

# Measure Authoring Tool Installation Guide

The Measure Authoring Tool (MAT) was designed using many open source products including the Google Web Toolkit (GWT) framework, Java JDK, MySQL and Eclipse. GWT allows a developer to write client side code in Java and GWT converts it to JavaScript. The MAT uses MySQL as its backend database server and the IDE is Eclipse for Java EE Developers.

Due to the use of these open source products, a working knowledge of Java development and some research into how the products work with each other in your environment may be necessary. Research from discussions on product forums, help documents, internet searches and knowledge of the local environment where the MAT will be running may all need to be checked if there are errors during the install.

## Install Eclipse

Download Eclipse (MAT has been tested for Oxygen) for Java EE Developers. Eclipse IDE is a free, open source IDE for writing Java applications found on the Eclipse download page at

<http://www.eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/oxygen1a>

1. Select the version that best fits your operating system.
2. Extract the zip file to a location where you would like Eclipse to be installed.
3. Run the application file.

## Install Google Web Toolkit (GWT)

1. Install GWT Plugin for Eclipse
  1. Open Eclipse IDE, navigate to the workbench, and select **[Help] -> [Eclipse Marketplace]**
  2. Search for <http://marketplace.eclipse.org/content/gwt-eclipse-plugin>
  3. Install the result with the title **GWT Eclipse Plugin 3.0.0**

## Configure Java (JDK)

1. The application has not been tested with version above Java 1.8; please ensure this version is in the environment.
2. Verify that JAVA\_HOME and PATH system variables are pointing to the proper folder(s).
3. For example, in a Windows environment, the JAVA\_HOME (Environment Variables under Advanced System Settings (should point to the Java SDK 1.8.x folder and PATH should point to the Java 1.8.x/bin.

## Configure Maven

1. Install and Configure Maven: <https://maven.apache.org/install.html>

## MAT Code Base Import

1. From the MAT GitHub source code link, extract the code base into the Eclipse workspace folder in the MAT Environment.
2. Import the code base into an Eclipse project:  
**[File] -> [Import] -> [Maven] -> [Existing Maven Projects] ->**

Browse to and Select << workspace >> -> **[Finish]**

## Installing Local Maven Dependencies

There are two dependencies that need to be installed locally for the Measure Authoring Tool project to work. You can use the following commands to install them. For more information on this command, check out this documentation: <https://maven.apache.org/guides/mini/guide-3rd-party-jars-local.html>.

```
mvn install:install-file -Dfile=</path/to/project/mat/lib/CQLtoELM-{version}.jar> -DgroupId=mat -DartifactId=CQLtoELM -Dversion=<version> -Dpackaging=jar
```

```
mvn install:install-file -Dfile=</path/to/project/mat/lib/vsac-{version}.jar> -DgroupId=mat -DartifactId=vsac -Dversion=<version> -Dpackaging=jar
```

After running the two commands, right click on the Measure Authoring Tool project -> Maven -> Update Project (ensure force update is checked) -> Ok.

## Bonnie (Optional)

1. MAT is integrated with the Bonnie application. For this integration to be successful locally, you must obtain Bonnie application credentials and update the run configurations.
2. Bonnie's homepage can be found at <https://bonnie.healthit.gov/>.
3. **Note:** MAT can still run without updating the Bonnie credentials. However, any Bonnie-related features will not work locally until the run configurations have been successfully updated.

## Create MAT Database

1. Install MySQL (MAT currently uses MySQL Community Version 6.x) available from MySQL.com: <https://dev.mysql.com/downloads/installer/>
2. Run the MySQL community server installer for your operating system and the MySQL workbench (which comes with the download).
3. Create a new MySQL Connection to localhost.  
**Note:** Make sure to keep track of the username and password you used as you will need this later.
4. From the MAT Code base, find the mat\_schema.sql file and then execute the script in the database that was just created.

## For App Server Use (Optional)

**Note:** If MAT is to be run on an application server, the developer will need to run the build to create a .war file.

1. Run a mvn clean install
2. After the build has run, the file is placed into a target folder under MAT as follows:  
mat/target/MeasureAuthoringTool.war

## Configure MAT files for Local Development Environment

**Note:** Changes will need to be made to some of the files in the code base to allow the MAT to connect to local MySQL Database.

