1. Node – It’s a JS RUNTIME Environment
2. NPM – Node Package Manager

JS (JavaScript) file will get executed in the browser.

JS Engine

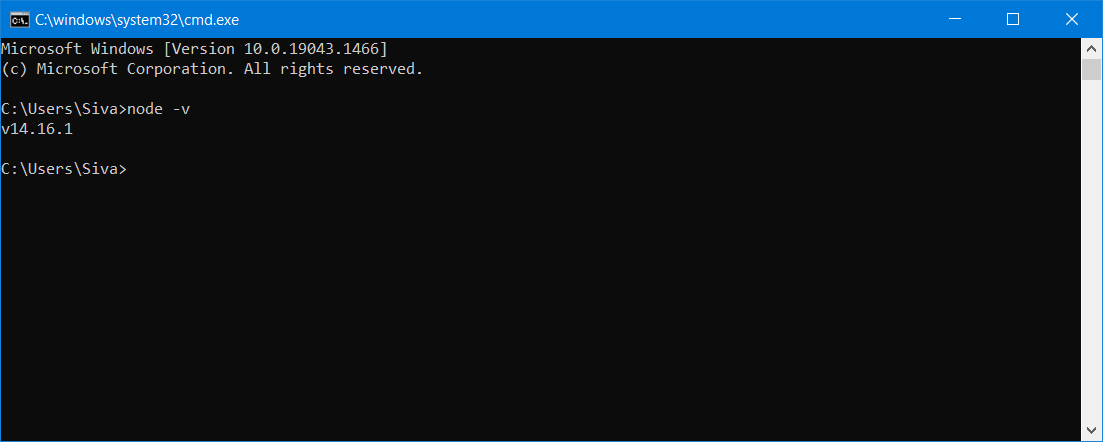
All the browser will consist of 3 engines

1. HTML Rendering Engine (HTML – output)
2. JS Engine (To Run the JS code)
3. Styling Engine (UI & CSS)

Nodejs.org

LTS – Long Term Support

“node -v” -- To see the version of Node installed in your system

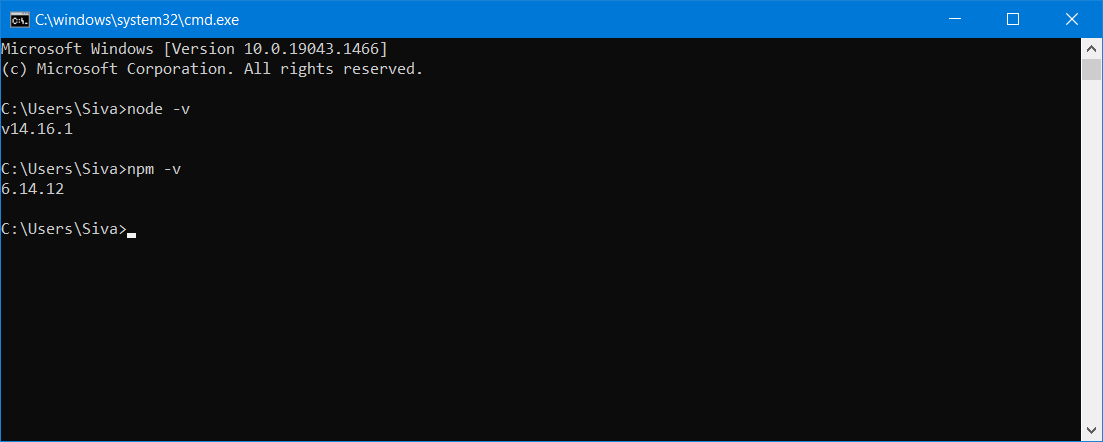


Maven – Project Management Tool for JAVA Applications (pom.xml)

Build, Test, Package, Deploy it & To manage All the dependencies

Npm – It’s a project mgmt. tool for the JS based project.

