

**Fusion242 Ltd**  
White Lodge  
Horton Road  
Horton, SL3 9NU

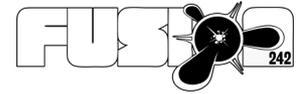
support@fusion242.com  
www.fusion242.com  
**Company Representative**  
Ben Walshaw

# FDM

## Field Data Manager Installation Guide v1.3

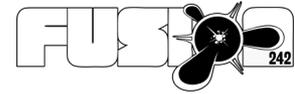
### Statement of Confidentiality

This document and supporting materials contain confidential and proprietary business information of fusion242 ltd. These materials may be printed or photographed for use in within your organisation, but are not to be shared with other parties without consent from fusion242 ltd.



## Tables of Contents

INTRODUCTION	3
PRE-REQUISITES	4
PREPARING THE DATABASE	5
DEPLOYING FDM ON TOMCAT	6
<b>The <code>fdm.properties</code> File</b>	6
<b>Deploying the FDM WAR Files</b>	8
<b>Advanced Configurations of Packages</b>	9
CONFIGURATION FILES	10
PREPARING A COMPANY INSTANCE	12



## Introduction

This document contains instructions for installing the Field Data Manager product from Fusion242.

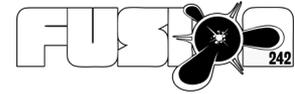
It covers how to:

1. Prepare the database
2. Install 3<sup>rd</sup> party pre-requisites
3. Deploy the web application components
4. Edit the FDM properties file
5. Manage the configuration folder
6. Change logos and other components of the core system

For instructions on using the system please see the:

1. FDM Administrators Guide
2. FDM Operation Guide
3. The Fusion242 YouTube channel

Information available via our website ([www.fusion242.com](http://www.fusion242.com))



## Pre-requisites

Field Data Manager is written using the Java programming language.

It requires a number of 3<sup>rd</sup> party components to operate, which are all available for free download.

1. Tomcat 9+
2. Java (JDK) 1.8+
3. MySQL Server

Other compatible technologies can be used but would need to be tested.

This guide will show the basic process for deploying the FDM software onto Tomcat and MySQL, although more advanced configuration of these products will not be covered.

## Preparing the Database

The FDM software utilises a database, hosted on MySQL.

1. Install the MySQL server and take note of the install host name and ports
2. Ensure that the database servers' time zone is set to UTC (<https://stackoverflow.com/questions/19023978/should-mysql-have-its-timezone-set-to-utc>). The FDM database connections will set their session values to use UTC time, however, for external queries and processes the global time zone should be set to UTC and returned queries converted to local time zone.
3. Create a database (e.g. called `fdm`) which will hold the FDM tables
4. Create a user and assign it to the database
5. Using the command line or MySQL workbench software, load the **000\_FDM\_CoreStructure.sql** file from the installation package.
6. With your database selected, execute the SQL script. This will build the structures for the tables.
7. The next step is to install any initial data records required. These are held in the script **001\_FDM\_CoreData.sql** script. Load this and execute it against the database.

After this process the database will contain the tables required for the FDM software to function.

If the database is not set to store datetime and timestamp fields as UTC there may be problems when using the default data importer within FDM as this uses UTC as the base time zone format.

When you run the FDM application for the first time and log in, you will see a window appear that allows you to patch the database to the required version.

This may happen when deploying any new versions, so you should always aim to log into the system after deploying a new version using an admin account. Once the patches are applied the system will operate correctly – not applying patches may cause the FDM software to experience errors.

## Deploying FDM On Tomcat

First install the Java1.8 JDK. This is required to run Tomcat and the FDM software.

Download and install the Tomcat software.

Run the Tomcat server by executing the start script from the bin folder.

Visit the location <http://localhost:8080> (if running on your local machine, otherwise the hostname of your server) and you should see the welcome page from Tomcat. This means that the software is functioning and is ready for FDM to be deployed.

### The `fdm.properties` File

Before starting the FDM software in Tomcat, we need to prepare our configuration file. This is included in the installation package.

The `fdm.configuration` file should be stored in one of the following locations:

1. `\etc\fdm.properties`
2. `C:\fdm\fdm.properties`

Save the `fdm.properties` into your chosen location and then open it within an editor.

