# Node js

* It is a free and open-source JavaScript interpreter/server environment (written in c++) that allows us to run JS outside of a web browser
* Platform independent
  + Meaning it can run on any OS
* Made from google chromium V8 engine

# NPM

* Node Package Manager
  + It is included with the download of Node.js
* Similar to Maven or Nuget Package Manager, it can manage our dependencies and versions and this is done through a **package.json**
* It will create a folder called node\_modules in our angular app

## CLI

* Npm install -g typescript – install typescript globally for us
* Npm install -g @angular/cli – installs angular cli globally for us

# What is Typescript?

* It is an open-source language and **object-oriented** language developed by Microsoft
* Essentially, it made JavaScript into an actual object-oriented language
* It is superset of JavaScript
  + It has all the features of JavaScript but with more.
* It isn’t perfect and has problems with polymorphism
* It is **strictly-typed**
  + Datatypes are enforced in typescript

## CLI

* Tsc -v -will give you the current version of typescript
  + This might give you a problem if you didn’t set your execution policy to RemoteSigned
    - To set it up, open powershell as an administrator and run set-executionpolicy remotesigned and hit y for yes
* Tsc [filepath] – transpile the TS into JS
* Tsc [filepath] -w -will automatically transpile our TS into JS every time we save that file

# Angular

* It is an open-source framework TypeScript based front-end framework
* Angular first started as Angular JS (JavaScript based), but in Angular 2 they switched from JS to TS
  + Angular JS did not support dynamic loading of the page or server-side programming language (Less speed)
  + Angular 2 better support for mobile devices
  + They changed architecture design **from MVC design to a Component and Directive design**
* It uses a component structure
  + A component is a single unit in Angular that displays some sort of visual to the user.
  + It is comprised of HTML, CSS, TS, and (optional) speck that represents a section/view of the application

## Single Page Application

* Allows us to route to pre-loading components (JS, HTML, CSS) onto the same page without reloading the page
* Allows us to navigate in the same page without refreshing the whole page.

## Advantages

* It is more mobile friendly
  + Mostly because once it is loaded, the user can still navigate through the website without having to reload the page
  + Very useful for users with unstable internet connection (cellular network)
* Caching capabilities (it can store information in your system, so you don’t have to keep pulling it in)
  + It will synchronize your local data once connection is restored

## Disadvantages

* Doesn’t perform well with SEO (Search engine optimization)
  + Us human can perceive the webpage easily by the visual of the website, but a robot doesn’t see things visually, it only sees the files it retrieves and Angular uses JS to dynamically add and remove elements, so your html doesn’t really have information that displays the website unlike what you see.
* Initial load of the page might take a long time.

## CLI

* Ng version – gives you the current version of your angular cli
* Ng new [projectName] – Creates a project for you
* Ng serve – Will run that angular application

# Angular Architecture

## Node\_module

* Provides NPM packages for the entire workspace

## Package.json

* It tells us what our angular application needs from the node\_modules folder
* It also tells us the version of the package that we need

## Src Folder

* Contains the source files which gives information about the entire application logic, data, assets

## Assets

* Contains images, audio, video, or whatever external files you need to display your website

## Environment

* This folder contains information that your angular might need to reference on multiple times in your project

## Tsconfig.json

* It is the configuration to how you want your typescript to transpile or change certain properties of it.