Npm -v

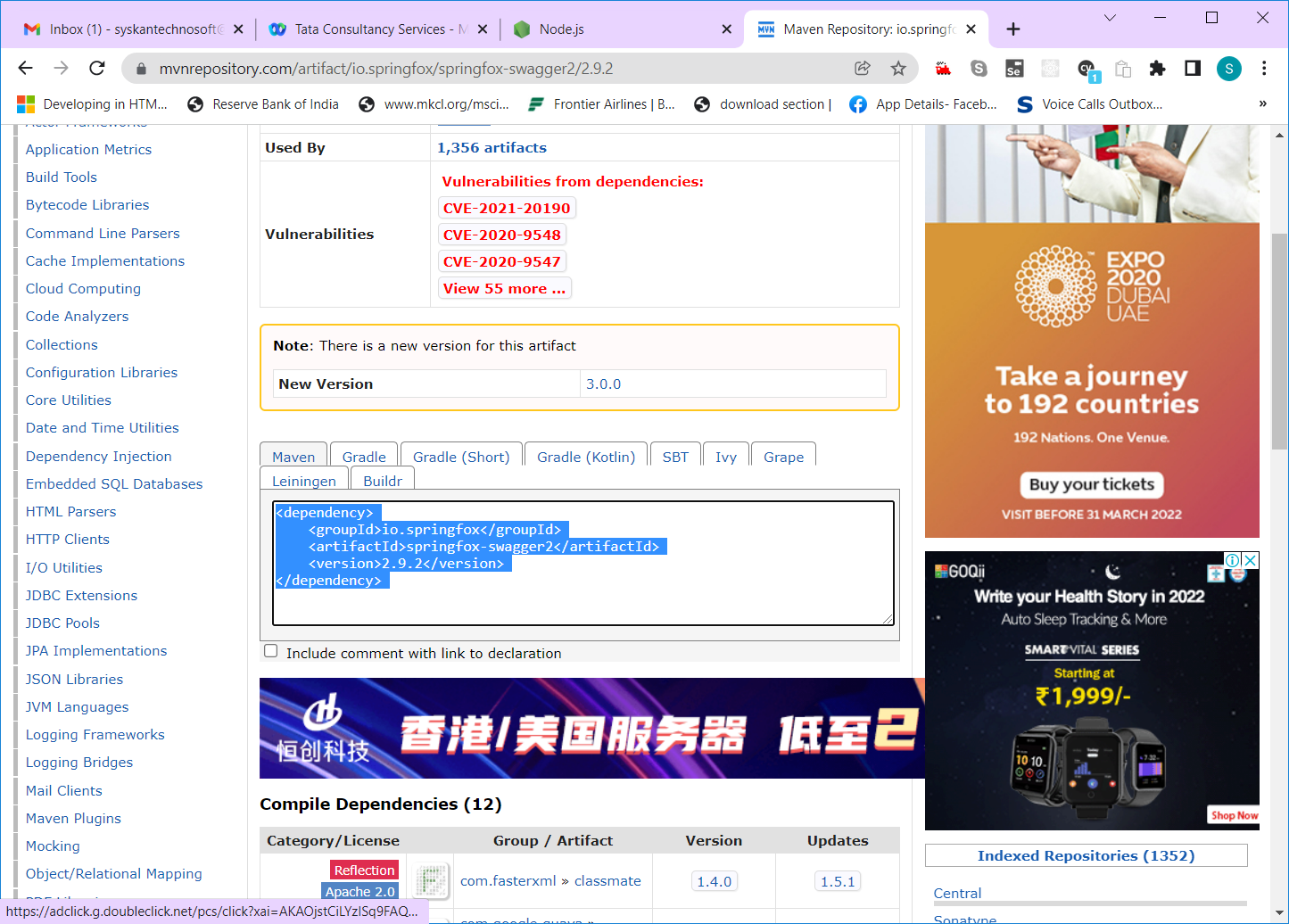


JS based Front End frameworks (Angular, React, Vue….)

