# Learning Outcome

Able to develop the real time scenarios based on Node JS applications.

1. Create an Angular application which can create Captcha.
2. Create Responsive Web Application using Angular.

# Activity 1

## Aim: Create an Angular application which can create Captcha.

**Learning outcome:** Able to develop the real time scenarios based on Node JS applications.

**Duration:** 5 hour

**List of Hardware/Software requirements:**

* Git
* Node.js and npm
* Angular CLI
* IDE (e.g. Visual Studio Code)

**Code/Program/Procedure (with comments):**

**Create the Angular application**

* + - 1. Let's create the application with the Angular base structure using the @angular/cli with the route file and the SCSS style format.

ng new angular-recaptcha-v2

? Would you like to add Angular routing? Yes

? Which stylesheet format would you like to use? SCSS [ https://sass-lang.com/documentation/syntax#scss ]

CREATE angular-recaptcha-v2/README.md (1064 bytes)

CREATE angular-recaptcha-v2/.editorconfig (274 bytes)

CREATE angular-recaptcha-v2/.gitignore (604 bytes)

CREATE angular-recaptcha-v2/angular.json (3291 bytes)

CREATE angular-recaptcha-v2/package.json (1082 bytes)

CREATE angular-recaptcha-v2/tsconfig.json (783 bytes)

CREATE angular-recaptcha-v2/.browserslistrc (703 bytes)

CREATE angular-recaptcha-v2/karma.conf.js (1437 bytes)

CREATE angular-recaptcha-v2/tsconfig.app.json (287 bytes)

CREATE angular-recaptcha-v2/tsconfig.spec.json (333 bytes)

CREATE angular-recaptcha-v2/src/favicon.ico (948 bytes)

CREATE angular-recaptcha-v2/src/index.html (304 bytes)

CREATE angular-recaptcha-v2/src/main.ts (372 bytes)

CREATE angular-recaptcha-v2/src/polyfills.ts (2820 bytes)

CREATE angular-recaptcha-v2/src/styles.scss (80 bytes)

CREATE angular-recaptcha-v2/src/test.ts (788 bytes)

CREATE angular-recaptcha-v2/src/assets/.gitkeep (0 bytes)

CREATE angular-recaptcha-v2/src/environments/environment.prod.ts (51 bytes)

CREATE angular-recaptcha-v2/src/environments/environment.ts (658 bytes)

CREATE angular-recaptcha-v2/src/app/app-routing.module.ts (245 bytes)

CREATE angular-recaptcha-v2/src/app/app.module.ts (393 bytes)

CREATE angular-recaptcha-v2/src/app/app.component.scss (0 bytes)

CREATE angular-recaptcha-v2/src/app/app.component.html (24617 bytes)

CREATE angular-recaptcha-v2/src/app/app.component.spec.ts (1115 bytes)

CREATE angular-recaptcha-v2/src/app/app.component.ts (225 bytes)

✔ Packages installed successfully.

* + - 1. Install and configure the Bootstrap CSS framework. Do steps 2 and 3 of the post Adding the Bootstrap CSS framework to an Angular application.

3. Configure the siteKey variable with the Google reCAPTCHA key in the src/environments/environment.ts and src/environments/environment.prod.ts files as below.

recaptcha: {

siteKey: '6LfKNi0cAAAAACeYwFRY9\_d\_qjGhpiwYUo5gNW5-',

},

1. Install the ng-recaptcha library.

npm install ng-recaptcha

1. Import the FormsModule, RecaptchaFormsModule, RecaptchaModule modules. Configure the Google reCAPTCHA key. Change the app.module.ts file and add the lines as below.

import { FormsModule } from '@angular/forms';

import { RECAPTCHA\_SETTINGS, RecaptchaFormsModule, RecaptchaModule, RecaptchaSettings } from 'ng-recaptcha';

import { environment } from '../environments/environment';

imports: [

BrowserModule,

FormsModule,

AppRoutingModule,

RecaptchaModule,

RecaptchaFormsModule,

],

providers: [

{

provide: RECAPTCHA\_SETTINGS,

useValue: {

siteKey: environment.recaptcha.siteKey,

} as RecaptchaSettings,

},

],

1. Remove the contents of the AppComponent class from the src/app/app.component.ts file. Import the NgForm component and create the send method as below.

import { Component } from '@angular/core';

import { NgForm } from '@angular/forms';

@Component({

selector: 'app-root',

templateUrl: './app.component.html',

styleUrls: ['./app.component.scss'],

})

export class AppComponent {

token: string|undefined;

constructor() {

this.token = undefined;

