e-TB Manager Developer Handbook

Febryary 2017

By Ricardo Memoria

# 

# 

Summary

[**Introduction**](#_ih9ua9hdj21d) **2**

[What is e-TB Manager?](#_8lump0t1db0h) 2

[Target Audience](#_jawk4xo8yyys) 2

[Required Skills](#_y5kwjieo7av8) 2

[**Getting Started**](#_dbq3w86ly19v) **3**

[Check software requirements](#_bna1wnqm5675) 3

[Download the source code](#_65x6jnm7zat0) 4

[Setup MySQL](#_bhhpibd93e9q) 5

[**Building from the source code**](#_7mqx01kvdugr) **7**

[Skipping tests](#_nhqfz4q3yqrd) 7

[Building client code](#_wg9vro2y94t9) 7

[**Configuring and running**](#_sqyhl1byn19j) **9**

[Configuration file](#_eqzl4tiy5xx6) 9

[Database connection](#_ek3frkh5p8ca) 9

[Web parameters](#_hzwhnh3gp87j) 9

[E-mail delivery](#_f0t3w1d7ar) 10

[Development parameter](#_6alt4w9monrh) 11

[Running e-TB Manager](#_olgtv6q3wb3v) 11

# 

# 

# Introduction

This document explains in details how to setup a development environment and generate a new distribution package for e-TB Manager version 3, from its source code.

## What is e-TB Manager?

e-TB Manager is a web-based tool for managing all the information needed by national TB control programs. It integrates data across all aspects of TB control, including information on suspects, patients, medicines, laboratory testing, diagnosis, treatment, and outcome.

## Target Audience

The main audience of this document is software developers interested in modifying and maintain e-TB Manager 3 source code. This document explains in details how to setup the development environment and generate a new distribution package from the source code.

## Required Skills

Before change the source code, it is important that the developer feels comfortable with the following languages and technologies:

* Java 1.8+, for server side programming. It is also recommended experience with WEB development, Spring framework, REST API using Spring MVC framework, Spring Boot, JPA (Hibernate), Dozer, JUnit and Maven;
* JavaScript ES6 + ReactJS, for client side programming;
* MySQL 5.6+ and HSQLDB, if you intend to make changes in the database structure;
* HTML5/CSS3, for UI, page flow and design and animation;

Please check the sections below for a detailed list of frameworks.

# Getting Started

Before you perform any change in the source code, you will need to follow the steps below:

* Check software requirements;
* Download the source code;
* Setup MySQL;

## Check software requirements

This section will list the softwares you will need, depending on the tasks that you intend to perform. Installation instructions for them are widely available in the Internet and vary depending on the operating system you use. All softwares used in the development side are freely distributed as open-source, so license will not be an issue.

The requirements will depending on the kind of task that you intend to do, so to avoid unnecessary software installation, the list is split by the following groups:

**Package building** - If you just want to generate an e-TB Manager installation package from the source code, this is the minimum list of programs you will need to have installed in your computer. It is also the minimum requirement for any other type of environment below that you will need before working with e-TB Manager:

* Java JDK 1.8+
* Maven 3+

**Source code repository** - If you are part of the development team in MSH, most probably you will be given access to the repository where the source code is located. Since we use git, you will be required to have it installed in your computer as well:

* Git

**Database** - In e-TB Manager version 3, you have the choice to use either MySQL or HSQLDB as the data storage. HSQLDB is an embedded server, so no installation is required, while if you want to use MySQL, you will have to install it.

* MySQL 5.7+

**Client side development** - If you want to change client side code, installing the apps below will make your life easier:

* Node 5.0+
* Gulp 3.8+
* Mocha

**IDE** - If you want to change the source code, most probably you want to install an IDE to give you better source code editor. e-TB Manager doesn't require any specific IDE for its development, and you can even use a simple text editor, but internally we use IntelliJ in our daily work with the code.

## Download the source code

This section is for those that work with the development team in MSH, and are sharing the same source code repository. If this is not your case, and you already have a copy of the source code, just skip this section.

e-TB Manager v3 uses Git as its source code repository. Today the remote repository is located in BitBucket at the URL:

|  |
| --- |
| https://rmemoria@bitbucket.org/etbmanager/etbm3.git |

This is a private repository and required authorization from MSH in order to get access to it.

