH is set to BillsReportMonth, that is, the filename of the related report to use for the account printing functionality. It is possible to use a different report by installing it in the report folder (see Reports) and by changing this parameter.

> **ℹ** An application restart is required to see the modification applied.

## 5.1.12 PHARMACEUTICALORDER

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| PHARMACEUTICALORDER | PharmaceuticalOrder | any kind of .jasper file name |

Open Hospital can produce a report for the Pharmacy Stock critical levels which helps to identify which pharmaceuticals are running low and which ones need to be ordered (see 6.1 Pharmaceuticals in the User's Guide).

By default, PHARMACEUTICALORDER is set to PharmaceuticalOrder, that is, the filename of the related report to use for the pharmacy printing functionality. It is possible to use a different report by installing it in the report folder (see Reports) and by changing this parameter.

**i**     An application restart is required to see the modification applied.

## 5.1.13 PHARMACEUTICALSTOCK

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| PHARMACEUTICALSTOCK | PharmaceuticalStock | any kind of .jasper file name |

Open Hospital can produce a report for the Pharmacy Stock Status which identifies which pharmaceuticals are currently available (see **6.1 Pharmaceuticals** in the *User's Guide*).

By default, PHARMACEUTICALSTOCK is set to PharmaceuticalStock that is the filename of the related report to use for the pharmacy printing functionality. It is possible to use a different report by installing it in the report folder (see Reports) and by changing this parameter.

**i**     An application restart is required to see the modification applied.

## 5.1.14 PATIENTEXTENDED

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| PATIENTEXTENDED | yes | yes, no |

Open Hospital allows for the registration of a patient with extended or more detailed information (see 10.4 Insert a new Patient Extended in the User's Guide).

By default, PATIENTEXTENDED is set to yes. It is possible to set it to no to reduce the amount of data collected and so reduce the workload for the staff involved in data entry.

**i**     An application restart is required to see the modification applied.

## 5.1.15 OPDEXTENDED

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|-----|---------------|--------------|
| OPDEXTENDED | yes | yes, no |

Open Hospital can link every OPD visit to a registered patient to generate a comprehensive clinical history. For each visit, the user will be asked to select a registered patient or to register a new one, so each visit is attached to the patient's history (see **5.3 OPD Extended** and **10.9 Clinical Sheet** in the *User's Guide*).

By default, OPDEXTENDED is set to yes. It is possible to set it to no to reduce the amount of data to be collected (only age and sex, no patient registration) and so reduce the workload for the staff involved in data entry.

**ℹ** An application restart is required to see the modification applied.

## 5.1.16 MATERNITYRESTARTINJUNE

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|-----|---------------|--------------|
| MATERNITYRESTARTINJUNE | no | yes, no |

Open Hospital allows changing the way the admissions in Maternity ward are counted within the year; in Open Hospital the first admission of the year for every ward has progressive number 1 (one) and it increments itself automatically up to the end of the year; in some facilities, this is partially true and the progressive numbering starts from June only for the Maternity ward. This option controls which behavior is followed.

By default, MATERNITYRESTARTINJUNE is set to no.

**ℹ** An application restart is required to see the modification applied.

## 5.1.17 LABEXTENDED

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|-----|---------------|--------------|
| LABEXTENDED | yes | yes, no |

Open Hospital can force every laboratory exam to be linked to a registered patient, to have a comprehensive clinical history. For each exam, the user will be asked to select a registered patient so from that moment the exam will be attached to the patient's history (see 7.3.2 New Laboratory Exam in the User's Guide).

By default, LABEXTENDED is set to yes. Anyway, is possible to set it to no to reduce the amount of data to be collected (only name, age, sex, no patient registration) and so reduce the workload for the staff involved in data entry.

ℹ  An application restart is required to see the modification applied.

### 5.1.18 LABMULTIPLEINSERT

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|-----|---------------|--------------|
| LABMULTIPLEINSERT | yes | yes, no |

Open Hospital can insert multiple laboratory tests per patient at one time to avoid the repetitive operation of selecting a patient or writing his/her name; note that the LABEXTENDED option must also be set to yes (see 7.3.3 Laboratory Multiple Insert in the User's Guide).

By default, LABMULTIPLEINSERT is set to yes, but if LABEXTENDED is set to no it will be just ignored.

ℹ  An application restart is required to see the modification applied.

### 5.1.19 INTERNALPHARMACIES

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|-----|---------------|--------------|
| INTERNALPHARMACIES | yes | yes, no |

Open Hospital can register all dispensing to patients within a ward. Activating this option, the Pharmaceutical Stock Ward functionality will be available in the application (see Merge function in the User's Guide).

By default, INTERNALPHARMACIES is set to yes.

ℹ  An application restart is required to see the modification applied.

### 5.1.20 MERGEFUNCTION

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|-----|---------------|--------------|
| MERGEFUNCTION | no | yes, no |

Open Hospital can merge two patient's histories into one. This is particularly useful in the case of double registration where different information was collected over time. Activating this option, the

Merge functionality will be available in the Admission/Patient module (see Merge function in the User's Guide).

By default, MERGEFUNCTION is set to no.

> ℹ   An application restart is required to see the modification applied.

## 5.1.21 INTERNALVIEWER

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| INTERNALVIEWER | yes | yes, any executable filename in the path or with an absolute path |

Open Hospital can use a different PDF reader for generated reports. If this parameter is set to yes, the Jasper PDF Viewer is used (see Report Launcher in the User's Guide). If another PDF viewer is required, specify the executable's filename. One must ensure that the executable is available in the desktop environment.

**Example in Windows**:

```
INTERNALVIEWER=AcroRd32.exe

INTERNALVIEWER= C:\\Program Files (x86)\\Adobe\\Reader 10.0\\Reader\\AcroRd32.exe
```

> ℹ   Please note the double file and folder separator "\\".

**Example in Linux**:

```
INTERNALVIEWER=evince
```

By default, INTERNALVIEWER is set to yes.

> ℹ   An application restart is required to see the modification applied.

## 5.1.22 SMSENABLED

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| SMSENABLED | no | yes, no |

Open Hospital can indicate whether information can be sent to patients via SMS notifications. Enabling or disabling this option only affects the possibility to set, or not set, a therapy as notifiable (see 10.10.7 Notify & SMS in the User's Guide).

By default, SMSENABLED is set to no.

### 5.1.22.1 Configuration

To set up a GSM device, the GSM mode has to be defined in the sms.properties file (see sms.properties).

The default settings for a GSM modem should work with the majority of GSM devices (like phones, smartphones, USB modems, etc...). The exception is the PORT parameter which must be changed to match the SERIAL port address used by the device plugged into the system.

If the PORT param is correct but Open Hospital is not able to communicate with the device (try switching to DEBUG log level – or lower - during this setup – see log4j.properties) and try to use the SetupGSM utility (OH startup script with the "G" option: oh.sh -G / oh.bat → select "G"). (see Folders).

### 5.1.22.2 SetupGSM

To use the SetupGSM utility just follow these instructions:

1. Plug the device into the system and be sure the system recognizes it, load a proper driver for it and assign a serial port (COM)

2. Launch the utility: the utility will scan all plugged devices and will try to recognize the modem (or phone) within them

3. If the device is recognized as a modem confirmation message prompt message as follows appears:
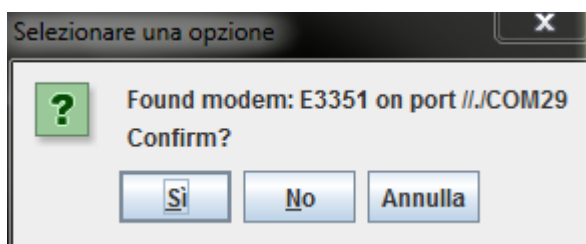


4. If the name of the device is the one meant to be used with the system then just click "yes", otherwise click "no", and the scan will continue

5. Once "yes" is selected the sms.properties file is automatically modified by the utility with the proper port address

## 5.1.23 MAINMENUALWAYSONTOP

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| MAINMENUALWAYSONTOP | no | yes, no |

Open Hospital can keep the main menu always on top so it cannot be overlapped or hidden by other windows.

