Administration

Updating a few data-driven files that need to be customized to the specific user's environment and business use-case.

Configuration files

There are two types of files used by the application

1) DB Directory: The configuration details related to the products/materials, the record of the material/product updates are stored here
2) Reports Directly: The generated inventory reports are stored here.

The following are the configuration files to be updated to suit your system. They are populated with sample values and are self-explanatory. Just update/add to suit your requirements

MyInvenotoryMgmt.config : The DB and Reports directory paths are specified. These paths should have been created already with necessary read/write access

MaterialDesc.csv: Mapping of the Material ID to its corresponding description.

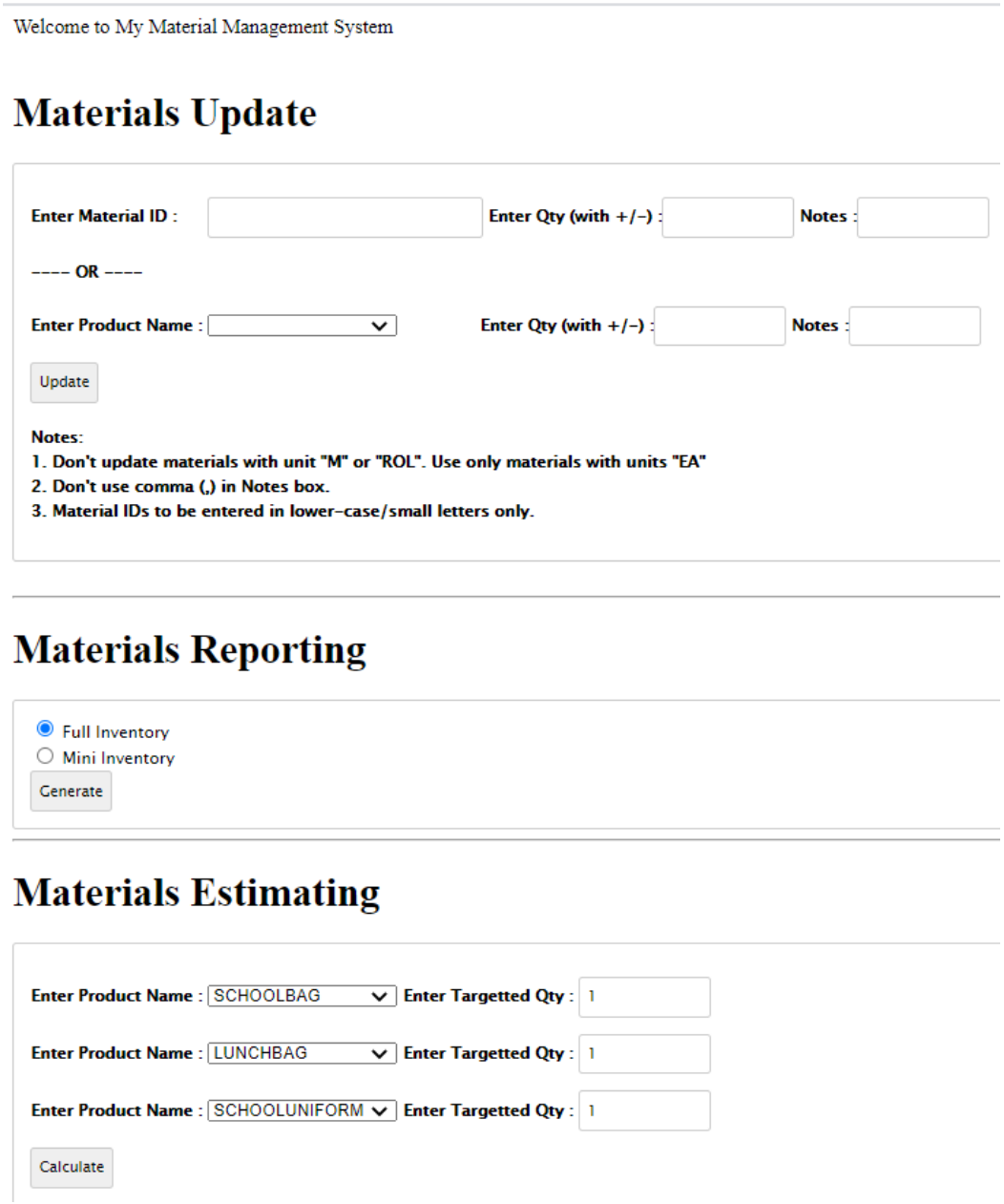
MaterialMiniInventory.csv: A list of Material IDs that are considered to be inventoried for a mini-inventory. Generally, the high-values materials could be placed here.

