



Administrator's Guide

Open Hospital

Table of Contents

1	<i>Abstract</i>	4
2	<i>Introduction</i>	5
3	<i>Installation</i>	6
3.1	OpenHospital	6
3.1.1	Java Runtime Environment	6
3.1.2	MySQL Server and OpenHospital DB	7
3.1.3	Configure OpenHospital	8
3.1.4	Run OpenHospital	9
3.2	Portable OpenHospital	10
4	<i>Folders</i>	11
4.1	Foldertree in OpenHospital	12
4.2	Foldertree in Portable OpenHospital for Linux	12
4.3	Foldertree in Portable OpenHospital for Windows	13
5	<i>Configuration Files</i>	14
5.1	generalData.properties	15
5.1.1	LANGUAGE	16
5.1.2	SINGLEUSER	17
5.1.3	AUTOMATICLOT	18
5.1.4	PATIENTSHEET	19
5.1.5	PATIENTBILL	20
5.1.6	BILLSREPORT	21
5.1.7	BILLSREPORTMONTH	22
5.1.8	PHARMACEUTICALORDER	23
5.1.9	PHARMACEUTICALSTOCK	24
5.1.10	PATIENTEXTENDED	25
5.1.11	OPDEXTENDED	26
5.1.12	MATERNITYRESTARTINJUNE	27
5.1.13	LABEXTENDED	28
5.1.14	LABMULTIPLEINSERT	29
5.1.15	INTERNALPHARMACIES	30
5.1.16	MERGEFUNCTION	31
5.1.17	INTERNALVIEWER	32
5.1.18	SMSENABLED (not yet implemented)	33
5.1.19	MAINMENUALWAYSONTOP	34
5.1.20	RECEIPTPRINTER	35
5.1.21	VIDEOMODULEENABLED	36
5.1.22	PATIENTVACCINEEXTENDED	37
5.1.23	ENHANCEDSEARCH	38
5.1.24	XMPPMODULEENABLED	39
5.1.25	DEBUG	40
5.2	database.properties	41
5.3	log4j.properties	42
5.4	txtPrinter.properties	43
5.5	xmpp.properties	44
5.6	Bundles	45
5.6.1	New Translations	45
6	<i>Reports</i>	46
6.1	Compatibility: iReport® 3.0.0	46

1 Abstract

This application is the first of a set of software products that ISF¹ has developed to support the management and the activities of the St. Luke Hospital in Angal (Uganda). After that mission, the St. Luke Hospital has become only the first one of a long list of hospitals that found this software useful.

All the work was realized as an open-source² project using only open-source development software.

This application software consists of the following parts:

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

This document describes how to install the application and how to configure it in your environment, enabling and disabling functionalities as well as manage users.

The reader will not find in this manual the information he needs for the use of the application software; to have such information you should consult our *User's Guide*, supplied with the application software.

¹ Informatici Senza Frontiere = Computer scientists without frontiers

² Open-source = software for which the original **source code** is made available to anyone free of charge.

2 Introduction

The aim of this program is to manage, in the simplest manner, the hospital administrative operations like registering patients and laboratory analysis, and to produce statistics for the health ministry. In the following chapters the users will find all the information they need in order to use properly the program.

3 Installation

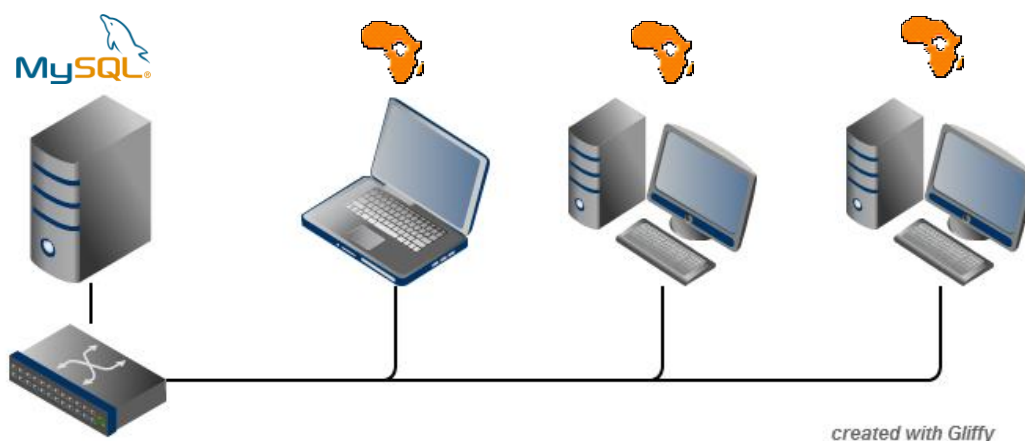
3.1 *OpenHospital*

OpenHospital is an stand-alone Java software that not require installation. The .zip file can be copied and moved everywhere in your system, whether it is Windows or Linux.