Location of NodeJS : C:\Program Files\nodejs

Maven Project --- pom.xml (Backbone) – groupId, artifactId, version

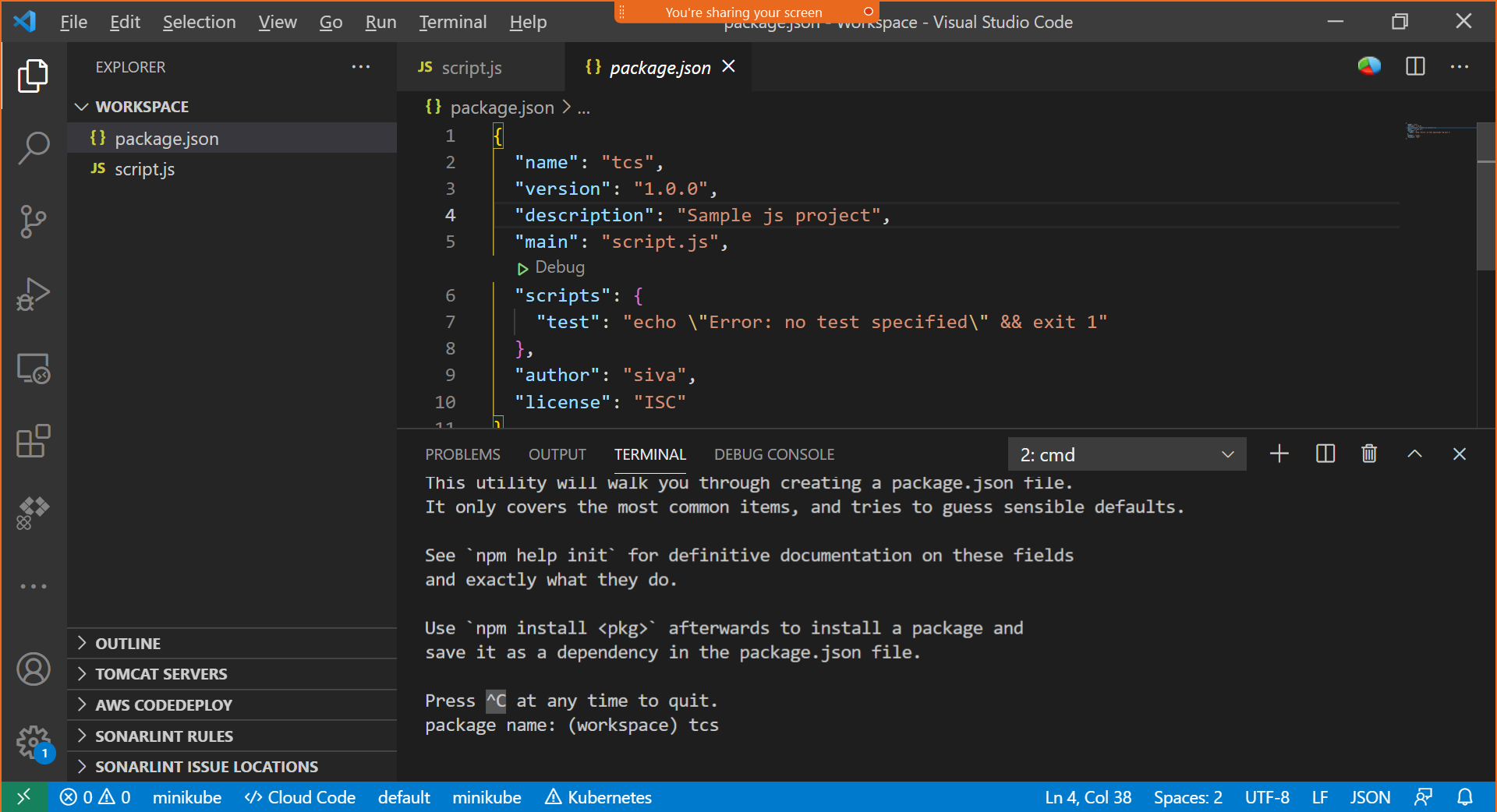
Mvnrepositories.com

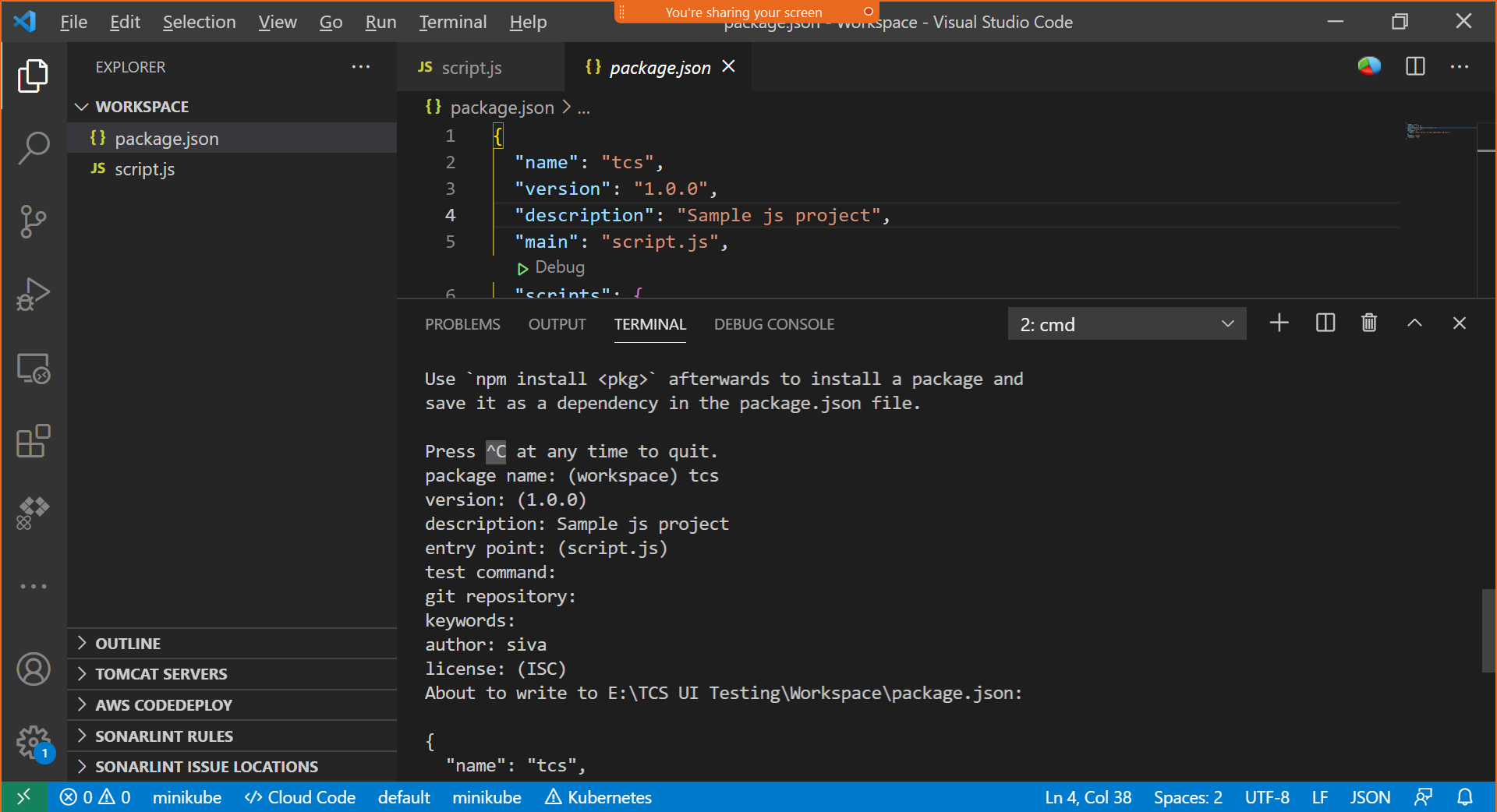


Local Repository (Location in your physical/virtual machine)

Remote Repository (mvnrepository… repo.maven.org)

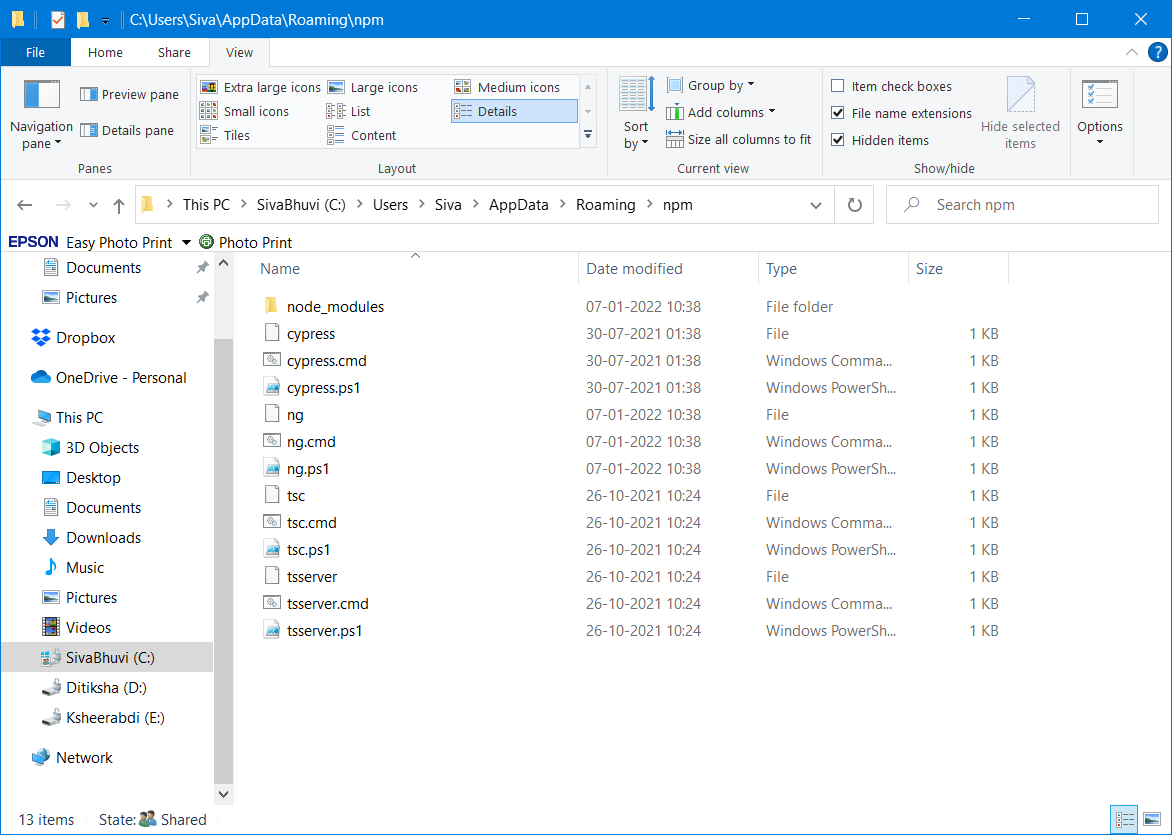
For JS based framework --- package.json -- all the dependencies will be added here





