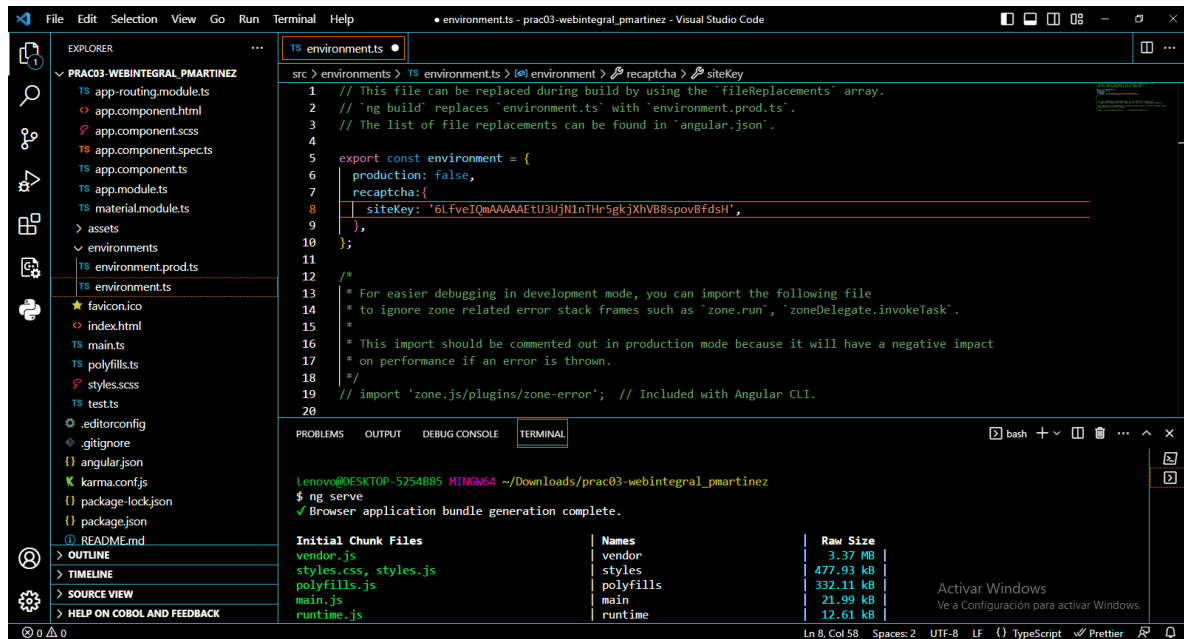


Pedro Emmanuel Martínez Rodríguez

GDGS3091

U3 prac02_Recaptcha Implementación del Captcha

Creación del enviromen para agregar el site key del recaptcha

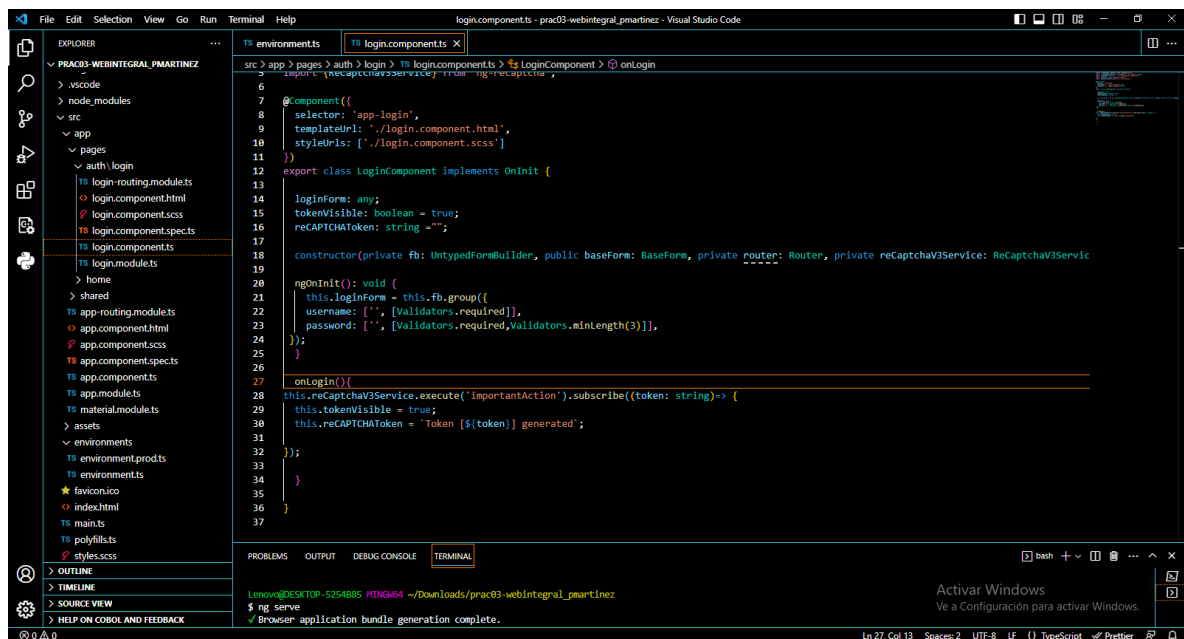


The screenshot shows the Visual Studio Code interface. The Explorer panel on the left displays the file structure of the 'PRAC03-WEBINTEGRAL_PMARTINEZ' project, with 'environment.ts' selected under the 'environments' folder. The main editor area shows the content of 'environment.ts', which is a TypeScript file for configuring the Angular environment. It includes comments about file replacements and a configuration object for the environment. The terminal window at the bottom shows the command 'ng serve' being executed, and the output indicates that the browser application bundle generation is complete. A table of initial chunk files is also displayed in the terminal output.

```
src > environments > TS environment.ts > @environment > @recaptcha > @siteKey
1 // This file can be replaced during build by using the `fileReplacements` array.
2 // `ng build` replaces `environment.ts` with `environment.prod.ts`.
3 // The list of file replacements can be found in `angular.json`.
4
5 export const environment = {
6   production: false,
7   recaptcha: {
8     siteKey: '6LfveIQmAAAAEttU3UjNtHr5gkjKhV88spovBfdsh',
9   },
10 };
11
12 /*
13  * For easier debugging in development mode, you can import the following file
14  * to ignore zone related error frames such as `zone.run`, `zoneDelegate.invokeTask`.
15  *
16  * This import should be commented out in production mode because it will have a negative impact
17  * on performance if an error is thrown.
18  */
19 // import 'zone.js/plugins/zone-error'; // Included with Angular CLI.
20
```

Initial Chunk Files	Names	Raw Size
vendor.js	vendor	3.37 MB
styles.css, styles.js	styles	477.92 kB
polyfills.js	polyfills	332.11 kB
main.js	main	21.99 kB
runtime.js	runtime	12.61 kB