It acts like a client application. The server part is carried out by the DBMS (Database Management System) that is, from its first version, the free and open-source MySQL.

The real part of the installation it concerns the JRE installation, the MySQL Server and the creation of the DB, which is done through the execution of a script in the folder mysql/db/ within the .zip file folders.

The classic installation schema in a regular wired LAN network is shown in the follow figure:



The Administrator/Installer tasks are hereby listed:

- Install the Java Runtime Environment 1.6 or higher
- Install the MySQL server on a computer that will act as the OH-Server
- Copy the OpenHospital application on every client (it may be copied on the server too)
- Configure each OpenHospital copy to point the OH-Server

For Java Runtime Environment installation refer to next chapter [3.1.1 Java Runtime Environment](#) and to online documentation.

For MySQL installation and DB creation refer to chapter [3.1.1 MySQL Server and OH DB](#) and to online documentation.

For OpenHospital installation on every client, as told, is enough to copy and extract the zip file.

For OpenHospital configuration refer to chapter [5.2 database.properties](#)

3.1.1 Java Runtime Environment

Depending on the operating system you have, there are different ways to install a Java Virtual Machine and the Java Runtime Environment.

For instance on a Microsoft Windows® system you have to download and execute a setup file, while in Linux based systems you have to type only one command in a terminal.

OpenHospital require a **JRE 1.6 or higher** to work properly. Anyway, at the time of reading this document, many changes could have been made at the Java platform-depending installation, so we invite you to refer to online documentation.

3.1.2 MySQL Server and OpenHospital DB

Depending on the operating system you have, there are several software to download to install and manage a MySQL server. In any case it will be a normal installation of the software, with some aspects to pay attention.

For instance on a Microsoft Windows® system you will have MySQL Workbench® as a client, while in Linux based systems you have MySQL Administrator and MySQL Query Browser to perform the same tasks.

Anyway the best approach for installation is to use the command-line mysql client which come up with essentially same commands and syntax for every platform.

3.1.2.1 MySQL Server installation

For the download and installation of a MySQL Server on your server machine we invite you to refer to latest online documentation at the moment of reading this document.

Anyway, the main aspects the Administrator have to pay attention are:

- **on a Linux machine:** during the installation it will asked for a “root” password; very important to choose it carefully and to keep it safely in an installation document.
- **on a Windows machine:** during the installation it will asked for several configuration options; the best configuration will be (it could differs a bit):
 - **Custom Installation**
 - Install all components except Developer ones
 - Select **Detailed Configuration**
 - Select **Server Machine**
 - Select **Multifunctional Database**
 - Set a minimum of concurrent connection as the number of clients multiplied by two
 - Select **Enable TCP/IP Networking** and write down the **Port Number** in an installation document
 - Select UTF8 as default charset for **Best Support For Multilingualism**
 - Select **Install As Window Service**
 - Select **Include Bin Directory In Windows PATH**
 - Select **Modify Security Settings**, choose the **root password** and write it down in an installation document

Once the installation has been completed we are ready for the OpenHospital DB creation. To check if the server is active and running we can 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>
```

that means that we successfully connected to MySQL Server just installed with the root user and it is waiting for our commands.

3.1.2.2 OpenHospital DB

In order to create the OpenHospital database everything we need is the set of SQL script files contained in the OpenHospital zip file in the folder mysql/db/.

If we specified to **Include Bin Directory In Windows PATH** during the installation process (in Linux is always true) we can run the MySQL client everywhere in our system.

Open a terminal in the folder mysql/db/ and run the following commands:

```
# mysql -u root -p
Enter password: ****
```

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

```
mysql>
```

We launch the main script “create_all.sql” with the command:

```
mysql> source create_all.sql
```

The server starts to process our scripts and after some instants it will be ready again for new prompt. The DB has been created and we can check it with command:

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

The script has created the structure of the DB, populated with some default data and created the “isf” user with password “isf123”.

The isf user is the one that OpenHospital will use to connect to the DB and its password can be changed any time 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)
```

When we are through we exit the MySQL client command line with the command:

```
mysql> quit;
```

3.1.3 Configure OpenHospital

OpenHospital comes with several configuration files called “properties files”. The one we need to modify in order to let OpenHospital to find its database is the database.properties file, located in the “resource” folder rsc/.

We have to specify:

- ❖ the **IP address** of the machine where the MySQL Server has been installed and containing the OpenHospital database
- ❖ the “isf” user **password**, if changed from “isf123”
- ❖ the MySQL Server **Port Number**, if changed from “3306”

NB: These settings have to be put on every OpenHospital copy, it's to say, on every client.

More informations in chapter [5.2 database.properties](#).