1. Comment out the <cookie-config> tag at the bottom of the web.xml file.
2. mat/war/WEB-INF/mat-persistence.xml
  1. In this file, change the 'dataSource' bean to provide login credential to the local MySQL database. The box below is an example of the mat-persistence.xml file. The username and password values indicated with '**username**' and '**password** (bolded)' need to be changed to the "username" and "password" you used in the previous step to set up your local database.

```
<bean id="dataSource" class="org.apache.commons.dbcp.BasicDataSource" destroy-
method="close">
  <property name="driverClassName" value="com.mysql.jdbc.Driver"/>
  <property name="url" value="jdbc:mysql://localhost:3306/MAT_APP_BLANK"></property>
  <property name="username" value="username"></property>
  <property name="password" value="password"></property>
</bean>
```

3. Mat/war/WEB-INF/applicationContext-mail.xml
  1. Make the **host value property name (bolded)** changes in the box below to the 'mailSender' bean to point to the proper mail server.

```
<bean id="mailSender" class="org.springframework.mail.javamail.JavaMailSenderImpl">
  <property name="host" value="mail.some-server.org"/>
</bean>
```

2. Make the **from value property name (bolded)** changes in the box below to the 'templateMessage' bean to set the correct 'From' email address.

```
<bean id="templateMessage" class="org.springframework.mail.SimpleMailMessage">
  <property name="from" value="someaddr@someaddr.org"/>
</bean>
```

## Setting up Program Arguments

Program arguments are used for GWT to determine how to run the application.

**[Run] -> [Run Configurations]** -> Select your MAT project on the Left Hand Side and in the **[Arguments]** tab -> add the following information below into the Program arguments box.

```
-logLevel INFO -port 8888 -remoteUI "${gwt_remote_ui_server_port};${unique_id}" -codeServerPort 9997 -war\war
mat.Bonnie mat.Mat mat.Login
```

## Setting up VM Arguments

VM arguments are used to pass environment specific parameters to the application. These parameters include the VSAC URL; Bonnie Client ID, Client Secret, and Redirect URL; Encryption Algorithms; and logging settings.

**[Run] -> [Run Configurations] ->** Select your MAT project on the Left Hand Side and on the **[Arguments]** tab -> add the following information below into the VM arguments box.

```
-Xmx512m
-DVSAC_DRC_URL=https://vsac.nlm.nih.gov/vsac
-DSERVER_TICKET_URL=https://vsac.nlm.nih.gov/vsac/ws/Ticket
-DSERVER_SINGLE_VALUESET_URL=https://vsac.nlm.nih.gov/vsac/ws/RetrieveValueSet?
-DSERVER_MULTIPLE_VALUESET_URL_NEW=https://vsac.nlm.nih.gov/vsac/svs/RetrieveMultipleValueSets?
-DSERVICE_URL=http://umlsks.nlm.nih.gov -DENVIRONMENT=DEV -Dlog4j.ignoreTCL=true
-DPROFILE_SERVICE=https://vsac.nlm.nih.gov/vsac/profiles
-DVERSION_SERVICE=https://vsac.nlm.nih.gov/vsac/oid/
-D2FA_AUTH_CLASS=mat.server.twofactorauth.DefaultOTPValidatorForUser
-DBONNIE_RESPONSE_TYPE=code
-DBONNIE_REDIRECT_URI=https://yourredirectURI.com
-DBONNIE_CLIENT_ID=1234567890
-DBONNIE_URI=https://bonnieURL.org
-DALGORITHM=EncryptionAlgorithm
-DPASSWORDKEY=PasswordKey
-Dlog4j.configuration=PathToLog4jPropertiesFile
```