From the command line, to download the code, go to a folder of your preference and issue the following git command:

|  |
| --- |
| git clone https://rmemoria@bitbucket.org/etbmanager/etbm3.git |

You will need a user name and password for that. Provide your credentials and wait while git downloads the source code to your local computer's folder.

Inside the repository, there are two main branches: master and development. The master contains the latest stable version. The development branch contains the latest code under development and should not be used in a production environment. By default, when you download the source code, the branch in use is master. The master branch in the remote repository is read-only for most of the users, so you probably want to change the code in the development branch. Issue the command below to switch to the development branch:

|  |
| --- |
| git checkout development |

Suppose you made changes in the source code and want to include them in the git repository. You want to do the following sequence of git commands:

1. Add all changed files to be committed

|  |
| --- |
| git add . |

1. Commit the changes providing a good description

|  |
| --- |
| git commit -m "description of the changes" |

1. Upload the changes to the remote server

|  |
| --- |
| git push origin development |

These three commands guarantee that all your changes are saved to your local and remote repository.

## Setup MySQL

If you want to use HSQLDB instead of MySQL, please skip this section. Although HSQLDB is a fully embedded database server and doesn't require any installation, MySQL provides better tools for monitoring, query execution and analysis, but in terms of system execution, both of them provide the same features.

You probably have already installed MySQL in your computer (if no, search in the Internet for several possible ways of installing MySQL).

During development, by default, e-TB Manager will try to connect to MySQL using the following parameters:

* Database: etbm3
* Username: root
* Password: admin

These parameters can be easily changed in the etbmanager.properties file, that must be available in the working directory.

The following query can be used to create a new database in MySQL:

|  |
| --- |
| create database etbm3 character set utf8 |

The UTF8 is the recommended character set, since it supports Latin, Cyrillic, Chinese, Japanese, Arab and most of the alphabets in the world.

If you want to execute test scripts, the default database name is etbm3\_test. So it is a good idea to create this database as well:

|  |
| --- |
| create database etbm3\_test character set utf8 |

To configure the settings used by e-TB Manager to connect to MySQL database, include the following lines in etbmanager.properties file (located in the working directory):

|  |
| --- |
| db.url=jdbc:mysql://localhost:3306/etbm3?useUnicode=yes&characterEncoding=UTF-8 db.user=root db.password=admin |

In the db.url parameter, the complement after the database name (etbm3) is required to guarantee that the system will handle different languages, like Russian, Chinese and others.

# Building from the source code

Building from the source code is a straightforward operation - all you need is Maven (and Java, obviously) installed and available in the operating system path. Simply go to the source code root folder and issue the following Maven command:

|  |
| --- |
| mvn clean package |

This single command will perform the following tasks (in the given order):

* Clean all temporary files;
* Download and install any necessary dependencies;
* Compile both client and server side;
* Perform unit test;
* Package the compiled files;
* Perform integration tests;

If Maven execution finishes without errors, the result of the building process is the following files:

|  |
| --- |
| target/etbmanager-<version>-windows.zip  target/etbmanager-<version>-unix.zip |

Where <version> is the current version being compiled. These files are intended to be distributed to install the system in Windows and Unix based operating systems (including Linux).

## Skipping tests

Although extremely important, tests can slow down the building process, so if you are sure about the result of the tests, you can skip it including the parameter -DskipTests:

|  |
| --- |
| mvn clean package -DskipTests |

## Building client code

Although client code is also compiled during the building process described above, there are situations where you want to work with client code. The building system uses Gulp (<http://gulpjs.com/>) to perform client code testing and compiling. The following gulp tasks are supported:

|  |
| --- |
| gulp build |

Generate artifacts of the client side for production, like Java Script code, images, fonts, and other resources.

|  |
| --- |
| gulp eslint |

Check if there is any error or warning about code style and linting (using ESLint).

|  |
| --- |
| gulp run |

Execute a local web server used for development.