3.1.4 Run OpenHospital

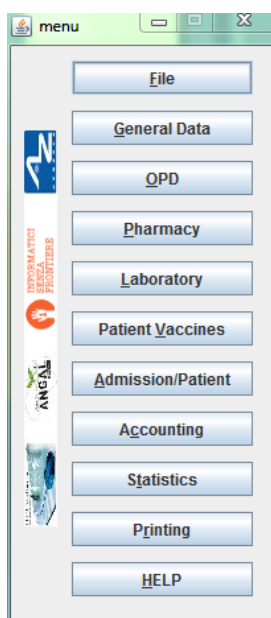
We are ready now to run the OpenHospital software by executing the proper launcher:

- on a **Linux machine**: openhospital.sh
- on a **Windows machine**: openhospital.cmd

If everything is ok you should see the OpenHospital main splash screen:



And then the main Menu:



You may create a shortcut (link) to the executable script on the Desktop, the Programs Menu or wherever you need by using the provided icon “oh.ico” that you find in the package.



3.2 *Portable OpenHospital*

OpenHospital, since version 1.2.0, has been distributed in a portable format.

Since OpenHospital is a java software is portable by definition. With the term “portable” here we refer to the possibility to have a copy self-contained of the software that includes JRE, MySQL Server and the DB, that can so be copied and run everywhere carrying together all the inserted data.

Since JRE and MySQL are not platform-independent but comes with several version depending on the platform, we have so:

- **Portable OpenHospital for Linux** platforms
- **Portable OpenHospital for Windows** platforms

Once downloaded the needed version you may start it by launching the related script inside the package:

- **oh.sh** in the Linux version
- **autorun.bat** in the Windows version

No others configurations are needed.

***NB:** if you want to use a Portable OpenHospital for Linux from a USB key you need to make sure that the file permissions in the archive remain unchanged once extracted, otherwise you will not be able to start the launcher or use the application in an appropriate manner.*

***NB:** is important to understand that portable versions are meant to have a look to the software and maybe use it in a very little facility (like a dispensary) with only one computer involved. If the facility need a network architecture the full installation is the best solution.*

4 Folders

OpenHospital has an internal folder tree hereby briefly explained:

- **bin** – contains the Java **binaries** (compiled software)
- **lib** – contains the Java **libraries** needed to run the software
- **rsc** – contains the **resources** of OpenHospital, like configuration files and other stuff
- **rsc/icons** – contains the **icons** used in OpenHospital for windows and buttons
- **rsc/images** – contains the **images** used in OpenHospital
- **rpt** – contains the JasperReports® **reports** used in OpenHospital
- **rpt/PDF** – contains the reports generated by OpenHospital in PDF format
- **bundle** – contains the **language** files
- **doc** – contains **documents** like version changelogs and manuals
- **log** – contains the log files useful for debugging
- **mysql** – contains the scripts and configurations for MySQL DB
- **scr** – contains the OpenHospital **source code**

These folders are organized differently depending on the version.

4.1 *Foldertree in OpenHospital*

In OpenHospital zip file you will find the following folder tree structure in the downloaded archive, where **<version>** is replaced with the current OpenHospital version:

- **OpenHospital_<version>/**
 - **bin**
 - **bundle**
 - **doc**
 - **lib**
 - **mysql**
 - **rpt**
 - **rsc**
 - **startup.cmd** ← **Launcher for Windows**
 - **startup.sh** ← **Launcher for Linux**
 - **oh.ico**

4.2 *Foldertree in Portable OpenHospital for Linux*

In Portable OpenHospital for Linux you will find the following folder tree structure in the downloaded archive, where **<version>** is replaced with the current OpenHospital version:

- **poh-linux-0.0.2-core-<version>/**
 - **etc/**
 - **jdk1.6.0_04/**
 - **mysql-5.0.51a-linux-i686/**
 - **oh-<version>/**
 - **bin**
 - **bundle**
 - **doc**
 - **lib**
 - **rpt**
 - **rsc**
 - **tmp/**
 - **var/**
 - **oh.sh** ← **Launcher for Linux**

4.3 *Foldertree in Portable OpenHospital for Windows*

In Portable OpenHospital for Windows you will find the following folder tree structure in the downloaded archive, where `<version>` is replaced with the current OpenHospital version:

- **poh-win32-0.0.4-core-`<version>`/**
 - **openhospital/**
 - jvm
 - modified
 - mysql
 - **oh**
 - bin
 - bundle
 - doc
 - lib
 - rpt
 - rsc
 - tmp
 - english.bat
 - french.bat
 - german.bat
 - swahili.bat
 - italian.bat
 - spanish.bat
 - oh.ico
 - **autorun.bat ← Launcher for Windows**
 - autorun.inf

5 Configuration Files

Some behaviors of OpenHospital depend by settings set in some text files called “properties files”. OpenHospital 1.7 comes with four main properties files:

- **database.properties** – for OpenHospital database connection
- **generalData.properties** – for OpenHospital configuration options
- **log4j.properties** – for the logging system
- **txtPrinter** – for the text printing system
- **xmpp.properties** – for the Xmpp Server

Furthermore there are other configurations files that are not meant to be modified by the user:

- **resolutions.xml** – for store locally the informations about webcams
- **version.properties** – that contains the current OpenHospital version

Every configuration file works only locally and is meant to pilot only a single instance of OpenHospital application.