By default, MAINMENUALWAYSONTOP is set to no.

> **i**  An application restart is required to see the modification applied.

### 5.1.24 RECEIPTPRINTER

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|---|---|---|
| RECEIPTPRINTER | no | yes, no |

Open Hospital can manage text or ZPL printers with aim of printing simple receipts in the Accounting module (see 8.2.1.11 Print receipt function in User's Guide). Once this option is activated, additional parameters are needed in the **txtPrinter.properties file** to adjust the output with the printer connected to the system (see txtPrinter.properties in this document).

By default, RECEIPTPRINTER is set to no.

> **i**  The text printer must be set as the default printer.

> **i**  An application restart is required to see the modification applied.

### 5.1.25 VIDEOMODULEENABLED

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|---|---|---|
| VIDEOMODULEENABLED | no | yes, no |

Open Hospital can drive any webcam with the aim of capturing patient's images in the Admission/Patient module (see 10.4.3 Patient Photo function in User's Guide). Once this option is activated the related button, **New Photo** in the **New Patient** window is available to start and use the webcam.

By default, VIDEOMODULEENABLED is set to no.

> **i**  An application restart is required to see the modification applied.

### 5.1.26 PATIENTVACCINEEXTENDED

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|---|---|---|
| PATIENTVACCINEEXTENDED | yes | yes, no |

Open Hospital can register vaccines given to patients (see 9 Vaccines in the User's Guide). This option allows toggling the patient's full name in the **Patient Vaccine Browser** window.

By default, PATIENTVACCINEEXTENDED is set to yes.

> ℹ️ An application restart is required to see the modification applied.

## 5.1.27 ENHANCEDSEARCH

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|---|---|---|
| ENHANCEDSEARCH | no | yes, no |

Open Hospital can optimize memory usage when the number of registered patients becomes large (see 10.2.2 Search patient Enhanced in the User's Guide). With the enhanced search only, the patient matching a search criterion will be loaded in the memory, otherwise, all patients registered in the system will be loaded in the **Patient Browser** window.

By default, ENHANCEDSEARCH is set to no. It is possible to toggle this option anytime to reduce the amount of memory needed by the computer or the server.

> ℹ️ An application restart is required to see the modification applied.

## 5.1.28 XMPPMODULEENABLED

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|---|---|---|
| XMPPMODULEENABLED | no | yes, no |

Open Hospital embeds a technology that allows users logged into the application to chat and share information related to the hospital activities (see 13 Communication in the User's Guide); the SINGLEUSER option must be set to no too. If this option is active, Open Hospital will look for an XMPP Server at startup time. The **xmpp.properties** will be used to point to the server that must be installed on the computer or on the OH Server by the System Administrator (see xmpp.properties in this document).

By default, XMPPMODULEENABLED is set to no, but if SINGLEUSER is set to yes XMPPMODULEENABLED is ignored.

> ℹ️ An application restart is required to see the modification applied.

## 5.1.29 DICOMMODULEENABLED

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| DICOMMODULEENABLED | no | yes, no |

Open Hospital includes a feature that allows attaching DICOM files to a patient's clinical sheet. When this option is enabled, a button, **DICOM**, appears in the patient Clinical sheet module.

By default, DICOMMODULEENABLED is set to no.

> **i** An application restart is required to see the modification applied.

### 5.1.29.1 DICOMTHUMBNAILS

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| DICOMTHUMBNAILS | yes | yes, no |

Open Hospital can enable or disable thumbnails in the Dicom Viewer.

By default, DICOMTHUMBNAILS is set to yes.
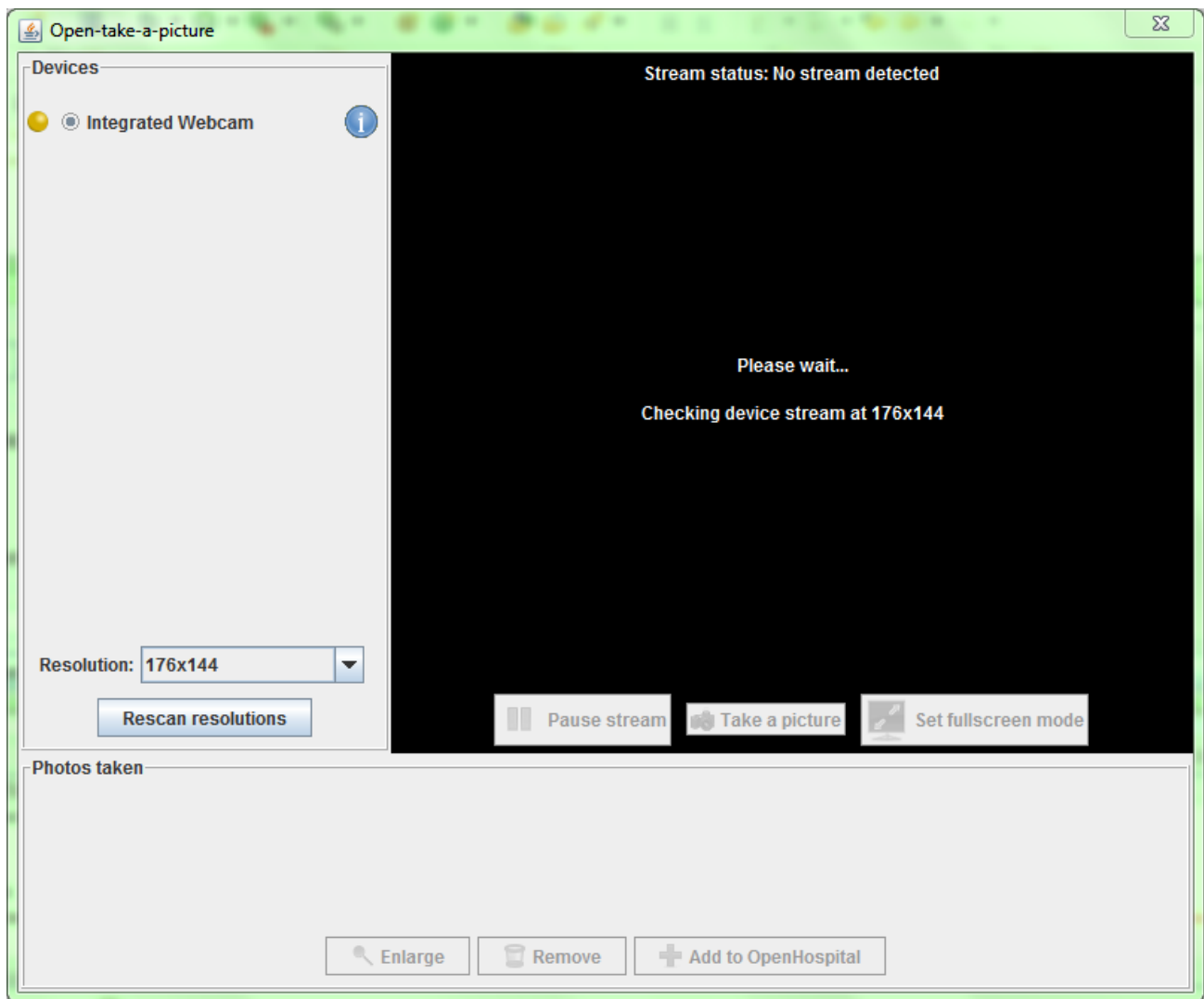
> **i** An application restart is required to see the modification applied.