Edit the values in the `fdm.properties` file to match your installation and requirements. The values available are:

<code>apiurl</code>	This is the location or the API as used by the client and API. If it is installed on the same Tomcat instance then this should be 127.0.01 as the host name, plus the location of the API e.g. <code>http://127.0.01:8080/api/rest/</code>
<code>apiexternalurl</code>	This is the API url that is referenced externally. For example in a password reset request email. e.g. <code>https://api.mydomain.com/rest/</code>
<code>apisystems</code>	This is a comma separated list of valid system identification strings that will be accepted by your installation. The first will be used by the FDM client and API. If you need to add a new system (e.g. a client that wants to connect to your FDM installation) you can add a new key and distribute to them. E.g. 1234, 1235, 1236

apikey	This is a key that is used by the client software to access API calls that are restricted from external use. It will be loaded by the API and Client and used to match on requests. The list is comma separated, with the first key being used by the FDM Client application automatically.
apirunasuser	The username of an account that will be used to run shared processes in the API. In multi-instance installations this account must exist in each instance as a system admin.
apirunaspassword	The password for the user account.
apirunaschannel	A unique channel ID used for authenticating the user. This should be a random 8 character key to ensure it doesn't clash with logging in to the client application.
debuglevel	Configures the level of logging to be used by FDM. Values are DEBUG, INFO, WARN, ERROR For production deployments you can use WARN or ERROR, and for development and prototyping installs use INFO (or debug if necessary).
licensekey	This is your license key for the FDM software and is used to aid with support and maintenance requests
runmode	FDM client can function in two modes, CLOUD and SINGLE In cloud mode a button is shown on the login screen that lets a user create a new instance within FDM. This is required on installation to create your first account. Once an instance is created this can be switched to SINGLE to prevent additional company instances from being created.
jdbc_class	JDBC class name. For MySQL just use : mysql
jdbc_servername	The hostname of the MySQL server (e.g. localhost)
jdbc_databasename	The name of the database holding the FDM tables
jdbc_username	The username required to access the database
jdbc_password	The password to access the database
maxaccountusers	How many users can an account/instance hold (e.g. 10000) This can be increased or decreased at any time
maxaccountdatastores	How many data stores can be created for an account/instance. This can be increased at any time
scripttimeoutseconds	How long can a script in the client execute before being force terminated. This should be increased as required depending on the scripts in the system.
configuration_directory	The location to find the other configuration documents used by FDM and included in the installation package. E.g. /etc/fdm/configuration For windows the path should use double slashes for directory names, e.g. c:\\fdm\\configuration

login_url	The URL of the FDM client installation login page. E.g. <a href="http://localhost:8080/fdmclient">http://localhost:8080/fdmclient</a>
email_hostname	SMTP server location
email_username	User name
email_password	Password
email_from	Email address to send emails from
support_email	Email address for support emails
admin_email	Email address for admin emails
email_subject_newaccount	This subject shown to new account creators on their welcome email.
email_subject_validationcode	No longer used
email_processor	Should this installation of FDM process the email queue and send emails generated by the system. 0 for no and 1 for yes.
url_newaccounthelp	URL for directing new users to help documentation and other materials
url_help	URL to send users to when clicking on Help from the FDM client
cloud_jdbc_class	Not currently used
cloud_jdbc_connectionstring	Not currently used
cloudtempopath	Folder location to use for temporary files used by the FDM software (e.g. /tmp/fdm/)
phantomjs_location	Location of the PhantomJS software installation This is completely optional and is an option for converting HTML reports into PDF. It is no longer used by the client.
advert_space_title	Not currently used
advert_space_url	Not currently used

Once the `fdm.properties` file has been deployed and prepared, the FDM software can be deployed into Tomcat.

## Deploying the FDM WAR Files

The FDM software is provided as two WAR files. These are archives that can be quickly deployed into the Tomcat application server.

Alternatively, the WAR files can be ‘unzipped’ using standard ZIP compliant software. The resulting directory structure can be copied in the application server.

When deploying the software images into Tomcat we would advise stopping the server first. This allows system administrators time to clear out logs and also deploy any additional files into the deployment (such as custom logos or script sandbox permissions etc.).

For the first time installation:

1. Stop the Tomcat server if it is running
2. Make sure you have the configuration files (next section) deployed on to your server. These are necessary before creating your first account – the server will still start without them but the software will not function correctly.
3. Clear the logs directory of all files
4. Clear the webapps folder of any contents not required
5. Copy the FDM client war file into the webapps folder.  
It is recommended to rename this to either a shortname (e.g. `fdm.war` or `fdmclient.war`) or to `ROOT.war` if you want the client available from the root url of Tomcat (e.g. <http://localhost:8080/>)
6. Copy the FDM API war into the webapps folder.  
It is recommended to rename this to either a shortname (e.g. `api.war` or `fdmapi.war`). This will become the URL for the API's that are referenced in the `fdm.properties` document. E.g. if you name it `fdmapi.war` the URL for the api will be: <http://localhost:8080/fdmapi/rest/>  
The `/rest/` part is added by the API deployment automatically
7. Restart the Tomcat server
8. View the `catalina.out` file in the logs directory to check that the software starts correctly. You can do this using a command such as: `tail catalina.out -f`
9. If the server starts successfully visit the home page by visiting: <http://localhost:8080/>  
in a browser (remember to add your shortname to the url if you didn't name the package `ROOT.war`)

## Advanced Configurations of Packages

The deployed WAR file for the client has a number of documents that can be edited or replaced to customise the core look and feel of the product.