This means that if we have many clients running OpenHospital they may run differently according to their own configuration files.

NB: every change to these files need an application restart in order to see the modifications applied

In following chapters we will see all these properties files and all their feature in details.

5.1 *generalData.properties*

OpenHospital 1.7 comes with the generalData.properties file set as follow by default:

```
# This file contains OpenHospital settins
LANGUAGE=en
SINGLEUSER=yes
AUTOMATICLOT=no
PATIENTSHEET=patient_clinical_sheet
PATIENTBILL=PatientBill
BILLSREPORT=BillsReport
BILLSREPORTMONTH=BillsReportMonth
PHARMACEUTICALORDER=PharmaceuticalOrder
PHARMACEUTICALSTOCK=PharmaceuticalStock
PATIENTEXTENDED=yes
OPDEXTENDED=yes
MATERNITYRESTARTINJUNE=no
LABEXTENDED=yes
LABMULTIPLEINSERT=yes
INTERNALPHARMACIES=yes
MERGEFUNCTION=yes
SMSENABLED=no
INTERNALVIEWER=yes
MAINMENUALWAYSONTOP=no
RECEIPTPRINTER=no
VIDEOMODULEENABLED=no
PATIENTVACCINEEXTENDED=yes
ENHANCEDSEARCH=no
XMPPMODULEENABLED=no
DEBUG=no
```

Every line is composed by key (capitalized) and a value (in blue) as follow:

key = value

Furthermore values can be:

- **Boolean:** yes | no or true | false
- **String:** usually a filename or a countrycode

In following chapters we will see the meaning of each of them.

5.1.1 LANGUAGE

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

key	default value	allowed values
LANGUAGE	en	en, it, fr, es, de, sw

OpenHospital 1.7 comes with six translations, identified by the international country code:

- en – English
- it – Italian
- fr – French
- es – Spanish
- de – German
- sw – Swahili

You can change the language used in the application simply by editing ***generalData.properties*** and setting this key. If a not allowed value is set, the local computer language will be applied.

NB: *An application restart is required in order to see the modification applied.*

5.1.2 SINGLEUSER

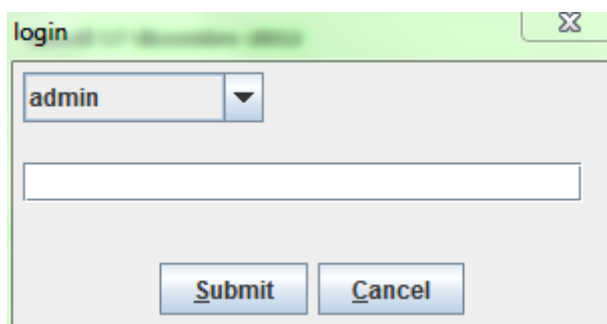
The following table shows the default value and the allowed ones:

key	default value	allowed values
SINGLEUSER	yes	yes , no

OpenHospital, since its first version, allow a multi-user usage. It 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.

OpenHospital has a predefined user called “admin” that have all rights to access all functionality in the application. When SINGLEUSER is set to **YES** (default) the program start with this user and all the functionalities are available.

If *SINGLEUSER* is set to **NO** the following login window will appear when start the program:



The default password for “admin” user is “admin”. By typing his password in the white field and by pressing **Submit**, you will see the main **Menu** with all functionalities enabled.

Once you’re logged in as “admin” user you can define users and group (see [Users and Groups \(Users\)](#)), or just stay with only that as a simple form of data protection.

NB: despite this setting works only locally the users and groups defined will work for every client connected to the server; anyway is possible to avoid login simply by acting on this parameter.

Disclaimer: data security must never be left solely to application but it must be ensured by a proper network architecture and by a rigid configuration of the clients.

5.1.3 AUTOMATICLOT

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

key	default value	allowed values
AUTOMATICLOT	no	yes, no

OpenHospital, since its first version, allow an automatic management of lots in the main pharmacy. It means that is possible to work in the pharmacy without taking care about lot definitions in loading the store (but not in discharging) and just specify preparation and expiring date for each charging movement (see **6.2.2.2 Insert stock movement** in the *User's Guide*).

By default AUTOMATICLOT is set to **NO**. Anyway is possible to set it anytime to **YES** to enable it.