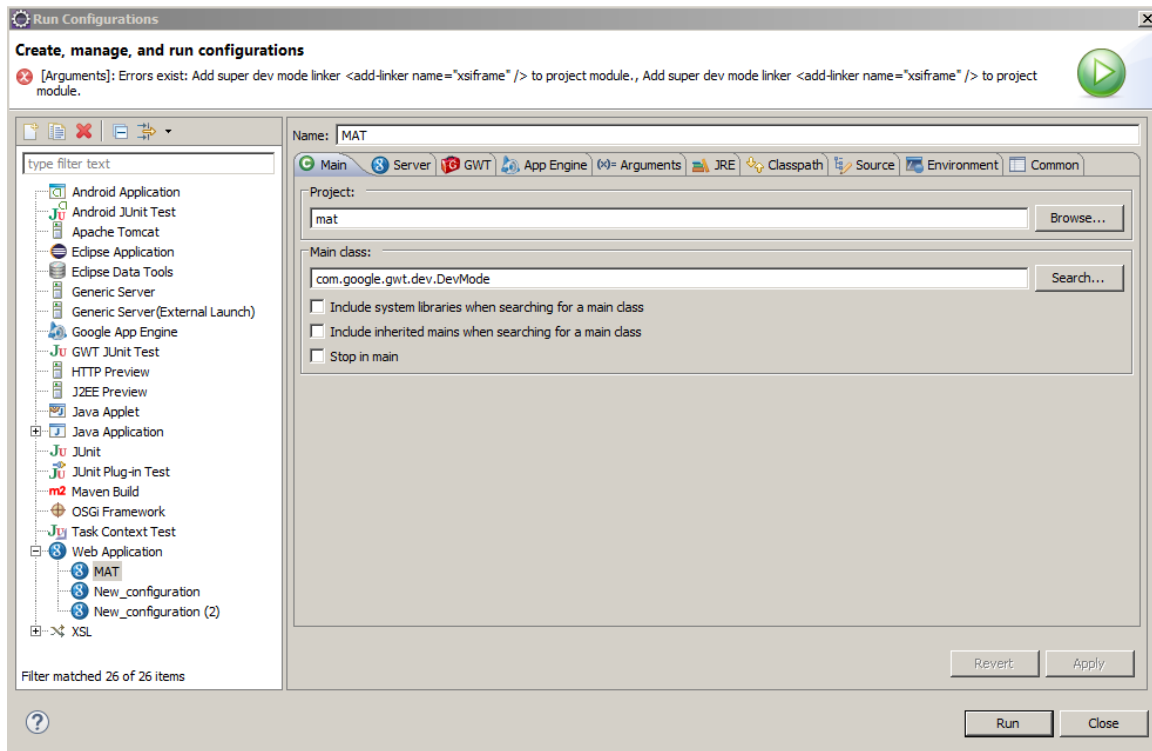
## Compile the MAT Code

1. Right click on the MAT project folder, then go to **Properties -> GWT -> General Settings** and select **Use GWT**. Make sure that the SDK is using GWT 2.8.1.
2. In Eclipse, select GDT Pulldown (Red Google Icon Button) -> **[GWT Compile Project]**.
3. In the GWT Compile Wizard:
  1. Browse to find the MAT project.
  2. Set the Log level to Debug
  3. Add the Login, MAT, and Bonnie Entry Point Modules.

## Run MAT

1. Make sure your database server is running
2. In the eclipse IDE Select, **[Run] -> [Run Configurations] -> [Web Application] -> [New]**
3. Set Main Class: "com.google.gwt.dev.DevMode".
4. Select **[Apply]** and followed by **[Run]**.

Figure 1: Eclipse Run Configuration Window



## Login in to MAT

1. To login to MAT, open MySQL Workbench and run the following queries:
  1. **SELECT \* FROM USER where USER\_ID='Admin'**  
Look at the **LOGIN\_ID** column, the value there is your UserID. The password default is '**gargleBlaster\_10**'.
2. Navigate to the MAT log in page GUI and use the UserID and password from the previous step and log in to MAT.
3. Once logged in, navigate to the **[Mat Account]** tab and enter the Admin user details under the **[Personal Information]** tab and the **[Security Questions]** tab to setup user's security questions.
4. To change the password to something new, use the **[Password]** tab.