**BIOS Application Development Setup:**

Perquisites:

1. JAVA = 1.8+ JDK installed and have JAVA\_HOME environment variable set to the installation folder.
2. MAVEN = Copy the latest maven 3.0+ installation folder and have M2\_HOME and M2 environment variable set to the installation folder.
3. IDE = Get the latest Spring Tool Suit IDE installed and check the JRE installed and Maven settings in the preference.
4. GIT = Get the latest GIT installed.

5. MySQL = Install latest MySQL database

**BIOSSERVER Project Setup:**

Perquisites:Spring Boot, Spring Data JPA and MySql Setup

1. In Spring Tool Suit create new “Maven project”.
2. For the Spring Boot Application add the below dependencies in pom.xml

<parent>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-parent</artifactId>

<version>1.5.10.RELEASE</version>

<relativePath/>

</parent>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-data-jpa</artifactId>

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-devtools</artifactId>

<optional>true</optional>

</dependency>

Also Add the start class from where application will start

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

<project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>

<start-class>com.bios.portal.BIOSMaster</start-class>

<java.version>1.8</java.version>

</properties>

1. Create the application.properties for the configuration of the app as below:

server.port = 9000

spring.datasource.url=jdbc:mysql://localhost:3306/yamuna

spring.datasource.username=root

spring.datasource.password=admin12345

spring.datasource.driver-class-name=com.mysql.jdbc.Driver

spring.jpa.show-sql = true

1. Create required packages and create the Application Main Class to start the application.

@SpringBootApplication

**public** **class** BIOSMaster {

**public** **static** **void** main(String[] args) {

SpringApplication.*run*(BIOSMaster.**class**, args);

}

}

1. Have the controllers, services, repositories and entities class to run the app.

**BIOSCLIENT Project Setup:**

**1. Install Node.js for Angular**

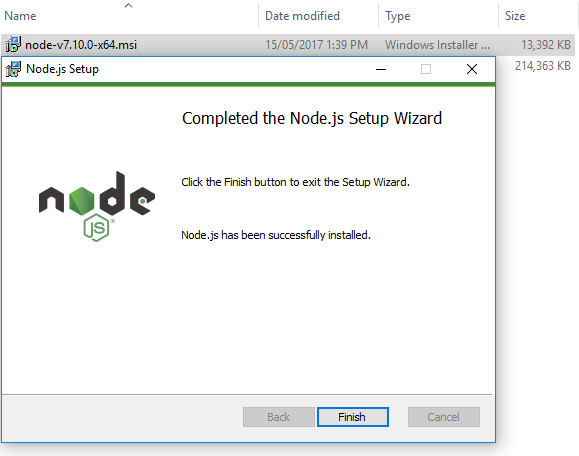
Firstly, checking whether or not you have Node.js installed, by command:

node -v & npm -v.

If the command goes unrecognized, we must install **Node.js**.

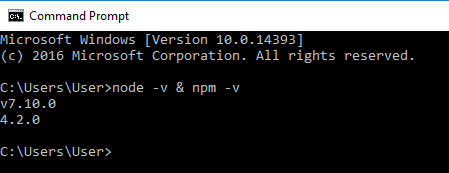
– Go to: [nodejs.org](https://nodejs.org/). Download Node.js installer

Double click on it and follow the guides to setup Node.js, successfully result:



– Open **cmd**, checking **Node.js** by command line:

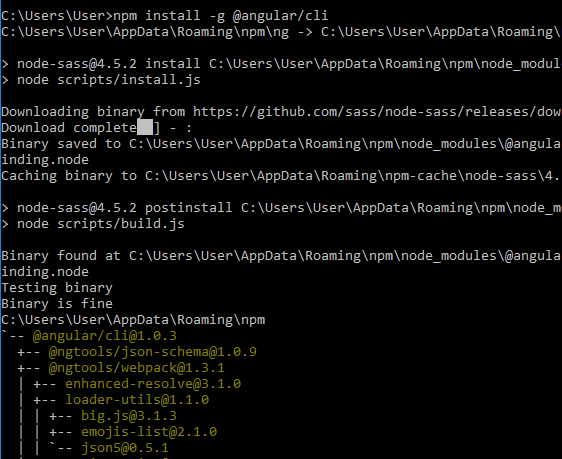
 node -v & npm -v:



##### 2. Install Angular-CLI

– Using command line

npm install -g @angular/cli:



– Check **Angular-CLI** after setup by command line

ng -v:



The **CLI** will install an **Angular 5**

##### 3. Create BIOSClient client project

##### Location of the SpringBoot project at:

##### C:\vw\vwworkspace

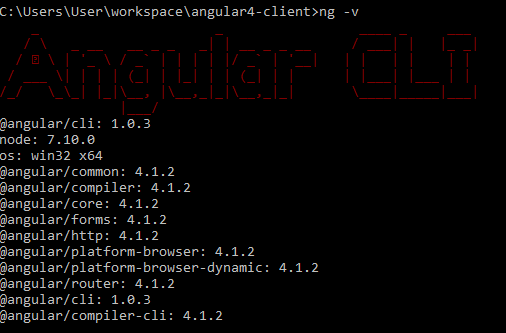
##### Now, open **cmd**, cd to C:\vw\vwworkspace

##### – Start a new **angular 5** project by command line: **ng new BIOSClient**:

##### C:\Yamuna workspace\doc\angular4-springboot-springtoolsuite-install-Angular4-client.png

To check angular version, go to **BIOSClient** folder, type:

ng -v:



Start **BIOSClient** project by cmd ng serve --open, results:

**Note: To hit BIOSserver, replace the host \_url in below location of BIOSClient.**

**BIOScCient\src\app\utils\constants.ts**