Local Repo – (A Folder where all the project dependencies are installed – node\_modules)

Global Repo – C:\Users\Siva\AppData\Roaming\npm -- Globally installed NPM packages

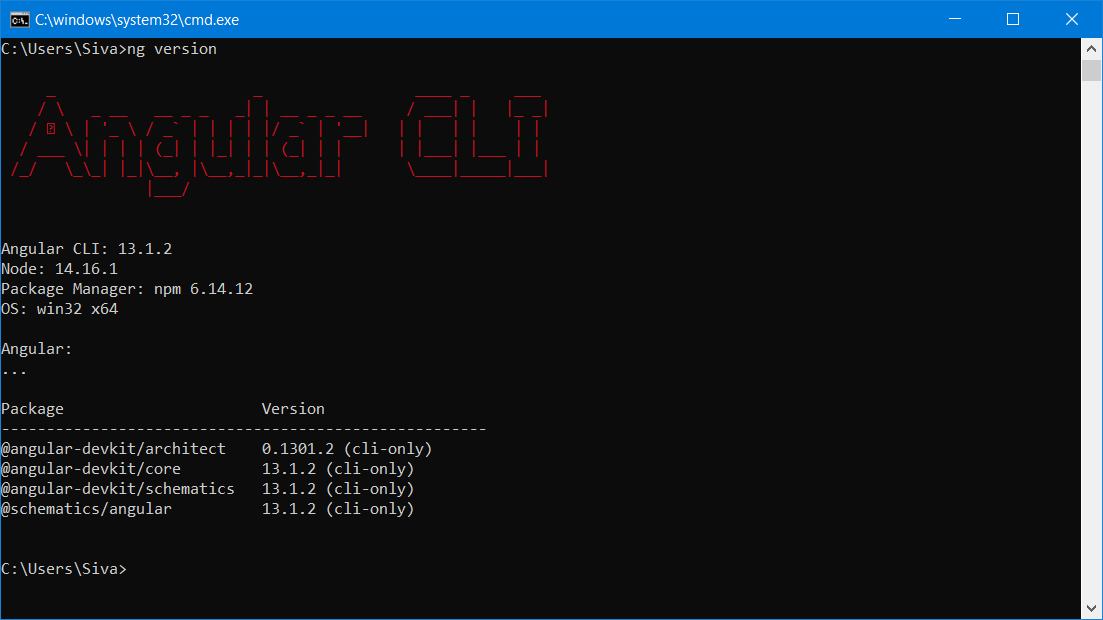


Remote Repo – npmjs.com

Installing Angular Locally – “npm i @angular/cli”

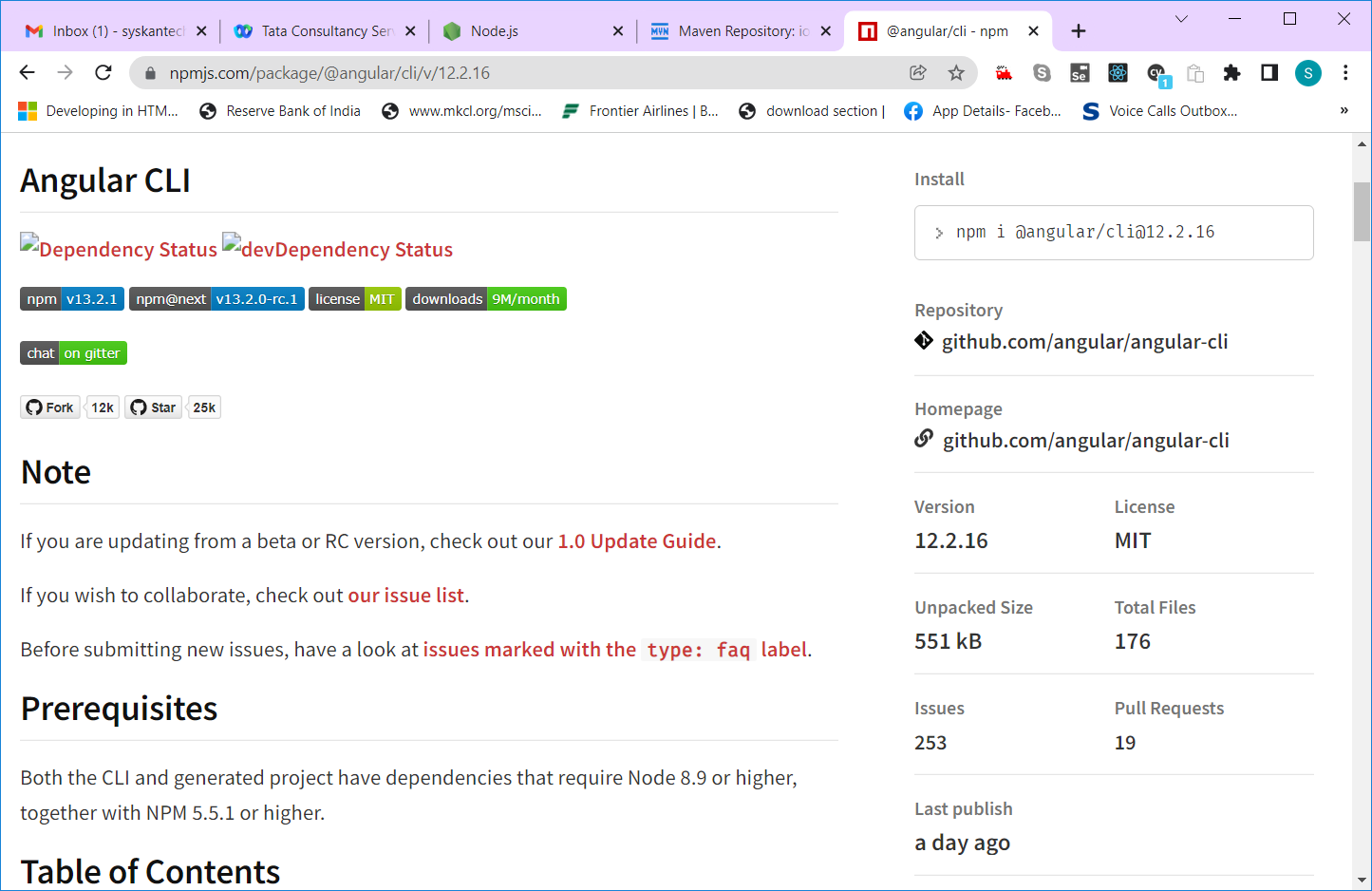
npm install -g @angular/cli – This will install angular cli globally

