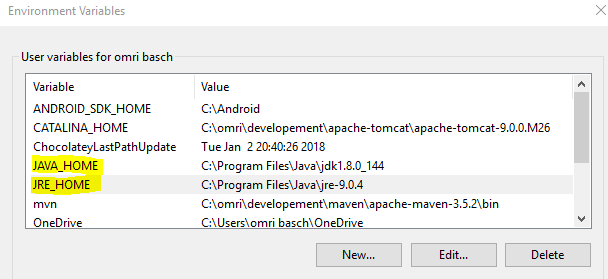
**Topic in Computer Security project readme (Omri Basch 203536917)**

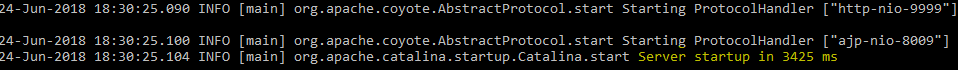
This document specifies how to setup the project, shows a common use case, and suggests how to test the project requirements.

**Server Setup:**- **Note**: “FileManager” mentioned in this doc is the folder in the zip I submitted, the project folder.

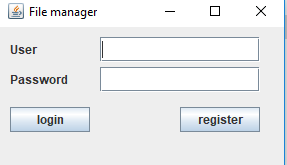
The following steps are required to setup and start the server for the project:

1. Download Apache Tomcat 9 (if you don’t have it already): go to <https://tomcat.apache.org/download-90.cgi> and click the “zip” under “Core”.  
   Extract the downloaded zip to wherever you like. For the rest of this document we’ll refer the “apache-tomcat-9.0.8” folder you just extracted as “**TOMCAT\_HOME**”.
2. Go to FileManager/tomcat and **copy** the folder “UploadServletApp”. **Paste** it in TOMCAT\_HOME /webapps.
3. Go to FileManager/tomcat and **copy** all three xml files: **context.xml , server.xml , web.xml.** **Paste** them in TOMCAT\_HOME/conf (overwrite existing).
4. If you don’t have Java Development Kit 8 installed, download it at <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html> and install.
5. If not set already, set JAVA\_HOME and JRE\_HOME environment variables:  
   on Windows, press Windows key + R to open Windows Run prompt. Type sysdm.cpl into the input field and hit Enter or press Ok. in the new window that opens, click on the Advanced tab and afterwards on the Environment Variables button in the bottom right of the window. Under “User variables for …” click “New”, type “JAVA\_JOME” as **Variable name** and the installed jdk folder (probably something like C:\Program Files\Java\jdk1.8.0\_144) as **Variable value**, and hit “OK”.  
   Under “User variables for …” click “New” , type “JAVA\_JRE” as **Variable name** and the installed jre folder (probably something like C:\Program Files\Java\jre-9.0.4) as **Variable value** and hit “OK”.  
   After that, the two User variables should appear as so:

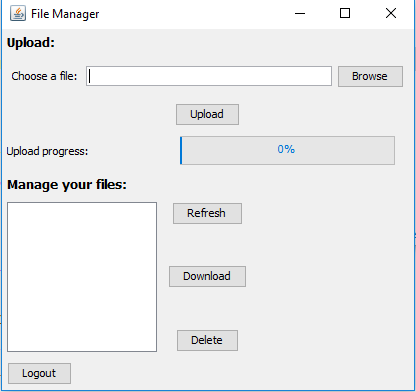
**Server Startup:**

To start the server, go to TOMCAT\_HOME/bin folder, and run the “**startup.bat**” file (alternatively run it from cmd). This will open a cmd window with server startup logs, give it a few seconds. Eventually you should see a log message that looks like this:

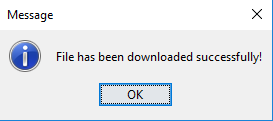
This means that the server startup worked.  
  
**Running the Client Software:**After server startup, simply go to FileManagerfolder and double click on “FileManager” jar. The following window will open:



Now you can test the app:  
After registering a user and performing login successfully, this window will open:

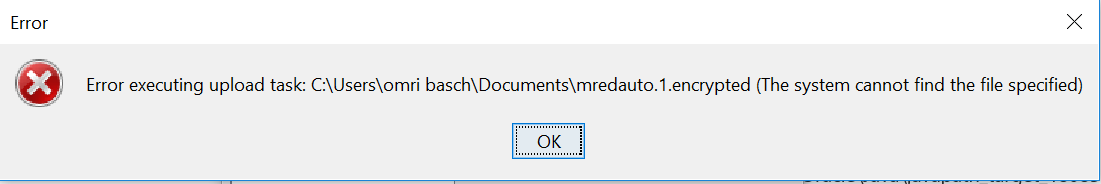


**Example use case:**

1. **User uploads files to server**: the user clicks “**Browse**” and select the file it wishes to upload, then clicks “**Upload**” button (Note the Upload progress bar that fills as the upload process progresses). After successful upload, the uploaded file name will show up in the list under “**Manage your files**”.
2. **User downloads a file from server:** the user selects a file from the list it wishes to download by clicking on it, then clicks on the “**Download**” button and selects the download destination. After successful download, a message would appear:  
     
   after hitting **OK** the user can view the downloaded file at the specified destination.
3. **User deletes a file from server:** the user selects a file it wishes to delete from server by clicking on it, then clicks on “**Delete**” button. After confirming the delete operation, the file will no longer appear in the list of files in server (and would be deleted from the server of course).
4. **User logs out:** the user clicks on the “**Logout**” button, and returns to the login/register window.

**Testing project requirements:**

* Inside TOMCAT\_HOME/webapps/UploadServletApp/WEB-INF/classes/users (from now on called **USER\_HOME**) you will find a folder for each user that registered. Note that these user folder names are not encrypted, to make testing the project easier.
* Inside each user’s folder there are the encrypted files that the user uploaded to server. you may upload a file via the client and then see the encrypted file on the server by accessing **USER\_HOME** /%the user name%.
* Inside each user’s folder there is another directory called “**auth**”, in it there is a file called “**auth.txt**”. This file is the special authentication file. Again, note that the “auth” folder name and the “auth.txt” file inside it do not have encrypted names, to make testing the server side easier.

You may test authentication by changing a user’s encrypted file on server at **USER\_HOME** /%the user name%, and then login to the user FileManager client, or hit the “**Refresh**” button if that user is already logged in.  
  
  
**Note:** If this error pops up when trying to upload a file: 

You are probably trying to upload a file from a controlled folder (like the “Documents” folder). Try and upload files from a different location **or** go windows defender security center -> virus & threat protection -> Virus & threat protection settings -> Manage Controlled folder access -> Allow an app through Controlled folder access -> Add an allowed app and add C:\ProgramData\Oracle\Java\javapath\_target\_180851843 java.exe