## 5.1.30 DEBUG

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
| --- | --- | --- |
| DEBUG | no | yes, no |

Open Hospital can run in debug mode when it would be useful to understand what is happening behind the user graphical interface. Currently, this option affects only the video module, which means that in place of the normal **New Patient** window (see 10.4.3 Patient Photo function in User's Guide), the program opens the new special window shown below:

From this window is possible to have more information about webcams connected to the system. One can try different resolutions to find the best one or learn more information about a problem.

Once a photo is produced, Open Hospital remembers the settings and uses them in the future.

By default, DEBUG is set to no.

> ℹ An application restart is required to see the modification applied.

### 5.1.31 ALLOWMULTIPLEOPENEDBILL

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|---|---|---|
| ALLOWMULTIPLEOPENEDBILL | no | yes, no |

Open Hospital allows a patient to have multiple open bills (invoices). If this option is enabled, when creating a new bill for a patient, if that patient already has an open bill the user is asked to confirm the creation of another one.

By default, ALLOWMULTIPLEOPENEDBILL is set to no.

An application restart is required to see the modification applied.

### 5.1.32 OPENEDBILLSREPORT

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|---|---|---|
| OPENEDBILLSREPORT | OH023_BillsReportMonth | OH023_BillsReportMonth |

Open Hospital can print the list of open bills. The parameter contains the jasper report file name to print the list of open bills.

By default, OPENEDBILLSREPORT is set to OH023_BillsReportMonth.

An application restart is required to see the modification applied.

### 5.1.33 ALLOWPRINTOPENEDBILL

The following table shows the default value and the allowed ones:

| Key | Default Value | Valid Values |
|---|---|---|
| ALLOWPRINTOPENEDBILL | no | yes, no |

Open Hospital can print the receipt of a single open invoice. When the parameter is enabled, in the Accounting module under the pending tab one can select an open bill and get the receipt by clicking the Receipt button in the buttons panel.

By default, ALLOWPRINTOPENEDBILL is set to no.

An application restart is required to see the modification applied.

# 5.2 database.properties

Open Hospital comes with a database.properties file with the following defaults:

```
jdbc.drivers=com.mysql.jdbc.Driver
jdbc.url=jdbc:mysql:
jdbc.server=localhost
jdbc.db=oh
jdbc.username=isf
jdbc.password=isf123
#jdbc.port=3306
```

The first two parameters are not meant to be modified by the user nor by the Administrator. The remaining five parameters may need modification according to the network settings.

- **jdbc.drivers**: the Java library used to connect to a database server

- **jdbc.url**: the address to point to the MySQL Server

- **jdbc.server**: the IP address of the databse server (where MariaDB/MySQL Server is installed); localhost means the same computer where this file is located; others values can be:
  *127.0.0.1* – loopback network
  *192.168.0.100* – an example of a private IP address (LAN)
  *217.147.110.117* – an example of a public IP address (WAN)

- **jdbc.db**: the name of the Open Hospital database

- **jdbc.username**: the database user allowed to connect and operate on the database

- **jdbc.password**: the user password

- **jdbc.port**: the TCP port where the database server is listening; by default, this port is 3306 but can be changed if needed to match the installation

The full address for the database using the defaults then is:

jdbc:mysql://localhost/oh

> ℹ An application restart is required to see the modification applied.

> ℹ The same settings must be applied in log4.properties, see log4j.properties in this manual.

## 5.3 dicom.properties

Open Hospital comes with a dicom.properties file set as follows by default:

```
#dicom.manager.impl=org.isf.dicom.manager.FileSystemDicomManager        for filesystem
#dicom.manager.impl=org.isf.dicom.manager.SqlDicomManager               for database
#dicom.max.size=1024B, 2048B, 1M, 16M, 256M, 512M, 1024M, 1G            size examples
dicom.manager.impl=org.isf.dicom.manager.SqlDicomManager
dicom.storage.filesystem=E:\\ISF\\DICOM\\OH_storage
dicom.max.size=4M
```

- **dicom.manager.impl**: the Java class that implements org.isf.dicom.manager.DicomManagerInterface and that manages the DICOM persistence operation. This property accepts on of the following:

  - org.isf.dicom.SqlDicomManager: The dicom file is stored in the database

  - org.isf.dicom.FileSystemDicomManager: The dicom file is stored in the file system

- **dicom.storage.filesystem**: if the value of the first parameter is org.isf.dicom.FileSystemDicomManager, this property specifies the path where the DICOM files are stored.

- **dicom.max.size**: is the maximum allowed size for a dicom/jpg image (the MariaDB/MySQL server setting max_allowed_packet should be set with the same or a larger value). If not

specified the default value of "4M" is used.

# 5.4 examination.properties

Open Hospital comes with the examination.properties file used in the examination module. The file is set as follow by default:

```
# This file contains PatientExamination module settings
LIST_SIZE = 10
HEIGHT_MIN = 0
HEIGHT_MAX = 250
HEIGHT_INIT = 0
#HEIGHT_STEP = 1
#WEIGHT_UNIT = kg
WEIGHT_MIN = 0
WEIGHT_MAX = 200
WEIGHT_INIT = 0
WEIGHT_STEP = 0.1
#AP_UNIT = mmHg
AP_MIN_INIT = 80
AP_MAX_INIT = 120
#HR_UNIT = bpm
HR_MIN = 0
HR_MAX = 240
HR_INIT = 60
#TEMP_UNIT = *C
TEMP_INIT = 36
TEMP_MIN = 30
TEMP_MAX = 50
TEMP_STEP = 0.1
#SAT_UNIT = %
SAT_INIT = 98
SAT_MIN = 50
#SAT_MAX = 100
SAT_STEP = 0.1
HGT_MIN = 30
HGT_MAX = 600
HGT_INIT = 80
DIURESIS_MIN = 0
DIURESIS_MAX = 2500
DIURESIS_INIT = 100
RR_INIT = 20
RR_MIN = 0
RR_MAX = 100
```

- **LIST_SIZE**: the maximum number of examinations that can be viewed in the history. Must be less than or equal to 10

- **HEIGHT_MIN**: the height minimum value (in cm), used to initialize the height slider in the

*Patient Examination* window

- **HEIGHT_MAX**: the height maximum value (in cm), used to initialize the height slider in the *Patient Examination* window

- **HEIGHT_INIT**: the height default value (in cm)

- **WEIGHT *MIN*: the weight minimum value (in Kg), used to initialize the Weight slider in the _Patient Examination** window

- **WEIGHT *MAX*: the weight maximum value (in Kg), used to initialize the Weight slider in the _Patient Examination** window

- **WEIGHT _INIT**: the weight default value (in Kg)

- **WEIGHT _STEP**: the step (in Kg) used when moving the weight slider

- **AP _MIN**: the Arterial pressure minimum value (in mmHg)

- **AP _MAX**: the Arterial pressure maximum value (in mmHg)

- **HR _MIN**: the Heart rate minimum value (in bmp)

- **HR _MAX**: the Heart rate maximum value (in bmp)

- **HR _INIT**: the Heart rate default value (in bmp)

- **TEMP _INIT**: the temperature default value (in °C)

- **TEMP _MIN**: the temperature minimum value (in °C)

- **TEMP _MAX**: the temperature maximum value (in °C)