ng version



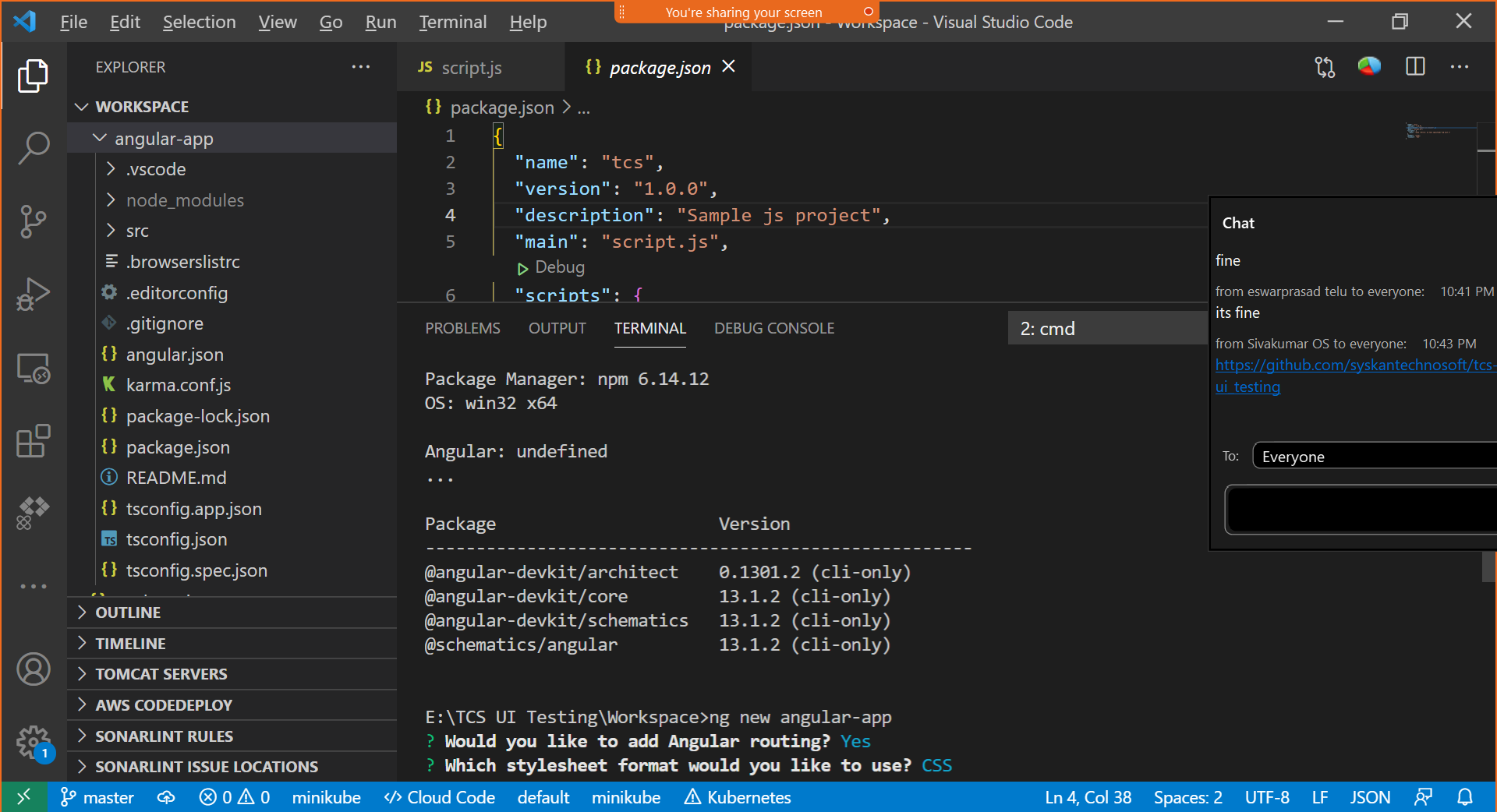
To download & Install specific version of Angular

