



Universitatea “Ștefan cel Mare” Suceava  
Facultatea de Inginerie Electrică și Știința Calculatoarelor  
Domeniul în Inginerie Electronică, Telecomunicații și Tehnologii Informaționale

# **INGINERIE SOFTWARE AVANSATĂ**

## **AUTODOCUMENTAȚIE**

**Profesor coordonator :**

**a.s.dr.ing. Ovidiu GHERMAN**

**Masterand :**

**Alexandru TELEAGĂ**

**Grupa: RCC 31111 A**

**Suceava, 2018**

# CUPRINS

1. INTRODUCERE .....	1
1.1. Scopul proiectului. ....	1
1.2. JSDoc. ....	1
1.3. Instalare JSDoc 3.....	1
2. CREAREA AUTODOCUMENTAȚIEI .....	4
2.1. Necesitatea. ....	4
2.2. Realizarea automată a documentației JavaScript. ....	4
3. CONCLUZII.....	6
4. BIBLIOGRAFIE .....	7

# 1. INTRODUCERE

## *1.1. Scopul proiectului.*

Scopul proiectului este de a genera în mod automat documentația codului Java Script utilizat în crearea site-ului web. În acest scop, am folosit un generator de documentație API pentru JavaScript, numit JSDoc.

## *1.2. JSDoc.*

JSDoc 3 este un generator de documentație API pentru JavaScript, similar cu Javadoc sau phpDocumentor, unde putem adăuga comentarii de documentație direct la codul sursă, chiar alături de codul însuși. Instrumentul JSDoc va scana codul sursă și va genera un site web cu autodocumentația în format HTML

## *1.3. Instalare JSDoc 3.*

Din păcate, instalarea JSDoc nu este la fel de simplă cum ar putea fi. Motivul pentru aceasta este că este scris în Perl, deci dacă nu este Perl deja instalat nu vom putea continua cu instalarea JSDoc 3. Deci, pentru început am instalat Active Perl [1].

Odată ce ActivePerl este instalat cu succes, următorul pas este să descărcăm și să decomprimăm JSDoc. În acest moment, este posibil să rulați JSDoc 3 din fereastra DOS, dar dacă o faceți, probabil că nu va funcționa. Această eroare se datorează unui pachet Perl lipsă.

Utilizând Managerul de pachete Perl, poate fi instalat pachetul HTML-Template lipsă, prin tastarea în consolă “ppm”, după cum reiese din fig.1.

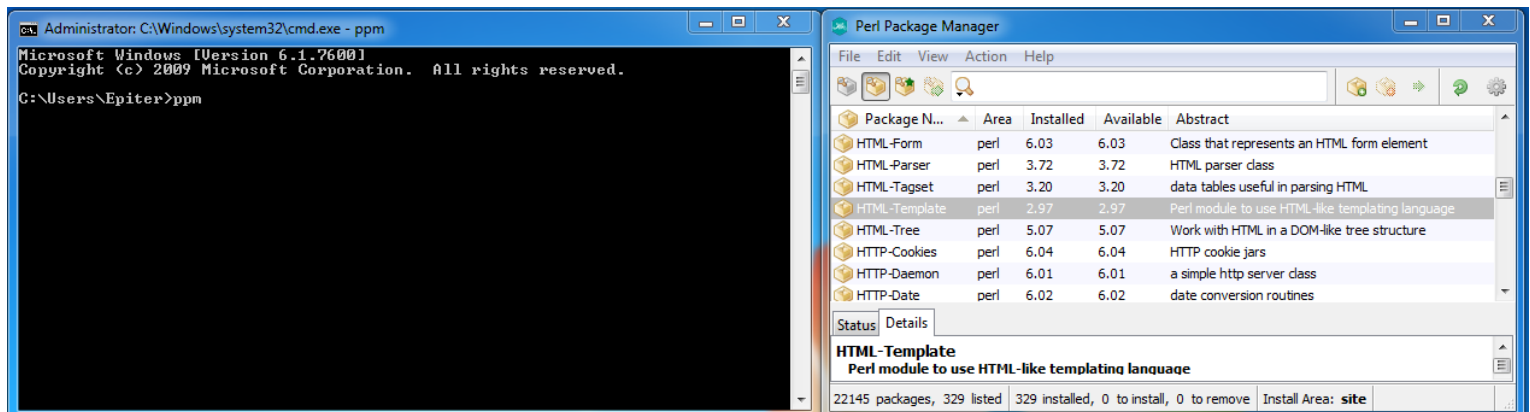


Fig.1. Managerul de pachete Perl

Selectăm HTML – Template și instalăm pachetul corespunzător ca în fig.2.