- **TEMP _STEP**: the temperature step (in °C) used when moving the slider

- **SAT _INIT**: the saturation default value (%)

- **SAT _MIN**: the saturation minimum value (%)

- **SAT _STEP**: the saturation step (%) used when moving the slider

- **HGT_INIT**: = the Hemo Glucose Test default value (in mg/dl)

- **DIURESIS_INIT**: = the Daily Urine Volume default value (in ml)

# 5.5 log4j.properties

The default log4j.properties file is:

```
# global logging to RollingFile (logs/ folder), available levels INFO, DEBUG, FINEST
(debug++)
log4j.rootCategory=INFO,RollingFile

# Null appender (off)
log4j.appender.null=org.apache.log4j.varia.NullAppender

# StdOut Appender (with classes) (not used)
log4j.appender.StdOut = org.apache.log4j.ConsoleAppender
log4j.appender.StdOut.layout=org.apache.log4j.PatternLayout
log4j.appender.StdOut.layout.ConversionPattern=[%d{dd/MMM/yyyy HH:mm:ss}]
[%X{OHUserGroup}:%X{OHUser}] %-p - %m%n

# File Appender (with classes), daily rotation
log4j.appender.RollingFile = org.apache.log4j.DailyRollingFileAppender
log4j.appender.RollingFile.DatePattern='.'yyyy-MM-dd
log4j.appender.RollingFile.File=logs/openhospital.log
log4j.appender.RollingFile.layout=org.apache.log4j.PatternLayout
log4j.appender.RollingFile.layout.ConversionPattern=[%d{dd/MMM/yyyy HH:mm:ss}]
[%X{OHUserGroup}:%X{OHUser}] %-p - %m (%l)%n

# DB Appender (table columns)
log4j.appender.DB=org.apache.log4j.jdbc.JDBCAppender
log4j.appender.DB.URL=jdbc:mysql://localhost:3306/oh?autoReconnect=true
log4j.appender.DB.user=isf
log4j.appender.DB.password=isf123
log4j.appender.DB.sql=INSERT INTO LOG (LOG_TYPE, LOG_CLASS, LOG_METHOD, LOG_TIME,
LOG_MESS, LOG_USER) VALUES (1, '%C', '%M', '%d{yyyy-MM-dd HH:mm:ss}', LEFT('%m',
1024), '%X{OHUser}')
log4j.appender.DB.layout=org.apache.log4j.PatternLayout

# Assigning appenders to packages (application loggers)
log4j.category.org.isf=DEBUG,RollingFile,StdOut
log4j.additivity.org.isf = false

# Assigning appenders to Hibernate packages (DB loggers)
# - hibernate.SQL to DEBUG for SQL queries to be logged
# - hibernate.type to TRACE for queries parameters to be logged with "binding
parameter [?]"
log4j.logger.org.hibernate=INFO,RollingFile,StdOut
log4j.logger.org.hibernate.SQL=INFO,RollingFile,StdOut
log4j.logger.org.hibernate.type=INFO,RollingFile,StdOut
```

This file is not meant to be modified by the user.

Once the system is up and running, the log level can be reduced from "DEBUG" to "INFO", so the log files are significantly reduced in size:

```
log4j.category.org.isf=INFO,RollingFile
```

Alternatively, in case of malfunctions of the application, the log level can be increased to "TRACE" so more detail on the operation of the application can be recorded:

```
log4j.category.org.isf=TRACE,RollingFile
```

Logging can be routed to the database (DB) by adding the respective **appender**:

```
log4j.category.org.isf=INFO,RollingFile,DB
```

Logging can also be routed to standard output (the console window) with:

```
log4j.category.org.isf=INFO,RollingFile,StdOut
```

If the DB appender is specified then the configuration must be set to match those in the database.properties file (see database.properties). The most important properties are those highlighted in **red** below:

```
# DB Appender (table columns)
log4j.appender.DB=org.apache.log4j.jdbc.JDBCAppender
log4j.appender.DB.URL=jdbc:mysql://localhost:3306/oh?autoReconnect=true
log4j.appender.DB.user=isf
log4j.appender.DB.password=isf123
```

> ℹ An application restart is required to see the effect of any applied parameter modification.

# 5.6 sms.properties

The default sms.properties file is:

```
################################################################
# Global configuration
################################################################
# use: gsm-gateway-service || skebby-gateway-service || textbelt-gateway-service
sms.gateway=textbelt-gateway-service
sms.gateway.thread.timeout=3000
sms.gateway.thread.loop=15
sms.gateway.thread.icc=+39


... other sections
```

- **sms.gateway**: the SMS sender mode. Can take three values:

- **gsm-gateway-service**: requires a GSM modem connected to a COM port. The administrator should configure the SMSGateway/GSM.properties file to set the suitable PORT.

  - **skebby-gateway-service**: requires a Skebby provider account.

  - **textbelt-gateway-service**: requires a Textbelt provider account.

- **TIMEOUT**: the timeout for the HTTP request. Required when sms.gateway is set to skebby-gateway-service or textbelt-gateway-service.

- **LOOP**: the delay used by the sender to fetch new SMS operations. The value is in seconds.

- **ICC**: the International Country Code that needs to be added to the phone numbers (if missing).

## 5.6.1 skebby-gateway-service

The default skebby-gateway-service section is:

```
################################################################
# Skebby configuration
################################################################
skebby-gateway-service.username=
skebby-gateway-service.password=
skebby-gateway-service.ribbon.base-url=https://api.skebby.it:443
# USER_KEY and ACCESS_TOKEN avoids the login call every time we need to send sms
skebby-gateway-service.accessToken=
skebby-gateway-service.userKey=
```

This file defines the parameters for sending SMS using the Skebby HTTP API.

- **username**: the username for authenticating to the Skebby server.

- **password**: the password for authenticating to the Skebby server.

- **ribbon.base-url**: the URL of the Skebby HTTP API.

- **accessToken**: the token generated by the Skebby provider upon user request (it replaces username)

- **userKey**: the key generated by the Skebby provider upon user request (it replaces password)

## 5.6.2 textbelt-gateway-service

The default textbelt-gateway-service section is:

```
################################################################
# Textbelt configuration
################################################################
# enables/disables server testing mode (so that textbelt will do fake actions)
textbelt-gateway-service.enable-testing-mode=false
# use: textbelt (in order to send 1 free sms per day) or your api key (if you
purchased sms)
textbelt-gateway-service.key=textbelt
textbelt-gateway-service.ribbon.base-url=https://textbelt.com:443
```

This file defines the parameters for sending SMS using the Skebby HTTP API.

- **enable-testing-mode**: if set to "true", textbelt will do fake actions upon user/application requests.

- **key**: the API key (if using a purchased sms) from Textbelt provider (one can use "textbelt" to send 1 free sms per day)

- **ribbon.base-url**: The URL of the Textbelt HTTP API.

# 5.7 txtPrinter.properties

The default txtPrinter.properties file is:

```
# This file contains text printing information
# MODE = TXT, PDF or ZPL
USE_DEFAULT_PRINTER=yes
PRINT_AS_PAID=no
PRINT_WITHOUT_ASK=no
MODE=PDF
#TXT_CHAR_HEIGHT=10
#TXT_CHAR_WIDTH=10
ZPL_FONT_TYPE=0
ZPL_ROW_HEIGHT=25
```

This file is used only if the RECEIPTPRINTER flag is enabled (see RECEIPTPRINTER in this document).