npm i @angular/cli@12.2.16

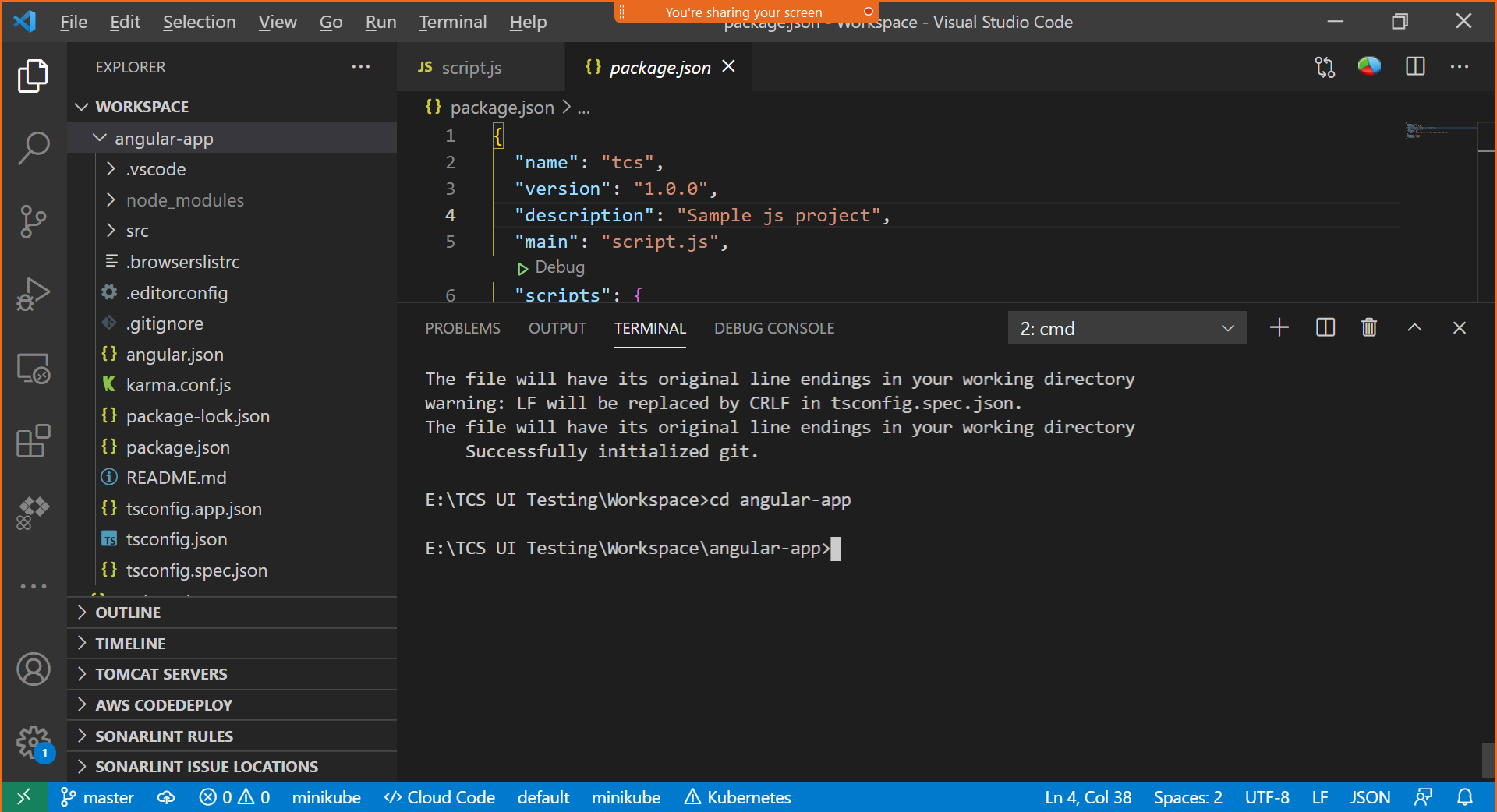


Creating Angular application

Ng new <app-name>



Cd <app-name>



"scripts": {

    "ng": "ng",

    "start": "ng serve",

    "build": "ng build",

    "watch": "ng build --watch --configuration development",

    "test": "ng test"

  },

Npm start === ng serve

Npm build === ng build

Npm watch === ng build –watch –configuration development

Npm test === ng test

“Angular is SPA JS based Framework”

SPA – Single Page Application

Traditional – MPA – Multi Page Application (Web Site – multiple web documents html/jsp/asp/php ….)

Index.html, services.html, aboutus.html, contactus.html, pricing.html, registration.html, login.html

Client – Server Concept – Everytime we request a URL, client needs to wait until server process and sends response for our request.

Client – Server ( Request – Response)

Team Lunch. (5 members)

1. Occupy the reserved table
2. Check the menu card & Order food (Servant – server )
3. Pass the order to kitchen dept
4. Food items will be prepared in kitchen by chef
5. Once completed, the server will take the prepared food and serve it to the table
6. We enjoy the food
7. Pay the bill
8. Leave

Client [ A Human/Computer & A Browser Software]

Server [ A Computer with Server Software – Tomcat/ IIS/ any other Server software]

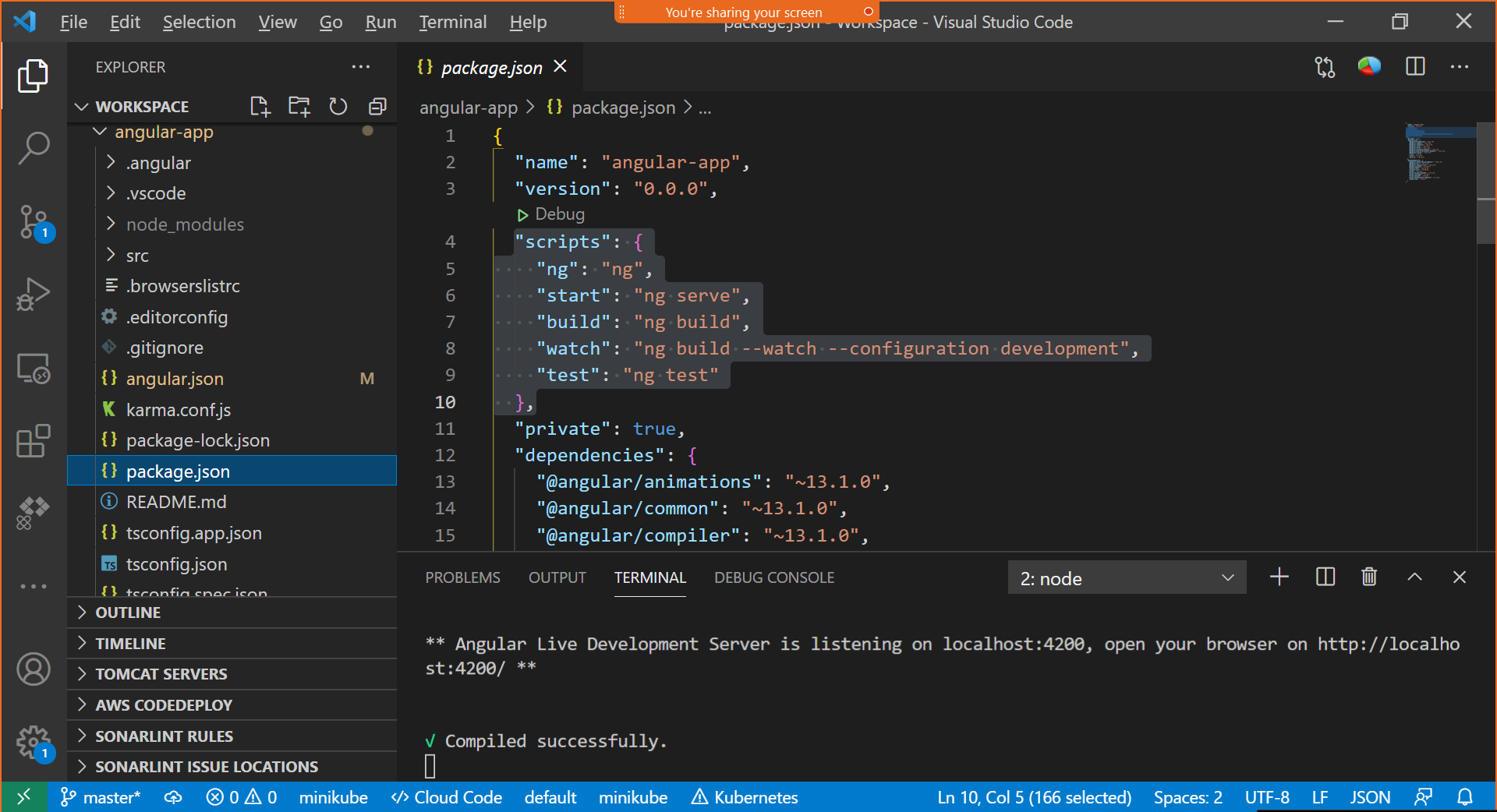
Client (Request)