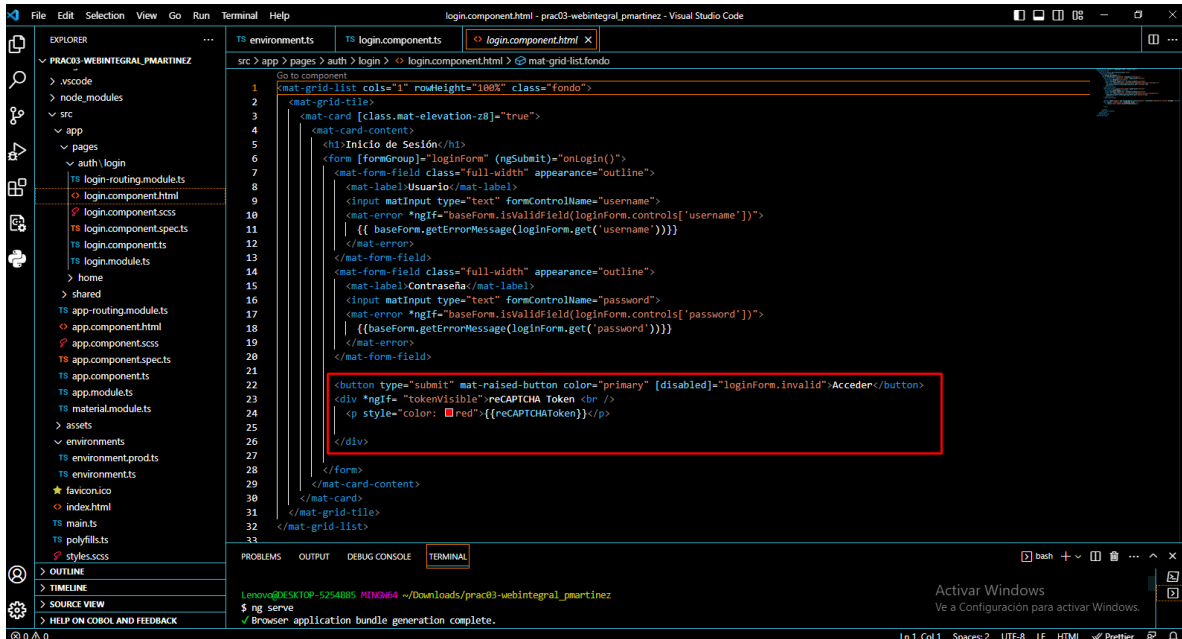
Se agrega la validación del recaptcha



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left displays the file structure of the 'PRAC03-WEBINTEGRAL_PMARTINEZ' project, with 'login.component.ts' selected under the 'src > app > pages > auth > login' folder. The main editor area shows the content of 'login.component.ts', which is a TypeScript file for the login component. It includes a selector, templateUrl, styleUrls, and a class that implements OnInit. The class has a constructor and an ngOnInit method. The terminal window at the bottom shows the command 'ng serve' being executed, and the output indicates that the browser application bundle generation is complete.

```
src > app > pages > auth > login > TS login.component.ts > @LoginComponent > @onLogin
6
7 @component({
8   selector: 'app-login',
9   templateUrl: './login.component.html',
10  styleUrls: ['./login.component.scss']
11 })
12 export class LoginComponent implements OnInit {
13
14   loginForm: any;
15   tokenVisible: boolean = true;
16   reCAPCHAToken: string = "";
17
18   constructor(private fb: UntypedFormBuilder, public baseForm: BaseForm, private router: Router, private reCaptchaV3Service: ReCaptchaV3Service) {}
19
20   ngOnInit(): void {
21     this.loginForm = this.fb.group({
22       username: ['', [Validators.required]],
23       password: ['', [Validators.required, Validators.minLength(3)]],
24     });
25   }
26
27   onLogin() {
28     this.reCaptchaV3Service.execute('importantAction').subscribe((token: string) => {
29       this.tokenVisible = true;
30       this.reCAPCHAToken = `Token ${token} generated`;
31     });
32   }
33 }
34
35
36
37
```

Se agrega el recaptcha al login



```
1 <mat-grid-list cols="1" rowHeight="100%" class="fondo">
2   <mat-grid-tile>
3     <mat-card [class.mat-elevation-z8]="true">
4       <mat-card-content>
5         <h1>Inicio de Sesión</h1>
6         <form [formGroup]="loginForm" (ngSubmit)="onLogin()">
7           <mat-form-field class="full-width" appearance="outline">
8             <mat-label>Usuario</mat-label>
9             <input matInput type="text" formControlName="username">
10            <mat-error *ngIf="baseForm.isValidField(loginForm.controls['username'])">
11              {{ baseForm.getErrorMessage(loginForm.get('username'))}}
12            </mat-error>
13          </mat-form-field>
14          <mat-form-field class="full-width" appearance="outline">
15            <mat-label>Contraseña</mat-label>
16            <input matInput type="text" formControlName="password">
17            <mat-error *ngIf="baseForm.isValidField(loginForm.controls['password'])">
18              {{baseForm.getErrorMessage(loginForm.get('password'))}}
19            </mat-error>
20          </mat-form-field>
21          <button type="submit" mat-raised-button color="primary" [disabled]="loginForm.invalid">Acceder</button>
22          <div *ngIf="tokenVisible">recaptcha Token <br />
23            <p style="color: red">{{recaptchaToken}}</p>
24          </div>
25        </form>
26      </mat-card-content>
27    </mat-card>
28  </mat-grid-tile>
29 </mat-grid-list>
```

Demostración de la funcionalidad del token generado