The file has seven parameters:

- **USE_DEFAULT_PRINTER**: If it is set to *yes*, Open Hospital will use the default printer, otherwise it will show the system printing dialog allowing a change to the printer at every print attempt:
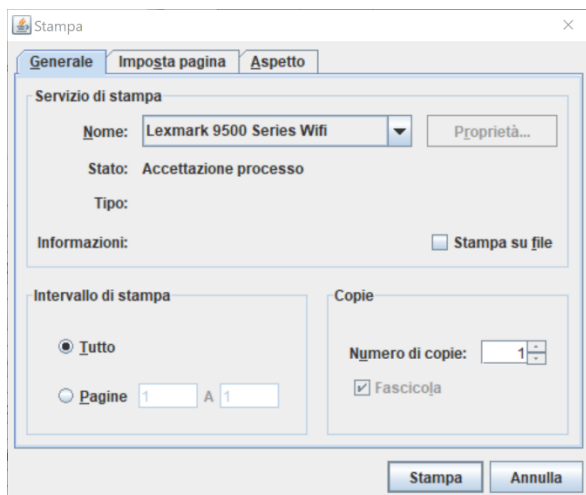
- **PRINT_AS_PAID**: This changes the behavior when a Bill is set to PAID in the Accounting module (see 8.2.1.11 Print receipt function in User's Guide).
  - If it is set to *yes*, Open Hospital will try to print a receipt just after a Bill is set to PAID
    - A confirmation window will appear (see below), otherwise, it must be done after, by pressing the **Edit Bill** button on the already closed ("C") bill
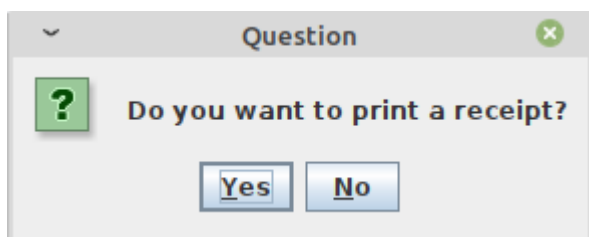


- **PRINT_WITHOUT_ASK**: if set to yes Open Hospital will try to print the receipt without the confirmation window
- **MODE**: the MODE can take three values:
  - **TXT**: the bill is printed as pure text (no graphics)
  - **PDF**: the bill will be printed as a PDF
  - **ZPL**: the text printer connected to the system works with ZPL language (an EPL evolution for Zebra Label Printer)
- **ZPL_FONT_TYPE**: a 0 (zero) value stands for a standard character; the value can be changed to "A", "B", "C", etc. according to the device datasheet, to obtain a better look
- **ZPL_FONT_SIZE**: an integer value to adjust the character size to obtain a better look

The best way to set these parameters for the device is to set them one by one and check the different results until the best fit is reached.

An application restart is required to see the modification applied.

# 5.8 xmpp.properties

Open Hospital comes with the xmpp.properties file set as follows by default:

```
# This file contains Xmpp Server information
DOMAIN=127.0.0.1
PORT=5222
```

This file is used only if the XMPPMODULEENABLED flag is enabled (see XMPPMODULEENABLED in this document).

To use the Communication module (see 13 Communication in the User's Guide), install an XMPP Server separately on the OH Server, typically on the same machine as the database server.

The most common XMPP Server is the free and open-source project Ignite Openfire® (http://www.igniterealtime.org/projects/openfire/) available for Linux, Windows, and Mac.

Once the XMPP Server is installed, active and running set the two parameters in the xmpp.properties file as follows:

- **DOMAIN** – the IP address of the XMPP server (it could be the same as the database server)

- **PORT** – 5222 or another if set differently

> ℹ️ An application restart is required to see the modification applied.

## 5.8.1 OpenFire Settings

The XMPP module needs some knowledge about the XMPP protocol and how an XMPP server works.

Every new log into Open Hospital creates an OpenFire user with the same username and password (even if passwords are saved differently in OH and OpenFire DB). For this reason, the OH "admin" user may conflict with OpenFire one not allowing the log in as admin and to chat.

> ℹ️ If the admin user needs to chat with other users consider creating another "superuser" in OH under the "admin" group.

To allow users to see each other and communicate via XMPP protocol, they must belong to the same OpenFire "group" and have the other users in their own "*roster*" (find more comprehensive information about "group" and "roster" in the online documentation).

Please follow these settings:

1. Install OpenFire server (better if from the zip file) on a writable path

2. Set in both OpenFire and in the xmpp.properties file an IP address rather than the FQDN (e.g., 127.0.0.1)

3. Set OpenFire to use Embedded DB

4. Create a group "OH" in the OpenFire server and enable the "Contact List (Roster) Sharing"

5. Add new users automatically created from OH to the "OH" group

6. Consideration should be given to using the OpenFire plugin to automatically add new users to a

group (e.g., registration plugin
[http://www.igniterealtime.org/projects/openfire/plugins/registration.jar](http://www.igniterealtime.org/projects/openfire/plugins/registration.jar))

# 5.9 Bundles

Bundles are the language (or translation) files provided with Open Hospital. As described in the LANGUAGE chapter, Open Hospital comes with different available languages in the folder **bundles/** within the package file:

**language_XY.properties**

where XY is an international country code.

These files are property files (a text file), which contain multiple **key** = **value** pairs containing localized text for the language specified.

All files must be encoded in **UTF-8** to accept any language-specific characters (e.g. è ì ò à ñ ú ù ¡ ¿ …).

## 5.9.1 New Translations

To create a new translation, simply copy the English file, rename it with the new country code in place of the "XY" in the filename, and start translating with a simple text editor. A text editor can also be used to modify existing translations that are incorrect.

For instance, if to create an Arabic translation, copy a new bundle from the English one by copying *language_en.properties* to a **new file** *language_ar.properties*.

Then edit the Arabic bundle file and set the LANGUAGE parameter to **ar** to start testing the translation.

# 6 Reports

Reports in Open Hospital are produced with JasperReports® technology that allows one to design a report in a WYSIWYG (What You See Is What You Get) way, connect it to a datasource (DB), test it, modify it, and then compile it to use it in the application software.

Open Hospital reports are all contained in reports rpt/ folder (see Folders in this document). Each report consists of two files:

- a **.jrxml** file: JasperReport XML file, that can be modified with a proper editor

- a **.jasper** file: Jasper file, that is the compiled version that can be run in Open Hospital; this file is produced on the jrxml base

- some reports could have also one or more **.properties** file that works for the localization, it means that the report has been translated to appear in more languages (default is English)

The editor to create and modify JasperReports® is TIBCO Jaspersoft® Studio version 6.14.0 or later, a free and open-source software.

With Jaspersoft® Studio it is possible to edit Open Hospital reports and re-compile them in the same location to be found and used by the Open Hospital application.

# 7 Installing Open Hospital 1.11.0 in Eclipse EE

⚠️ The code is available on GitHub at these links: openhospital-core, openhospital-gui, and openhospital-doc.

*Please use EGit (Eclipse Plugin for Git) to clone the code into the Eclipse instance.*

Assuming Eclipse EE, MariaDB/MySQL and the Java Virtual Machine 1.8 or higher (Java8 or greater) are already installed on the computer, this chapter will focus on cloning version 1.11.0 into the Eclipse J2EE (or Eclipse EE) environment.

Please follow these EGit User Guide - Working with remote Repositories.

## 7.1 Run the Project

ℹ️ Before running the project, the DB must be created as explained in chapter MySQL Server and Open Hospital DB.

To run the application, look for class *openhospital-gui/src/main/java/org/isf/menu/gui/Menu.java*, then right-click on it and choose *Run As → Java Application*. If everything is OK, the splash screen will appear. If not, look in the Eclipse console for any error or warning messages.