NB: *An application restart is required in order to see the modification applied.*

5.1.4 PATIENTSHEET

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

key	default value	allowed values
PATIENTSHEET	patient_clinical_sheet	any kind of .jasper file name

OpenHospital, since its first version, allows to 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. Anyway is possible to use a different report by installing it in the report folder (see [Reports](#)) and by changing this parameter.

NB: *An application restart is required in order to see the modification applied.*

5.1.5 PATIENTBILL

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

key	default value	allowed values
PATIENTBILL	PatientBill	any kind of .jasper file name

OpenHospital, since version 1.6.0, allows to manage patient bills and to 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. Anyway is possible to use a different report by installing it in the report folder (see [Reports](#)) and by changing this parameter.

NB: *An application restart is required in order to see the modification applied.*

5.1.6 BILLSREPORT

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

key	default value	allowed values
BILLSREPORT	BillsReport	any kind of .jasper file name

OpenHospital, since version 1.6.0, allows to manage patient bills and to 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. Anyway is possible to use a different report by installing it in the report folder (see [Reports](#)) and by changing this parameter.

NB: *An application restart is required in order to see the modification applied.*

5.1.7 BILLSREPORTMONTH

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

key	default value	allowed values
BILLSREPORTMONTH	BillsReportMonth	any kind of .jasper file name

OpenHospital, since version 1.6.0, allows to manage patient bills and to 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. Anyway is possible to use a different report by installing it in the report folder (see [Reports](#)) and by changing this parameter.

NB: *An application restart is required in order to see the modification applied.*

5.1.8 PHARMACEUTICALORDER

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

key	default value	allowed values
PHARMACEUTICALORDER	PharmaceuticalOrder	any kind of .jasper file name

OpenHospital, since version 1.7.0, allows to produce a report for the Pharmacy Stock critical levels, that help to know which pharmaceuticals are going to end 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. Anyway is possible to use a different report by installing it in the report folder (see [Reports](#)) and by changing this parameter.

NB: *An application restart is required in order to see the modification applied.*

5.1.9 PHARMACEUTICALSTOCK

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

key	default value	allowed values
PHARMACEUTICALSTOCK	PharmaceuticalStock	any kind of .jasper file name

OpenHospital, since version 1.7.0, allows to produce a report for the Pharmacy Stock Status, that help to know which pharmaceuticals are currently stored (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. Anyway is possible to use a different report by installing it in the report folder (see [Reports](#)) and by changing this parameter.

NB: *An application restart is required in order to see the modification applied.*

5.1.10PATIENTEXTENDED

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

key	default value	allowed values
PATIENTEXTENDED	yes	yes , no

OpenHospital, since version 1.3.1, allows register patient with a big set of information. This set of information are meant to “extend” the ones collected since version 1.2.1 (see **10.4 Insert a new Patient Extended** in the *User’s Guide*).

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

NB: *An application restart is required in order to see the modification applied.*

5.1.11 OPDEXTENDED

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

key	default value	allowed values
OPDEXTENDED	yes	yes, no

OpenHospital, since version 1.3.1, allows to link every OPD visit to a registered patient in order to have a comprehensive clinical history. For each visit the user will be asked to select a registered patient or to register a new one, so from that moment the visit will be attached to his/her history (see **5.3 OPD Extended** and **10.9 Clinical Sheet** in the *User's Guide*).

By default OPDEXTENDED is set to **yes**. Anyway is possible to set it to **no** in order 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.

NB: *An application restart is required in order to see the modification applied.*

5.1.12 MATERNITYRESTARTINJUNE

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

key	default value	allowed values
MATERNITYRESTARTINJUNE	no	yes, no

OpenHospital, since version 1.3.1, allows to change the way the admissions in Maternity ward are counted within the year; in OpenHospital 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. With this option we obtain exactly this behavior.

By default MATERNITYRESTARTINJUNE is set to `no`.

NB: *An application restart is required in order to see the modification applied.*

5.1.13 LABEXTENDED

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

key	default value	allowed values
LABEXTENDED	yes	yes , no

OpenHospital, since version 1.4.1, allows to force every laboratory exam to be linked to a registered patient, in order 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 his/her 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 in order to reduce the amount of data to be collected (only name, age and sex, no patient registration) and so reduce the workload for the staff involved in data entry.

NB: *An application restart is required in order to see the modification applied.*

5.1.14 LABMULTIPLEINSERT

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

key	default value	allowed values
LABMULTIPLEINSERT	yes	yes, no

OpenHospital, since version 1.4.1, allows to insert multiple laboratory tests per patient at a time in order to avoid the repetitive operation of selecting a patient or writing his/her name; the LABEXTENDED option must be set to **yes** too (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.

