

Our work can be evaluated using one of these two available options:

Option 1: Separate install in which frontend files and a backend jar file can be downloaded and tested separately

No servers needed for this type of testing.

Option 2: Complete install which enables calling the backend Java logic from Angular front end

This will ask you to set up both NodeJS and Apache Tomcat server. Guide for doing this is provided in this document.

Option 1: Separate install in which frontend files and a backend jar file can be downloaded and tested separately

1)Download the front-end source code from github(MSDFrontEnd). Review the code from a Web IDE of your choice. You can also run the front-end if you install node+express+mongodb in your machine(See option 2 for details).

2)For backend testing, download the jar called as execPlagDetect.jar in your local machine.

3)To run that jar from your command prompt, go into the directory in which that jar is installed and run it using the command : java -jar execPlagDetect.jar

The program will ask you for path1 and path2.

So for example if your java files reside in C:\User\Projects\Sorters then you should exactly type C:\User\Projects\Sorters when path1 is asked. Similarly give path2. Make sure that none of your folder names have any blank space in them. For eg C:\User\My Projects\Sorters wont work because of space between My and Projects.

Sample:

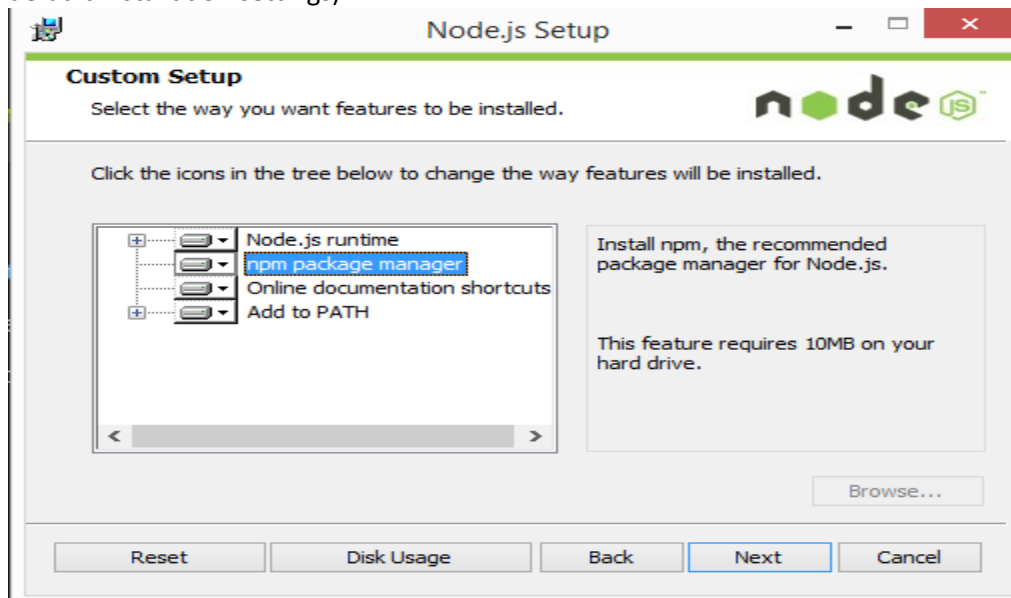
```
C:\Users\Yash Shah>cd desktop
C:\Users\Yash Shah\Desktop>java -jar execPlagDetect.jar
Enter path 1
C:\Temp\sorters
Enter path 2
C:\Temp\sorterstry
[sort, sort, createMaxHeap, createMaxHeap, maxHeapify, maxHeapify, swap, swap, m
yCompareIo, myCompareIo, myCompareIo, myCompareIo, sort, sort, recursiveSort, re
cursiveSort, randomizedPartition, randomizedPartition, partition, partition, myC
ompareIo, myCompareIo, myCompareIo, myCompareIo, sort, sort]
C:\Users\Yash Shah\Desktop>
```

Option 2: Complete install which enables calling the backend Java logic from Angular front end

- 1)Download the source code from github(MSDFrontEnd and MSDBackEnd)
- 2)Install Node.js

Installation Steps

1. Download the Windows installer from the [Nodes.js® web site](https://nodejs.org/en/).
2. Run the installer (the .msi file you downloaded in the previous step.)Follow the prompts in the installer (Accept the license agreement, click the NEXT button a bunch of times and accept the default installation settings).



3. Restart your computer. You won't be able to run Node.js® until you restart your computer.

3) Install MongoDB

1. [Download MongoDB](#) and install on the local development machine as described in the [MongoDB documentation](#). These instructions assume MongoDB was installed in /usr/local/mongo on a Mac
2. At the root of the file system, create a directory called data and in there create a directory called db, e.g., /data/db. This directory is where MongoDB will store all files related to the databases managed by MongoDB
3. Add the MongoDB binaries to the system PATH environment variable so that the database can be executed from anywhere in the terminal command line. Assuming MongoDB was installed in /usr/local/mongo, add the bin directory to the system path by adding the following line to the ~/.bash_property file

```
export PATH=/usr/local/mongo/bin:$PATH
```

4. Test that MongoDB starts by typing `mongod` at the command line. The command line will print several lines. Some of the lines should say something similar to

MongoDB starting : pid=56659 port=27017 dbpath=/data/db

...

waiting for connections on port 27017

5. Make sure MongoDB is running before you start Node.js.

4) After installation, Open `mongo.exe` to open the `mongodb` command prompt.

Create a database using command “`use YOUR_DATABASE_NAME`”.

Once it is created, edit the `server.js` to connect to that database by editing this line:

```
var connectionString = 'mongodb://localhost/YOUR_DATABASE_NAME';
```

5)Open the command prompt. Run “`npm install`” to install all the dependencies.

6) Once you have node successfully installed, go to the directory where the `server.js` file is located using command prompt/terminal. Use the command “`node server.js`” to run the frontend. The application will need port 3000. Make sure this is available.

6)Go to `localhost:3000` to see the webpage.

7)Open the `MSDBackEnd` project in editor of your choice. If you see errors with respect to package name, create a new package in the `src/main/` folder and copy all the files into that folder, and remove all the files from the other directory.

8)Add jar files from the library folder to build path.

9)Run `MsdServicesApplication.java`. This should start the apache tomcat server at port 8080.

10)Once you are in `localhost:3000`, navigate to “Project” link. You will be directed to a login page. Click on register to register as a new user. Once that is done, login to the application. Now give links of the java projects on your local machine. For example if your java files reside in “`C:\User\Projects\Sorters`” the you should exactly type “`C:\User\Projects\Sorters`” in input field. Similarly give link for second project. Click on “Perform Test” button to run the plagiarism test.