Open Hospital's advanced features require "native" libraries generated to match the computer's operating system. Right-click on the project's name, then select *Run As → Run Configurations*:
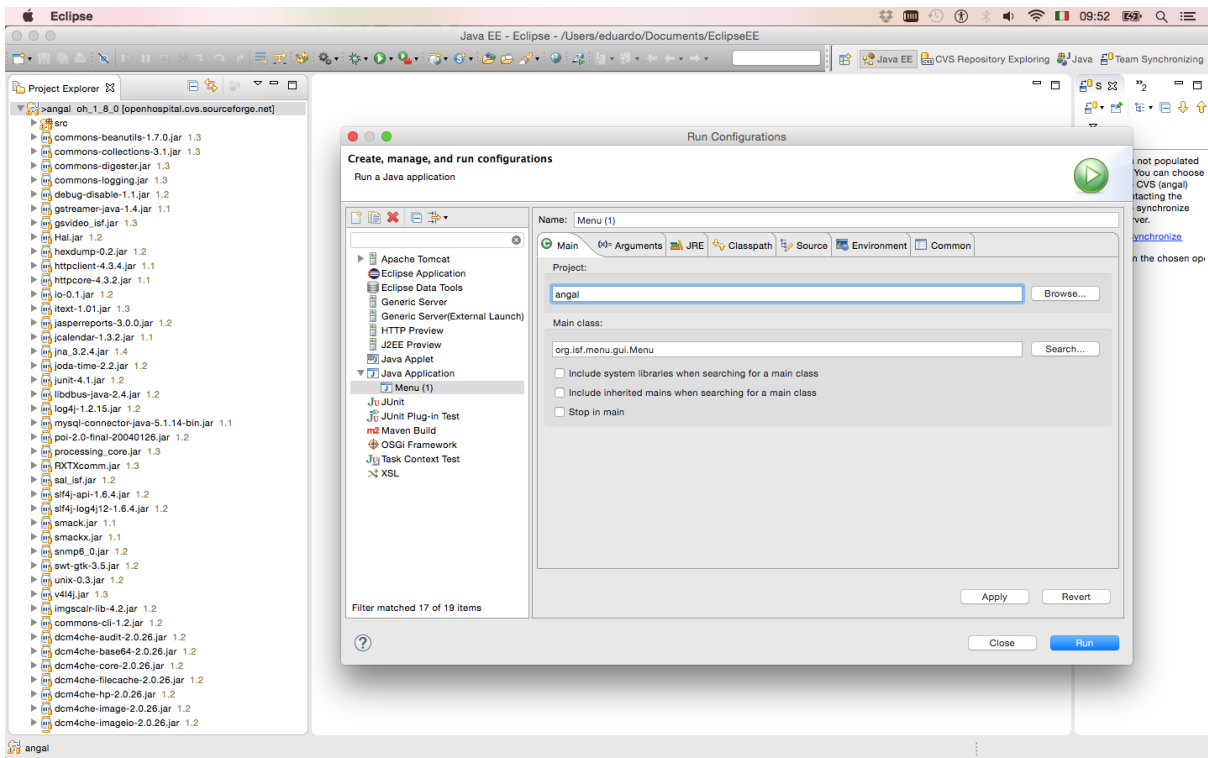
A "Run Configurations" window opens. Choose Java Application → <application's name> - Menu (1) on this example. Select the "Main" tab, write "angal" on the "Project" field and check if the "Main class" is org.isf.menu.gui.Menu.

Select the "Arguments" tab and fill the "VM arguments" path typing (on a single line):

```
-Dsun.java2d.dpiaware=false -Djava.library.path=lib\native\<OS name>
```

<OS name> can be "Mac_OS_X", "Solaris", "Linux", "Win64" (for 64-bit Windows) or "Windows" (for 32-bit Windows) according to the folder's schema found in the related folder "native". Click on "Run" to close the window and run the application.
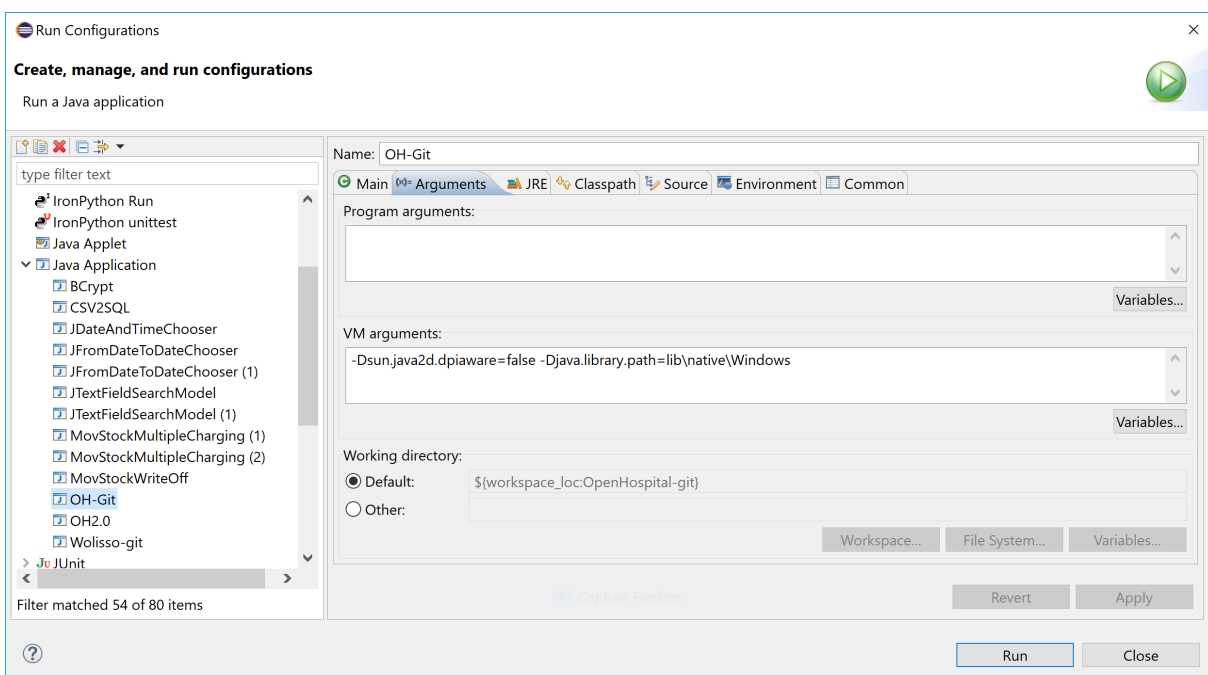
Now that the run configuration settings have been selected, OH can be launched by just clicking on the "Run" button (shown below by an arrow).
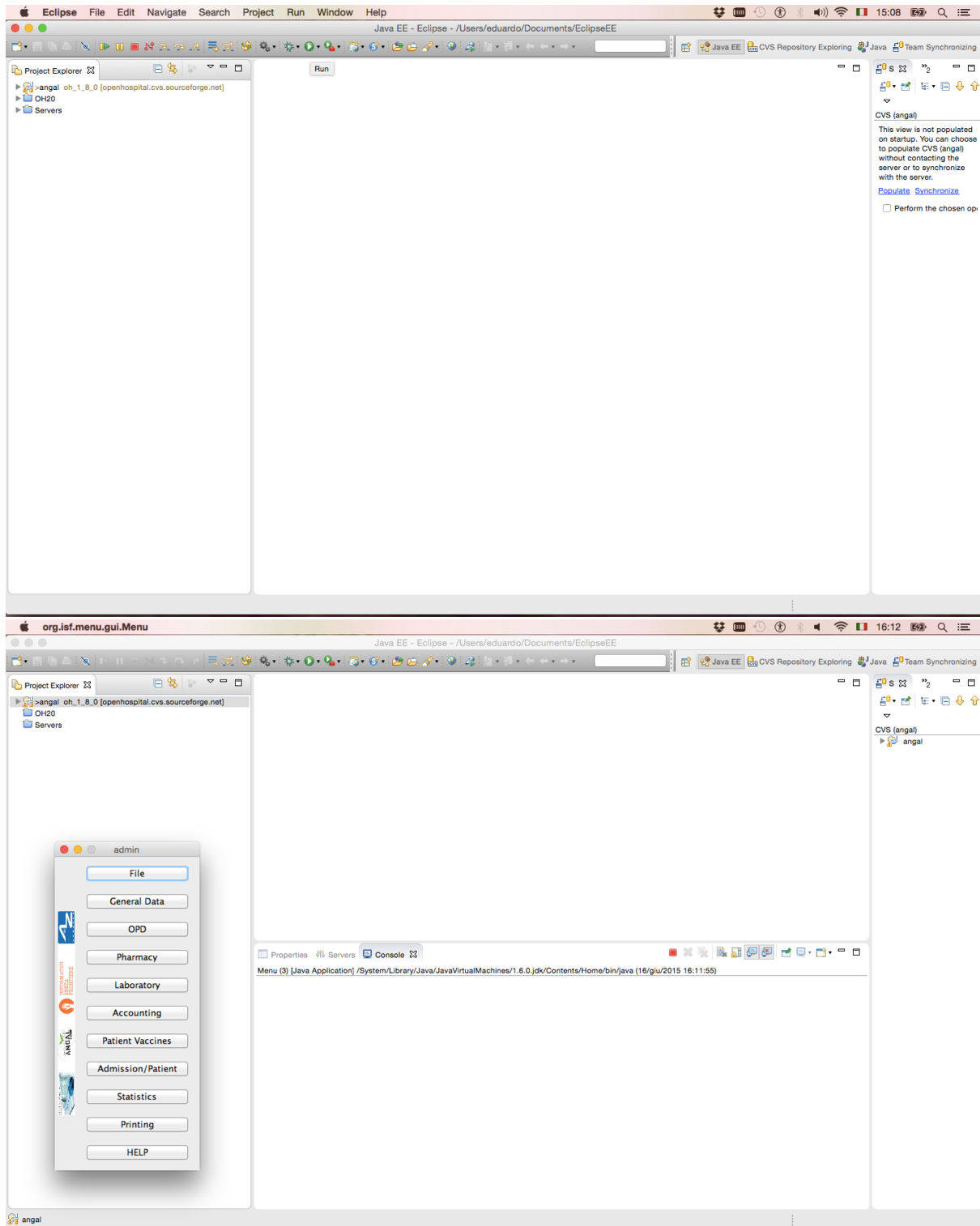