NB: *An application restart is required in order to see the modification applied.*

5.1.15 INTERNALPHARMACIES

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

key	default value	allowed values
INTERNALPHARMACIES	yes	yes, no

OpenHospital, since version 1.4.1, allows to register all giving to patients within a ward. Activating this option the Pharmaceutical Stock Ward functionality will be available in the application (see **6.3 Pharmaceuticals Stock Ward** in the *User's Guide*)

By default INTERNALPHARMACIES is set to [yes](#).

NB: *An application restart is required in order to see the modification applied.*

5.1.16 MERGEFUNCTION

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

key	default value	allowed values
MERGEFUNCTION	no	yes , no

OpenHospital, since version 1.4.1, allows to merge two patients history in only one. This is particularly useful in case of double registration that have collected different informations during the time. Activating this option the Merge functionality will be available in the Admission/Patient module (see **10.11 Merge function** in the *User's Guide*).

By default MERGEFUNCTION is set to no.

NB: *An application restart is required in order to see the modification applied.*

5.1.17 INTERNALVIEWER

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

key	default value	allowed values
INTERNALVIEWER	yes	yes, any executable filename in the path or with absolute path

OpenHospital, since version 1.4.1, allows to use a different PDF reader for generated reports. If this parameter is set to [yes](#), the Jasper PDF Viewer will be used (see **11.2 Report Launcher** in the *User's Guide*). If we want to use an external PDF viewer we may put its executable filename but we must be sure that it is available in the desktop environment.

Example in Windows:

```
INTERNALVIEWER=AcroRd32.exe
```

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

NB: Please note the double file & folder separator “\\”.

Example in Linux:

```
INTERNALVIEWER=evince
```

By default INTERNALVIEWER is set to [yes](#).

NB: An application restart is required in order to see the modification applied.

5.1.18 SMSENABLED (not yet implemented)

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

key	default value	allowed values
SMSENABLED	no	yes, no

OpenHospital, since version 1.5, allows to set this parameter that will be used to set some information ready for SMS notification to the patients. Enabling or disabling this option will only affect the possibility to set or not a therapy as notifiable (see **10.10.7 Notify & SMS** in the *User's Guide*)

By default SMSENABLED is set to no.

5.1.19 MAINMENUALWAYSONTOP

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

key	default value	allowed values
MAINMENUALWAYSONTOP	no	yes, no

OpenHospital, since version 1.5, allows to keep the main menu always on top so it cannot be overlapped by other windows.

By default MAINMENUALWAYSONTOP is set to no.

NB: An application restart is required in order to see the modification applied.

5.1.20 RECEIPTPRINTER

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

key	default value	allowed values
RECEIPTPRINTER	no	yes , no

OpenHospital, since version 1.5, allows to drive text or ZPL printers with aim of printing simple receipts in Accounting module (see **8.2.1.10 Print receipt** function in *User's Guide*). Once this option has been activated, some parameters have to be set in **txtPrinter property file** in order to adjust the output with the printer connected to the system (see [txtPrinter.properties](#) in this document).

By default RECEIPTPRINTER is set to no.

NB: *The text printer must be set as default printer.*

NB: *An application restart is required in order to see the modification applied.*

5.1.21 VIDEOMODULEENABLED

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

key	default value	allowed values
VIDEOMODULEENABLED	no	yes, no

OpenHospital, since version 1.5, allows to drive any webcam with the aim of shooting patients passports in Admission/Patient module (see **10.4.3 Patient Photo** function in *User's Guide*). Once this option has been activated the related button **New Photo** in **New Patient Extended** window will be available to start and use the webcam.

By default VIDEOMODULEENABLED is set to no.

NB: *An application restart is required in order to see the modification applied.*

5.1.22PATIENTVACCINEEXTENDED

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

key	default value	allowed values
PATIENTVACCINEEXTENDED	yes	yes, no

OpenHospital, since version 1.6, allows to register vaccines given to patients (see **9 Vaccines** in the User's Guide). This option allow to toggle the patient's fullname in the *Patient vaccines browsing* window.

By default PATIENTVACCINEEXTENDED is set to *yes*.

NB: An application restart is required in order to see the modification applied.

5.1.23 ENHANCEDSEARCH

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

key	default value	allowed values
ENHANCEDSEARCH	no	yes , no

OpenHospital, since version 1.6, allows to optimize memory usage when the number of registered patients becomes huge (see **10.2.2 Search patient Enhanced** in the User's Guide). With the enhanced search only the patient matching a search criteria will be loaded in the memory, otherwise all patients registered in the system will be loaded in the *Patients browser* window.

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

NB: *An application restart is required in order to see the modification applied.*

5.1.24 XMPPMODULEENABLED

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

key	default value	allowed values
XMPPMODULEENABLED	no	yes , no

OpenHospital, since version 1.7, embeds a new technology that allow user's logged in the application to chat and share informations 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, OpenHospital will look for an XMPP Server at startup time. The **xmpp.properties** will be used to point 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** it will be just ignored.