1. End user types the URL (requested resource) in the address bar and Enter
2. Browser will add more info to the requested resource (request object – headers, data inside the body, auth, cookies, session)

Server (Response)

1. Request object will be passed to server using http/https protocol
2. Server validate the request & process it. (It execute the server side code & generate response object)
3. Response object will be transferred to the respective client

Folder Structure of Angular App:



With the help of angular, creating custom html tag.

<siva> </siva> -- custom html tag

Ng serve – It’s transpiling ts to js and then node will execute these js and generate output

. represents the current folder

.. represents the parent folder

../.. represents the grant parent folder

/ represents the root folder (src)

Decorator – similar to Annotations in JAVA. It starts with @ symbol

It provides meta data to the underlying element (Class, interface….)

Index.html – A4 Sheet – Sticky notes (Component)

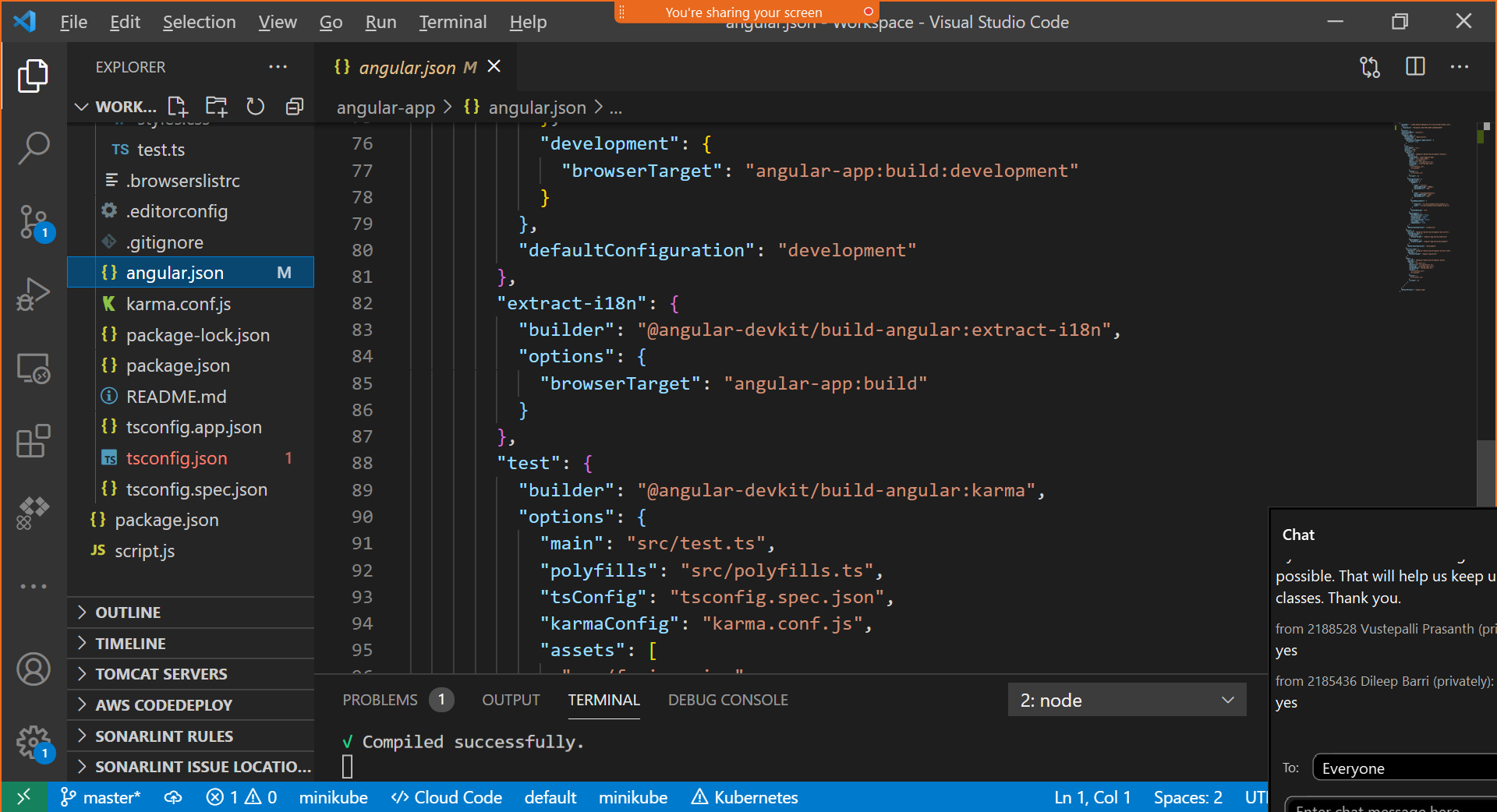
All the Sticky notes (Components) – will be available in the client side.

The entry point of Angular app is main.ts file.

