

Global error handler

Transcrição

```
import { ErrorHandler, Injectable } from '@angular/core';
import * as StackTrace from 'stacktrace-js';
import { LocationStrategy } from '@angular/common';
import { PathLocationStrategy } from '@angular/common';
import { UserService } from '../core/user/user.service';
import { HttpClient } from '../node_modules/@angular/common/http';
import { ServerLogService } from './server-log-service';

@Injectable({ providedIn: 'root' })
export class GlobalErrorHandler implements ErrorHandler {

  private userName: string;

  constructor(
    private locationStrategy: LocationStrategy,
    private userService: UserService,
    private serverLogService: ServerLogService
  ) {
    this.userService.getUser().subscribe(user => {
      if(user) return this.userName = user.name;
      this.userName = null;
    });
  }

  handleError(error) {
    console.log(error);
    const userName = this.userName;
    const message = error.message ? error.message : error.toString();
    const url = location instanceof PathLocationStrategy
      ? this.locationStrategy.path() : this.locationStrategy.path(false);
    const log = { userName, url, message, stack: error.stack };
    this.serverLogService.log(log)
      .subscribe(() => {
        console.log('Error logged by the server.')
      }, err => {
        console.log('Log service is offline');
      });
  }
}

export interface ServerLog {

  userName: string;
  url: string;
  message: string;
  stack: string;
}
```

```
providers: [  
  {  
    provide: ErrorHandler,  
    useClass: GlobalErrorHandler  
  }  
]
```