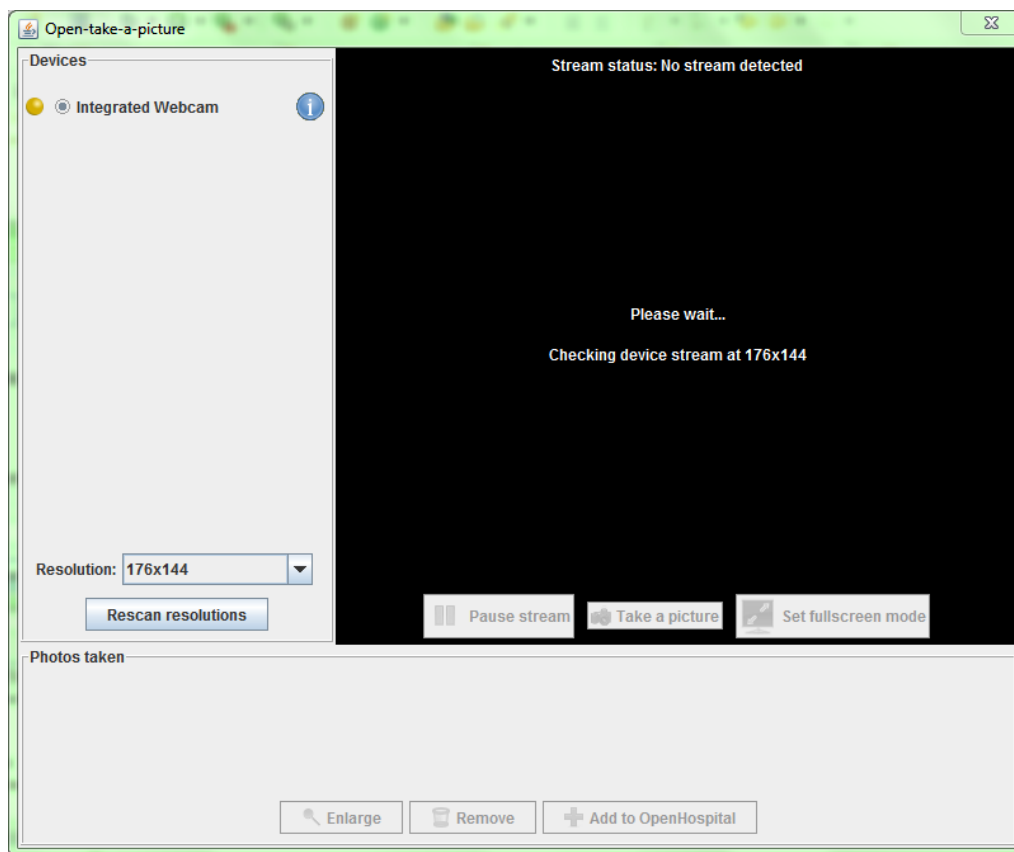
NB: *An application restart is required in order to see the modification applied.*

5.1.25DEBUG

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

key	default value	allowed values
DEBUG	no	yes , no

OpenHospital, since version 1.7, allows to run itself in debug mode that could be useful to understand what is happening behind the user graphical interface. At the moment this option will affect only the video module, that means that in place of the normal **New Patient Extended** window (see **10.4.3 Patient Photo** function in *User's Guide*), it will open the new special window showed below:



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

Once produced a photo, OpenHospital will remember the last setting used in the future.

By default DEBUG is set to no.

NB: An application restart is required in order to see the modification applied.

5.2 *database.properties*

OpenHospital 1.7 comes with the database.properties file set as follow by default:

```
# This file contains MySQL Server informations
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
```

First two parameters are not meant to be modified by the User, neither by the Administrator:

- **jdbc.drivers**: the java library used to connect to a MySQL Server
- **jdbc.url**: the address to point a MySQL Server

Then we have five parameters which may need to be set according to your network settings instead:

- **jdbc.server**: the IP address of the server where the 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 private IP address (LAN)
217.147.110.117 – an example of public IP address (WAN)
- **jdbc.db**: the name of the OpenHospital database
- **jdbc.username**: the MySQL user allowed to connect and operate on the database
- **jdbc.password**: the user password
- **jdbc.port**: the TCP port where the MySQL Server is listening; by default this port is 3306 but can be changed if needed; anyway must reflect the MySQL Server settings

The full address for the db will then result as:

`jdbc:mysql://localhost/oh`

NB: An application restart is required in order to see the modification applied.