Because of the default settings, Open Hospital's first run does not show the login window. The system defaults to "single-user" mode by default. This can be verified by the "SINGLEUSER=yes" string on the third row of the settings.properties file. In production use this is unacceptable as Open Hospital must be accessible only to the administrator and logged users. Click on the triangle near the project's name, do the same on the "rsc" subfolder and then double-click on the settings.properties file.
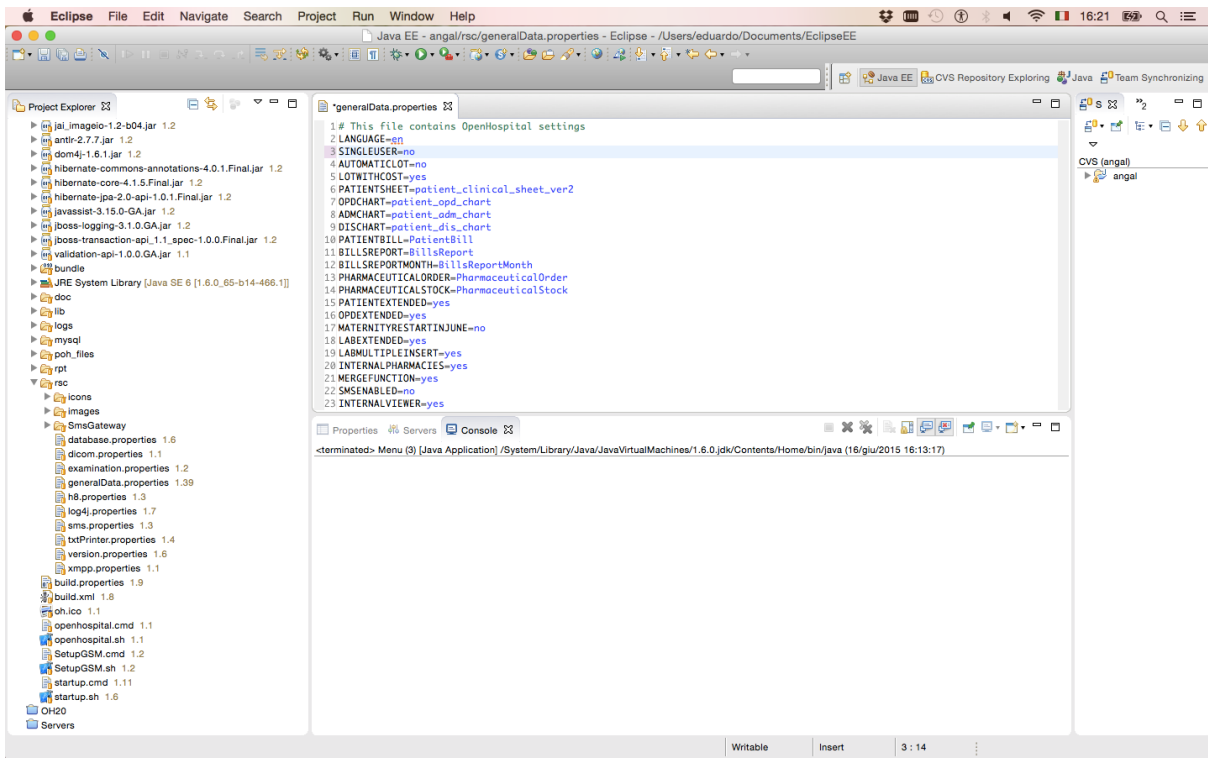
Replace "yes" with "no" on the "SINGLEUSER" row. Press CTRL+S (or Command+S on Mac) to save the change. Now rerun the application and after the splash window, a "login" window opens. Click on the drop-down menu to choose the role. To access as the administrator, select "admin" and type the password. The default is "admin". The administrator has the right to register users as guests. Guests have to choose "guest" from the drop-down menu and then type their password.
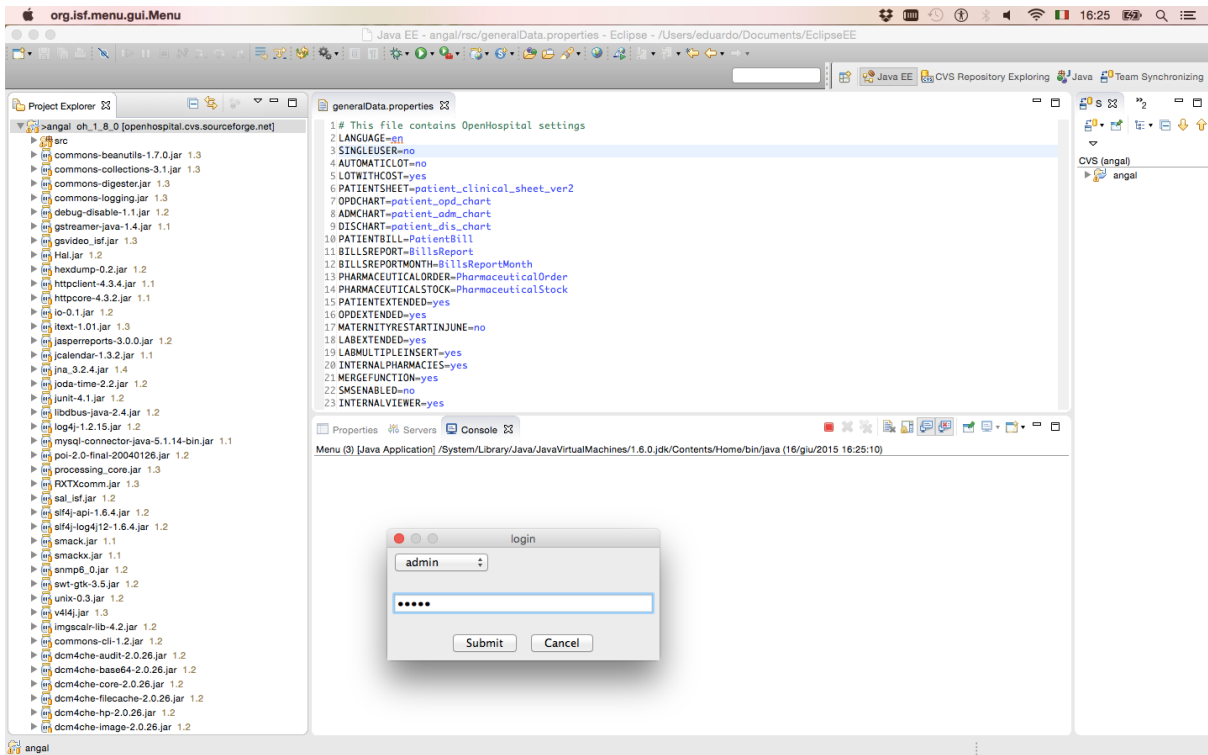