# Configuring and running

## Configuration file

When executed, e-TB Manager will initially try to open a file called etbmanager.properties, located in the working directory. This is a text file required by e-TB Manager, and contains configuration data, like database connection parameters, e-mail dispatch server and web parameters.

In a nutshell, these are the main configuration parameters used in a development environment:

## Database connection

Below is the list of parameters used by e-TB Manager to perform database connections:

* db.url - The jdbc url used to get a database connection;
* db.user - The user name, if required by the database;
* db.password - The user name password, if required by the database;

Example of database parameters for a MySQL database connection:

|  |
| --- |
| db.url=jdbc:mysql://localhost:3306/etbm3?useUnicode=yes&characterEncoding=UTF-8 db.user=root db.password=admin |

Example of database parameters for a HSQLDB database connection:

|  |
| --- |
| db.url=jdbc:hsqldb:file:database/etbmanager;default\_schema=true db.user=sa db.password=admin |

In this example, the database files will be located in the database folder, located in the working directory.

## Web parameters

* web.context - The URL context used by the application. The default is that e-TB Manager will serve requests in the root URL context, but once the context is specified, the URL will have the format http://<host>/<web.context>/. It is necessary to start the context with slash (Example: web.context=/etbmanager);
* web.port - The port number in use by e-TB Manager to respond for web requests. Default is 8080;

Example of web parameters:

|  |
| --- |
| web.context=/etbmanager  web.port=8088 |

## E-mail delivery

e-TB Manager constantly sends e-mail messages to inform users about important events. If you want to have this feature enabled, you must inform mail parameters to dispatch messages. The main parameters are:

* mail.host - The SMTP server name;
* mail.from - The recipient of the messages (used in the from clause of the message);
* mail.username - The username, if the SMTP server requires authentication;
* mail.password - The user password, if the SMTP server requires authentication;

These are basic mail parameters, but may vary according to the target server. For example, if you are using a GMail account, these parameters should be used:

|  |
| --- |
| mail.host = smtp.gmail.com mail.username = \*\*\*\*@gmail.com mail.password = \*\*\*\* mail.from= \*\*\*@gmail.com spring.mail.properties.mail.smtp.auth = true spring.mail.properties.mail.smtp.socketFactory.port = 465 spring.mail.properties.mail.smtp.socketFactory.class = javax.net.ssl.SSLSocketFactory spring.mail.properties.mail.smtp.socketFactory.fallback = false |

Just remember to replace the asterisks (\*) by the correct value.

A good idea is to install a local SMTP server to use during development. There are a lot of local SMTP servers available for that. One we have used is FakeSMTP (<http://nilhcem.com/FakeSMTP/>), which is a Java application that simulates a SMTP to be used only for development purposes. For a local server, these are the parameters:

|  |
| --- |
| mail.host=localhost mail.port=2525 mail.from=system@etbmanager.org |

## Development parameter

e-TB Manager has a development parameter that, when true, makes it run slightly different compared to production mode. To run in development mode, you must set the development parameter to true in etbmanager.properties file:

|  |
| --- |
| development=true |

Running on development mode brings the following side effects to the system:

* System will return the not final version of JavaScript files, but the one used under development;
* Log interesting messages for developers;
* Cache evicting in certain operations;

**IMPORTANT**: If you want to maintain the source code, always remember to set this parameter in the etbmanager.properties file.

## Running e-TB Manager

Probably you want to run e-TB Manager from an IDE, which will give you support for replacement of classes and resources without needing to restart the application. But regardless of how you want to run, the simplest way is to run it as a Java application, simply issuing the following command where the jar file was generated:

|  |
| --- |
| java -jar etbmanager-<version>.jar |

This will start e-TB Manager bootstrap. The e-TB Manager bootstrap process varies from system to system, but it doesn't take less than 10 seconds, and may take 1 or 2 minutes, depending on CPU and memory availability.

You may also run e-TB Manager from Maven, which only requires you to go to the root folder of the project source code:

|  |
| --- |
| mvn exec:exec -P development |

The -P argument is to inform Maven to run e-TB Manager using the development profile.