}

public send(form: NgForm): void {

if (form.invalid) {

for (const control of Object.keys(form.controls)) {

form.controls[control].markAsTouched();

}

return;

}

console.debug(`Token [${this.token}] generated`);

}

}

1. Remove the contents of the src/app/app.component.html file. Add the re-captcha component as below.

<div class="container-fluid py-3">

<h1>Angular reCAPTCHA v2</h1>

<form #form="ngForm">

<div class="row mt-3">

<div class="col-sm-12 mb-2">

<re-captcha id="recaptcha" name="recaptcha" #recaptcha="ngModel" [(ngModel)]="token" required [class.is-invalid]="recaptcha.invalid && (recaptcha.dirty || recaptcha.touched)"></re-captcha>

<div \*ngIf="recaptcha.invalid && (recaptcha.dirty || recaptcha.touched)" class="invalid-feedback">

<div \*ngIf="recaptcha.errors?.['required']">This field is required.</div>

</div>

</div>

</div>

<div class="row">

<div class="col-sm-12 mb-2">

<button type="button" class="btn btn-sm btn-primary" (click)="send(form)">Send</button>

</div>

</div>

</form>

</div>

1. Add the style in the src/app/app.component.scss file as below.

re-captcha.is-invalid > div {

border: 1px solid #dc3545 !important;

border-radius: 0.2rem;

}

1. Run the application with the command below.

npm start

> angular-recaptcha-v2@0.0.0 start

> ng serve

✔ Browser application bundle generation complete.

Initial Chunk Files | Names | Size

vendor.js | vendor | 2.73 MB

styles.css | styles | 266.58 kB

polyfills.js | polyfills | 128.52 kB

scripts.js | scripts | 76.67 kB

main.js | main | 15.54 kB

runtime.js | runtime | 6.64 kB

| Initial Total | 3.21 MB

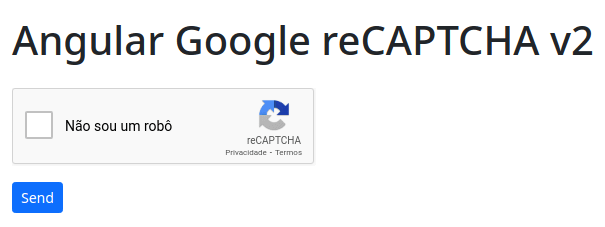
Build at: 2021-08-28T12:35:07.166Z - Hash: 0612b9d911a0acdf2b42 - Time: 10102ms

\*\* Angular Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200/ \*\*

✔ Compiled successfully.

1. Ready! Access the URL http://localhost:4200/ and check if the application is working. See the application working on GitHub Pages and Stackblitz.

**Output/Results snippet:**



**References:**

* <https://dev.to/rodrigokamada/adding-the-google-recaptcha-v2-to-an-angular-application-1o7o>
* <https://github.com/rodrigokamada/angular-recaptcha-v2>.

# 

# Activity 2

## Aim: Create an Angular application which can create Captcha.

**Learning outcome:** Able to develop the real time scenarios based on Node JS applications.

**Duration:** 5 hour

**List of Hardware/Software requirements:**

* Git
* Node.js and npm
* Angular CLI
* IDE (e.g. Visual Studio Code)

**Code/Program/Procedure (with comments)**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport"

content="width=device-width,

initial-scale=1">

<style>

\* {

box-sizing: border-box;

}

/\* Style the top navigation bar \*/

.topnav {

overflow: hidden;

background-color: #32CD32;

}

.topnav a {

float: left;

display: block;

color: #228B22;

text-align: center;

padding: 14px 16px;

text-decoration: none;

}

/\* Change color on hover \*/

.topnav a:hover {

background-color: #C0C0C0;

color: black;

}

@media screen and (max-width: 800px) {

.topnav a {

float: none;

width: 100%;

}

}

</style>

</head>

<body>

<h2>Responsive navigation </h2>

<p>Resize the browser window to see the effect:

When the screen is less than 800px,

the navigation will be displayed vertically

rather than of horizontally.</p>

<div class="topnav">

<a href="#">Link1</a>

<a href="#">Link2</a>

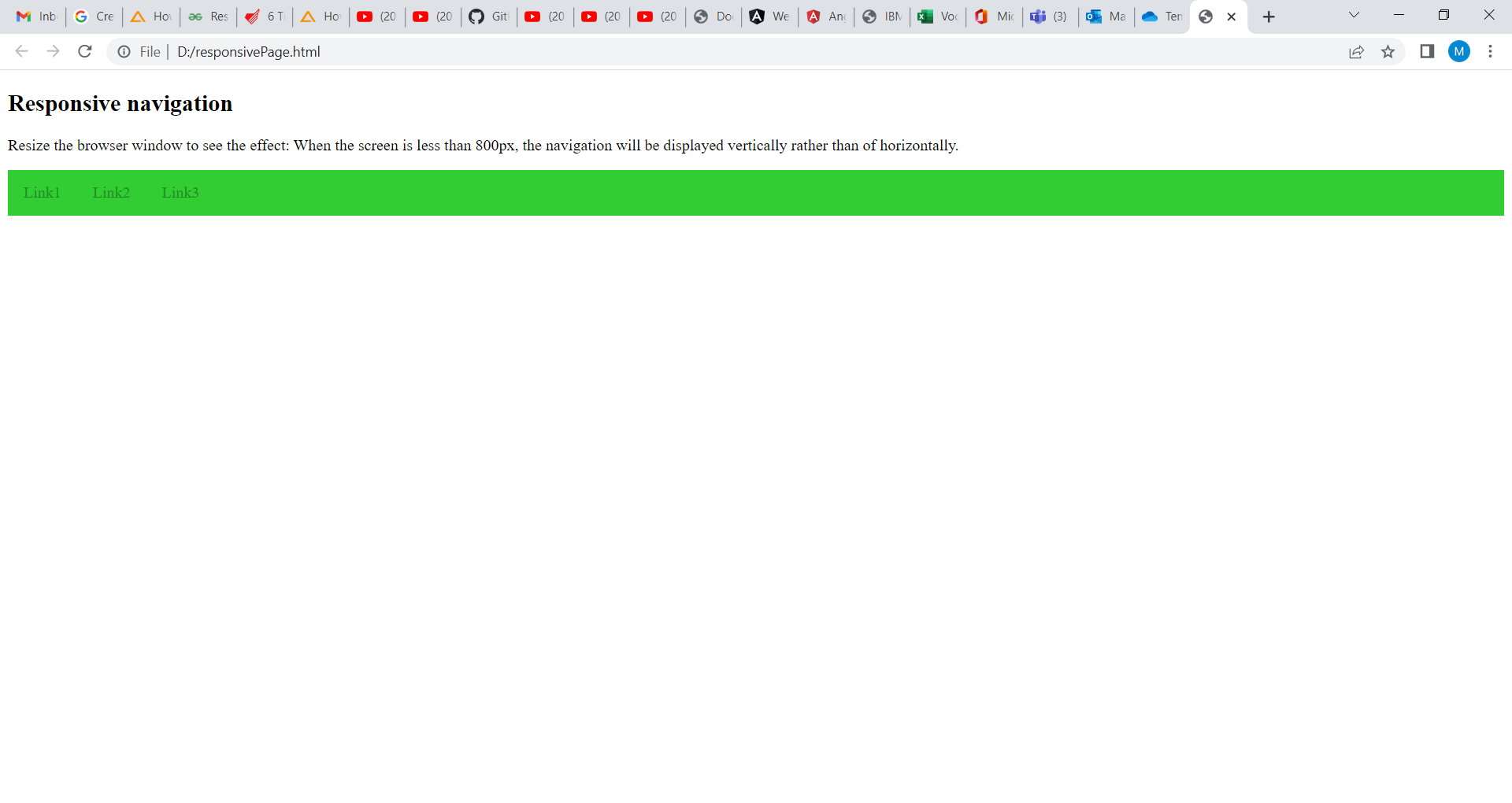
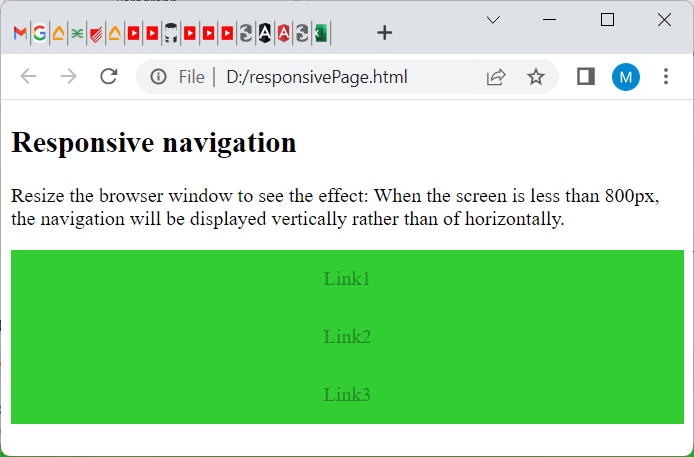
<a href="#">Link3</a>

</div>

</body>

</html>

**Output/Results snippet:**



**References:**

* <https://www.geeksforgeeks.org/responsive-page-in-angularjs/>