If the password is not correct, an error message is shown, asking the user to try again.

To better understand the processes underneath the software also enable the "Standard Output" in log4j.properties as explained in the chapter log4j.properties.

# 8 Update Open Hospital

## 8.1 Update Open Hospital (CLIENT mode)

> ℹ️ Following instructions only applies between "official releases" downloaded from the SourceForge website.

New releases, as well as this Manual and the User Manual, are regularly released on SourceForge/GitHub:

https://github.com/informatici/openhospital/releases

https://sourceforge.net/projects/openhospital/files/

There are two types of releases:

1. Minor Releases: only the third number changes (e.g. **1.8.**1 → **1.8.**2)

2. Major Releases: the first or second number changes (e.g. **1.**7.3 → **1.**8.0)

**Minor Releases**:

Changes are found in the CHANGELOG file.

Minor releases (if not otherwise specified) do not need changes to the database structure, so the working copy can be replaced on each client with the new one and then connect to the same DB (see database.properties and log4j.properties).

It is important to keep/preserve files that have been created or modified, especially configuration/settings and data files (see Configuration Files). The easiest way is to backup the working copy and re-apply those settings in the new installation (see the Backup & Restore chapter).

**Major Releases**:

Changes are found in the CHANGELOG file.

Major releases always have changes to the DB, and these changes are implemented via different scripts included in the sql (mysql for older version) folder (e.g.):

| OpenHospital_1_7_3.zip | OpenHospital_1_8_0.zip |
|---|---|
| OpenHospital_1_7_3/mysql/ | OpenHospital_1_8_0/mysql/ |

| OpenHospital_1_7_3.zip | OpenHospital_1_8_0.zip |
|---|---|
| ... | ... |
| step_30_help_manual.sql | step_30_help_manual.sql |
| step_31_alter_tables_innodb.sql | step_31_alter_tables_innodb.sql |
| step_32_convert_birthdate_to_date.sql | step_32_convert_birthdate_to_date.sql |
| step_33_grants_on_patientfolder.sql | step_33_grants_on_patientfolder.sql |
| | **step_34_slim_opd_table.sql** |
| | **step_35_doctors_log.sql** |
| | **step_36_patientexamination_and_rollback_step23.sql** |
| | **step_37_suppliers_table.sql** |
| | **step_38_dicom.sql** |
| | **step_39_patientfolder_submenu.sql** |
| | **step_40_sms_module.sql** |
| | **step_41_common_bundles.sql** |
| | **step_42_procedure_accounting.sql** |
| | **step_43_no_payments_fix.sql** |
| | **step_44_lot_cost.sql** |
| | **step_45_new_pharmacy_alter_table.sql** |
| | **step_46_patients_birthdate.sql** |
| | **step_47_ipd_out_disease.sql** |
| | **step_48_male_female_wards.sql** |
| | **step_49_encrypt_passwords.sql** |
| | **step_50_medical_ward_permissions.sql** |
| | **step_51_rectify_medical_ward.sql** |
| | **step_52_therapy_model_changes.sql** |
| | **step_53_sms_module_extension.sql** |
| | **step_54_enable_sms_on_visits.sql** |

Thus the update process is as follows:

1. Close the program if it is still running

2. Backup the current DB for safety (see Backup & Restore)

3. Open a terminal in the folder sql/ (e.g.):

```
C:\WINDOWS\system32> cd C:\Users\OH\OpenHospital.1.11.0\sql

C:\Users\OH\OpenHospital.1.11.0\sql>
```

4. Run the following commands (e.g.):

```
C:\Users\OH\OpenHospital.1.11.0\sql> mysql —u root -p

Enter password: ****
```

5. Use the 'root' password chosen during the installation process. The terminal should reply with the MySQL client command line prompt:

```
mysql>
```

6. Then execute all the SQL scripts that have been added from the previous version; for example:

```
mysql> source step_34_slim_opd_table.sql

mysql> source step_35_doctors_log.sql

mysql> source step_36_patientexamination_and_rollback_step23.sql

mysql> source step_37_suppliers_table.sql

mysql> source step_38_dicom.sql

mysql> source step_39_patientfolder_submenu.sql

...
```

7. Take notes if any error messages occur and report them to:
   https://sourceforge.net/p/openhospital/support-requests/

8. If everything is successful, replace the working copy on each client with the new one and connect it to the same DB (see database.properties and <log4j-properties,log4j.properties>>). Remember to keep files that have been changed, especially any changes to the configuration settings (see Configuration Files). The most common way is to backup the working copy and re-apply those settings in the new one (see Backup & Restore).

9.  Start the program and check the log files (data/logs) to ensure that the new version is running and has no error messages.

## 8.2 Update Open Hospital - PORTABLE mode

Open Hospital, used in PORTABLE mode, is not meant to be used in a production environment. To migrate a portable installation to a full client/server configuration, keeping the existing data, the following steps must be performed:

1.  Start Open Hospital in PORTABLE mode;

2.  While OH is running:

    a.  Backup of data (Database)

3.  Close Open Hospital;

4.  Install the database server as outlined here Database Server and Open Hospital DB;

5.  Restore of data (Database) onto the the new database server;

6.  Change settings to point to the new database (see database.properties and log4j.properties).

For questions about updating, migrating, or maintaining an Open Hospital installation, please contact us at: info@informaticisenzafrontiere.org and specify:

- A brief history of the organization;

- The country where the installation is and the language used;

- Technical details of the OH installation;

- The current Open Hospital version;

- Complete the Open Hospital Assessment online-form form

# 9 License