ProductBOM.csv: Product Name followed by the Material IDs with respective quantities that are needed to assemble that product.

Transaction.csv and ProductTransaction.csv: There two files are the logs of the updates performed by the user. The are updated/used by the application. Should be empty when starting the application for the first time.

Deploying/Running as a WebApp

1) Download the WAR and properties file from InventoryManagement/tree/master/webapp to the webapps folder of your apache-tomcat installation directory.
2) Start the Apache Tomcat web server
3) The application is accessible via http://localhost:8080/MyInventoryMgmt/Materials



1) **Update Flow**: Enter the material ID and its corresponding quantity that is being added to the stores. (+ for adding / - for removing). While dispatching a finished product, no need to update individual material. The product name and a negative quantity can be updated. The corresponding materials for that product as specified in the BOM will be removed from the inventory.
2) **Report Generation Flow:** Generate a full or mini inventory in a .csv file.
3) **Estimating Flow:** To know how much shortage/excess of materials is foreseen, if a specified target of various products to be assembled. The report will also tell the max. products of each type that can be assembled within existing inventory stock.

Building Source Code

Pre-requisite

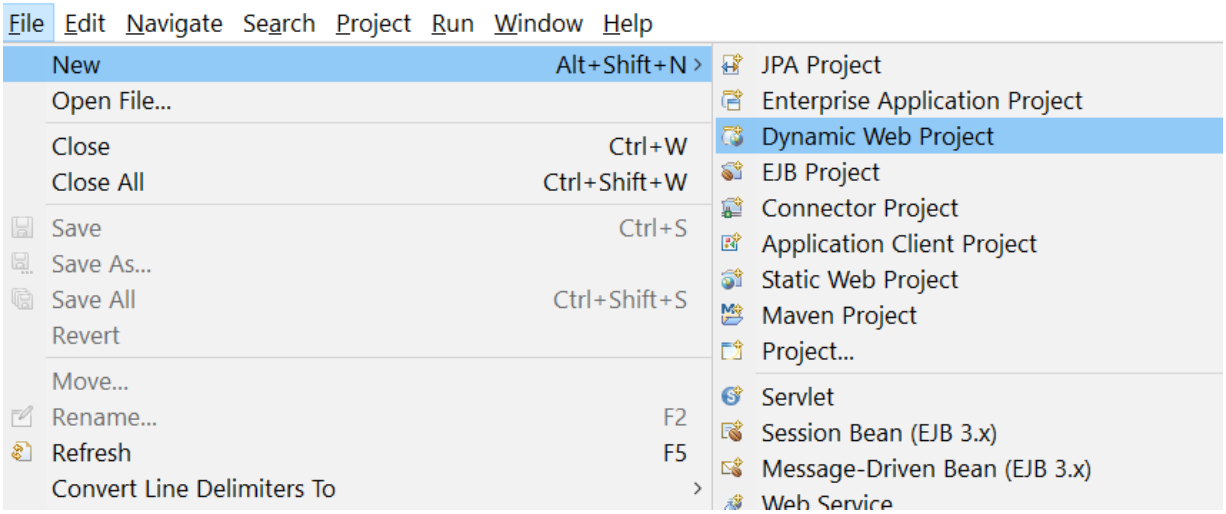
Installed JRE 8, Eclipse IDE for Java EE Developer and Apache Tomcat 7.

The version mentioned with which the project was tested, however other versions should also work.