5.3 log4j.properties

OpenHospital 1.7 comes with the log4j.properties file set as follow by default:

```
# logging di default su file, livello FINEST (debug++)
log4j.rootCategory=FINEST
#log4j.rootLogger=R

# 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}')
```

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 size of the log files and the tables are significantly reduced:

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

On the contrary, in case of malfunctions of the application, the log level can be increased to "TRACE", in this way will be shown more details on the operation of the application:

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

You can even disable the log file or the log DB, by deleting the respective **appender** if not needed:

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

Or decide to log only in the standard output (the console window) with:

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

NB: An application restart is required in order to see the modification applied.

5.4 *txtPrinter.properties*

OpenHospital 1.7 comes with the txtPrinter.properties file set as follow by default:

```
# This file contains text printing informations
PRINT_AS_PAID=yes
PRINT_WITHOUT_ASK=no
PAGE_HEIGHT=50
PAGE_WIDTH=47
ZPL=yes
ZPL_FONT_TYPE=0
ZPL_ROW_HEIGHT=25
```

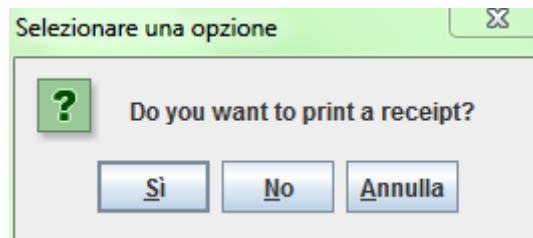
This file will be used only if the RECEIPTPRINTER flag is enabled (see [5.1.20 RECEIPTPRINTER](#) in this document)

This file has seven parameters, hereby explained:

- **PRINT_AS_PAID:** it will affect the behavior when a Bill is set to PAID in the Accounting module (see **8.2.1.10 Print receipt** function in *User's Guide*).

If it is set to **yes** OpenHospital will try to print a receipt for the Bill in text format in a page of size **PAGE_HEIGHT** and **PAGE_WIDTH**.

A confirmation window will appear:



Otherwise you must do it after, by press **Edit Bill** button on the already closed (“C”) bill

- **PRINT_WITHOUT_ASK:** if set to **yes** OpenHospital will try to print the receipt without confirmation window
- **ZPL:** if set to **yes** we are specifying that 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 standard character; you may change this value with “A”, “B”, “C”, etc... according with the device datasheet, in order to obtain a better look
- **ZPL_FONT_SIZE:** an integer value to adjust the character size in order to obtain a better look

The best way to set these parameters according with your device is to set them one by one and check the different results until you reach the best fit.

NB: An application restart is required in order to see the modification applied.

5.5 *xmpp.properties*

OpenHospital 1.7 comes with the xmpp.properties file set as follow by default:

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

This file will be used only if the XMPPMODULEENABLED flag is enabled (see [5.1.24 XMPPMODULEENABLED](#) in this document)

In order to use the Communication module (see **13 Communication** in the *User's Guide*) you must install an XMPP Server separately on the OH Server, let's say side by side with MySQL 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 we may set the two parameters in the xmpp.properties file as follow:

- **DOMAIN** – the IP address of the XMPP server (it could be the same of MySQL Server)
- **PORT** – 5222 or another if set differently

NB: *An application restart is required in order to see the modification applied.*

5.6 Bundles

Bundles are the language (or translation) files provided with OpenHospital. As told in [5.1.1 LANGUAGE](#) chapter, OpenHospital 1.7 comes with different available languages in folder `bundles/` within the package file:

language_XY.properties

where XY is an international country code.

Since they are property files, every language file is made by hundreds of key = value couple too.

Special characters are coded in the UNICODE coding system with 4 bytes, like:

U+00E1 = \u00E1 = á

U+00E8 = \u00E8 = è

U+00F6 = \u00F6 = ö

...

5.6.1 New Translations

In order to create a new translation or to fix some labels in the OpenHospital is enough to create a copy of the English one, rename it with the new country code in place of the “XY” in the filename, and start translating with a simple text editor.

For instance, if you want to create an Arab translation, you may create a new bundle from the English one:

language_en.properties → language_ar.properties

Then you can edit the Arab file and set the LANGUAGE parameter to **ar** to start testing your translation.

6 Reports

Reports in OpenHospital are produced with JasperReports® technology that allow 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.

OpenHospital 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 OpenHospital; this file is produced on the jrxml base

The Jasper editor to create and modify Jasper reports is iReport®, a free and open-source software.

With iReport® is possible to edit OpenHospital reports and re-compile them in order to be used in OpenHospital application.

6.1 *Compatibility: iReport® 3.0.0*

In order to have compliant and fully-working reports for OpenHospital we suggest to use iReport® version 3.0.0.

6.2 *Links*

JasperReports® - <http://www.jaspersoft.com/>

iReport® - <http://community.jaspersoft.com/project/ireport-designer>

iReport® 3.0.0 - [http://sourceforge.net/projects/ireport/files/iReport%20\(classic\)/](http://sourceforge.net/projects/ireport/files/iReport%20(classic)/)