In your angular template, you can’t add <html> & <script> tag

1. Src folder (Angular app & related files)
   1. App (angular modules, components, services, routes )
   2. Environments
   3. Assets (All static files, images, gif)
   4. Main.ts
   5. Index.html (Single Page)
   6. Style.css (global angular style sheet)
   7. Test.ts
   8. Polyfills.ts
   9. Favicon.ico
2. Node\_modules (All the dependencies)

Angular.json – configuration related to production & dev environment app build, config related to testing, build directory details (dist/app-name)



Dist – distribution

Angular – uses a scripting lang called “TypeScript”

TypeScript – Is a SuperSet of Java Script.

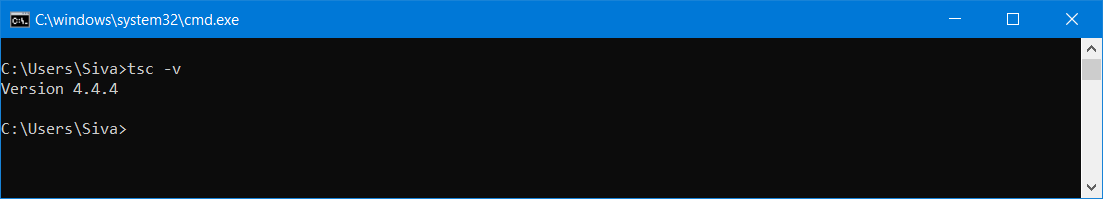
TypeScript is developed & maintained by Microsoft (Strongly typed Object oriented scripting lang)

Class, interface, inheritance, encapsulation except abstraction

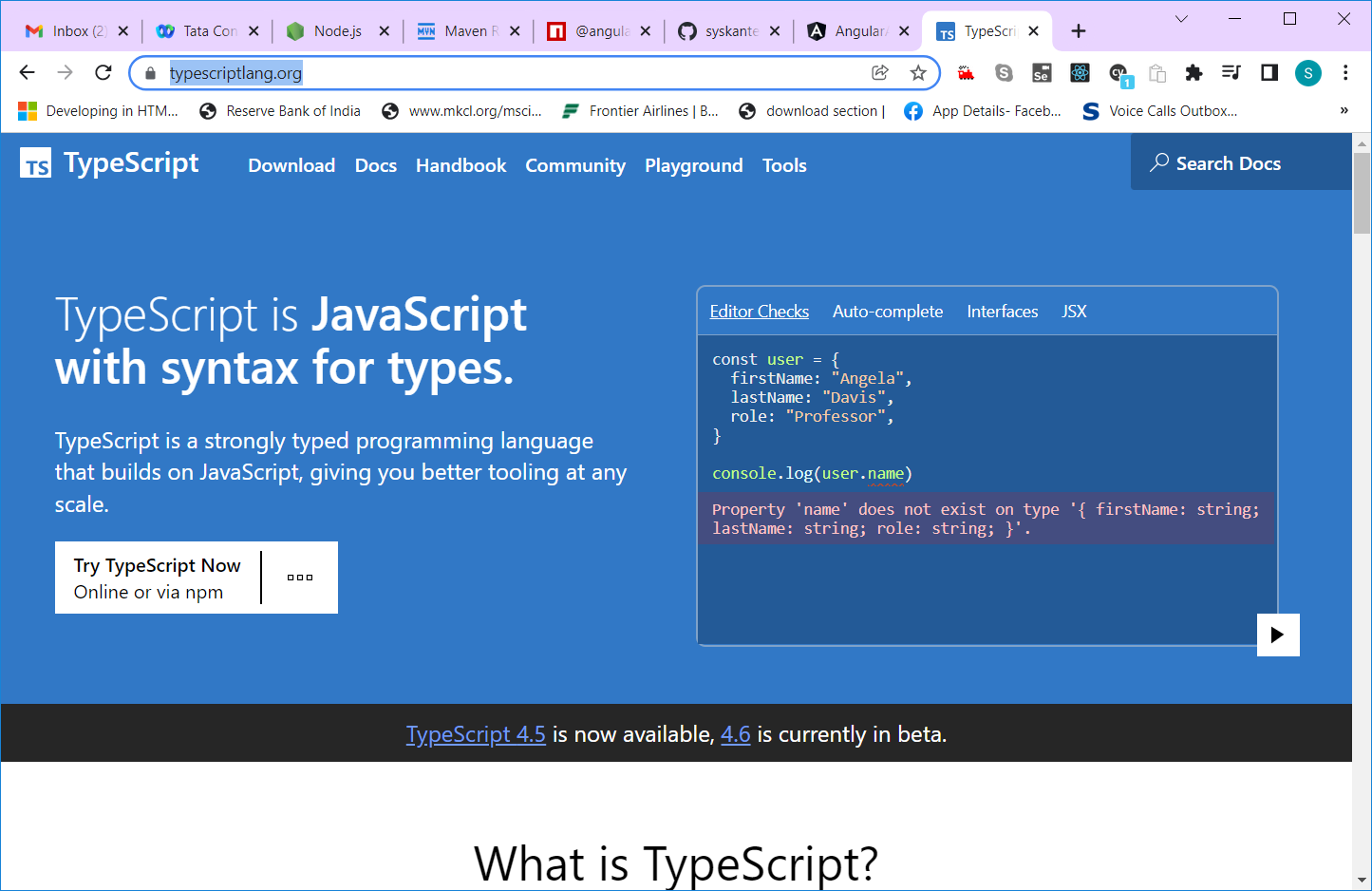
Npm I typescript -- local installation

Npm I -g typescript – global installation

Tsc -v



<https://www.typescriptlang.org/>



Early error detection

JS – finding the errors & resolving the errors

Package name

Importing classes & interfaces from package

Import { } from ‘’

tsc – typescript transpiler (translator + compiler)

tsc will convert .ts to .js file – transpilation

tsconfig.json – config about target js version (tsconfig.app.json & tsconfig.spec.json)

JavaScript is also called as ECMA Script – ES

ECMA – Electronics Computer Manufacturing Association

Karma.conf.js

DOM – Document Object Model – Tree representation of html file

One root (html) – two main branches (head & body) & many sub branches

ng update

ng update --force @angular/core

captureTimeout:120000 (in karma.conf.js)

ng test --code-coverage --browsers=ChromeHeadless --watch=false

npm i -g @angular/cli