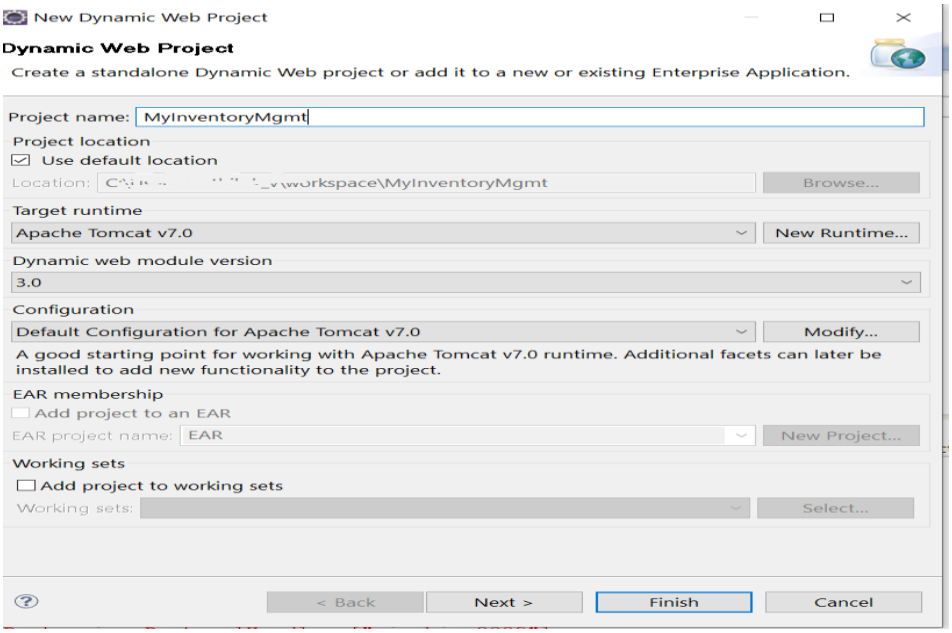
You are able to run the HelloWorld Servlet like mentioned here - https://medium.com/@prashant.srivastava7744/how-to-create-hello-world-servlet-example-using-eclipse-ide-with-tomcat-7-39fdbdc21573

Import the source code

1) Create a 'Dynamic Web Project"

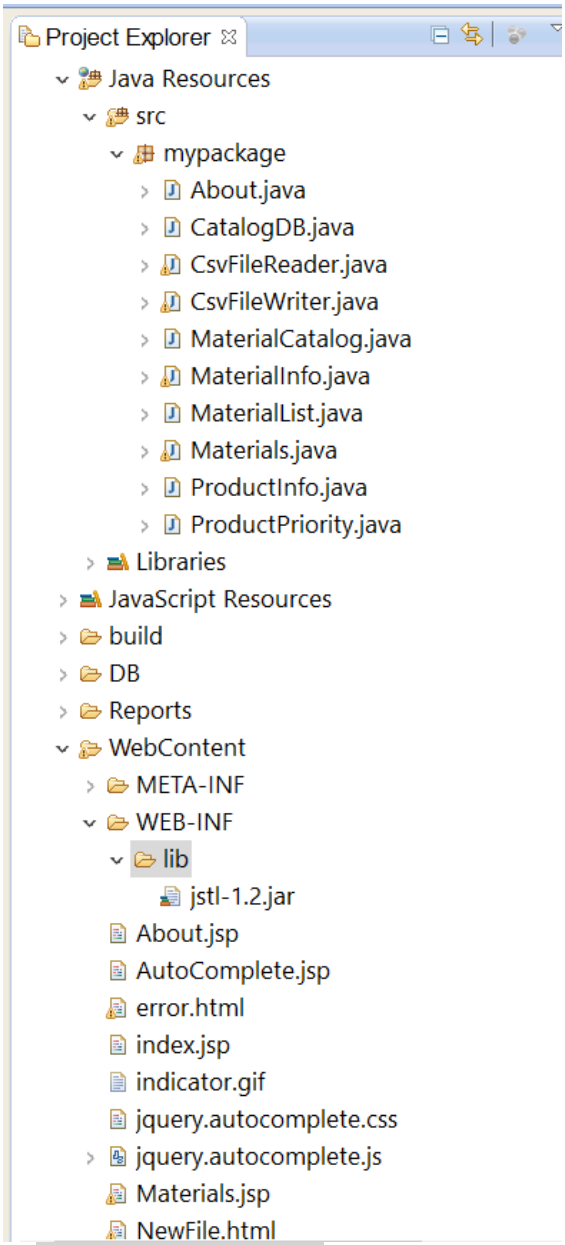


2) Give a name to the Project



3) Copy the contents of the Src directory in InventoryManagement/tree/master/webserver/TestServlet/src (github) into the Src directory of the Project created in the previous step i.e. MyInventoryMgmt\src (local workspace)
4) Similarly copy the contents of .jsp, .css, .html and .js files from the WebContent directory from InventoryManagement/tree/master/webserver/TestServlet/WebContentinto (github) into the WebContent directory of the Project created in the previous step i.e. MyInventoryMgmt\WebConten (local workspace).
5) Add the jstl-1.2.jar (publicly available) into the MyInventoryMgmt\WebContent\WEB-INF\lib (local workspace)

6) So, you should see a structure like this in Eclipse project explorer



7) The `ReadPropertiesFile()` has to be updated to specify an absolute path to the properties file based on your choice of the location of the MyInvenotoryMgmt.config file

With this, the project should be able to build and run successfully.