These are:

Look and Feel files are located in  
**VAADIN/themes/fusion242/**

They include the `logo.png` file used for the default logo, and the CSS documents setting colors and other visual aspects of the application.

## Configuration Files

The installation package includes a zip file containing various configuration files. These should be extracted into the folder described for configuration in the `fdm.properties` file.

These files are used by the FDM client for various purposes and include the login screen.

### **Initialsettings.properties**

These are the initial account settings applied to new accounts/instances.

The account name and cloud temp path shortcuts can be left as the default value.

The storage type can be file or s3

For S3 storage specify the bucket to be used by the account instance. A sub-folder will be created in this bucket using the account ID by default (`<!ACCOUNTNAME!>`)

The `loginurl` property should be changed to match your installation directory.

The `emailfrom` should also be set – it is used to send emails from the system such as welcoming new users.

**The `installurl` property should be configured** to point at the URL where the FDM installation is accessed from. It is needed when linking to the FDM services for viewing and downloading files from outside of the client.

### **sandboxclasses.config**

This document lists classes that can be accessed from within the FDM scripting sandbox. If a class package is listed here it can be used, otherwise it will be blocked, causing an error exception.

### **sandboxfactories.config**

This document lists a class that should be created in the FDM script engine. This will be created using the variable name specified in the file. For example:

```
myFactory=com.fusion242.pojo.factories.MyClassOfFunctionality
```

The scripting engine will create an instance of the `MyClassOfFunctionality` class and make it available to your scripts as variable name 'myFactory'. For example,

```
myFactory.doMyFunction();
```

### **loginpage.html**

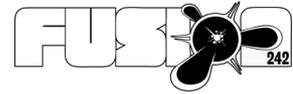
This page can be edited to customise the experience delivered to visitors of the FDM client.

There are a number of placeholders present that must be preserved, although their positioning can be changed in the HTML

These are identified by the `<div location="...">` tag structure.

A Copyright message identifying ***fusion242 ltd*** ([www.fusion242.com](http://www.fusion242.com)) as the copyright holder of the software must be displayed unless written permission has been given to remove this.

Example:



Fusion242 Field Data Manager (FDM) is Copyright Fusion242 Ltd ([www.fusion242.com](http://www.fusion242.com)) All rights Reserved.

### **welcomeemail.html**

This is the email template that will be sent to new users in the system. It is not relevant to any particular instance, so cannot be edited in the client.

### **pdfviewer.html**

This is the page template used to display a PDF document inside the client via the file viewing service.

### **imageviewer.html**

This is the page template used to display image documents inside the client via the file viewing service.

### **videoplayer.html**

This is the page template used to display a video inside the client via the file viewing service. It may require a number of pre-requisites to be installed on your server. An example of setting this service up is available on the fusion242 website. Search for 'video player'.

### **iconlist.txt**

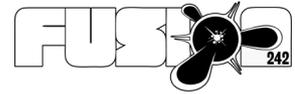
The icons that are available within the system are listed here. To add icons they must be listed in this file and also added to the icon library folder within the VAADIN folder.

From the folder you installed the client app on your server this will be :

VAADIN/themes/fusion242/icons/library

There are then two sub directories : 16 and 32

These need to contain a 16x16 pixel version of your icon and a 32x32 pixel version.



## Preparing A Company Instance

On the initial login page, assuming that the `fdm.properties` run mode is set to CLOUD, there is a sign up button.

Click on this and prepare your initial account information. This will create an instance ready for administration and configuration.

If preparing the system for use within a single organisation set the `fdm.properties` run mode to SINGLE which will remove the capability to add new account instances.

An account instance is a unique installation, which can have its own users, object structures and data storage, completely separate from other accounts. The cloud mode is useful if you want to provide unique solutions for multiple clients or departments using a single installation of the FDM software.

Administrators can toggle CLOUD mode on and off as required to create new accounts without making the capability public to general users.