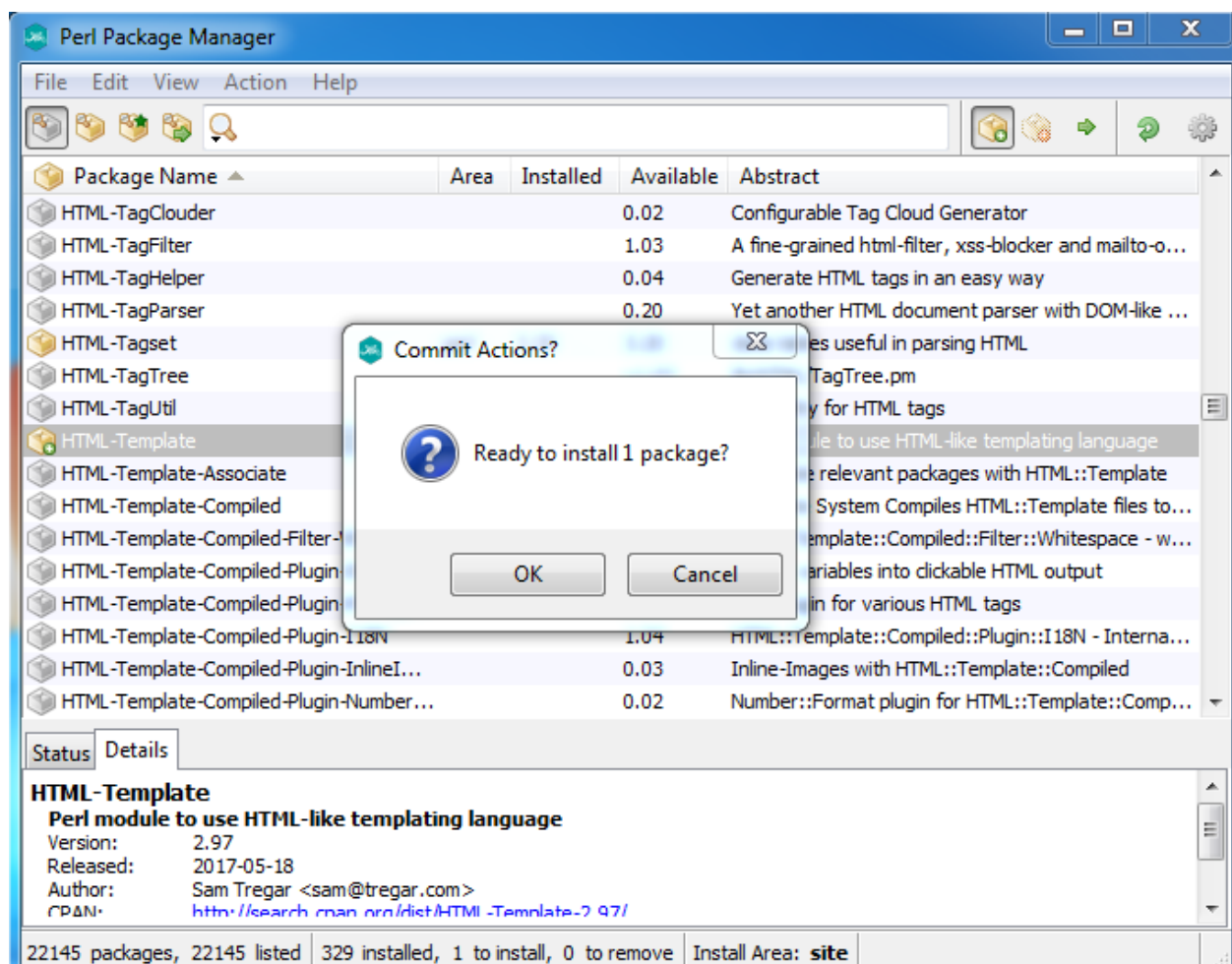


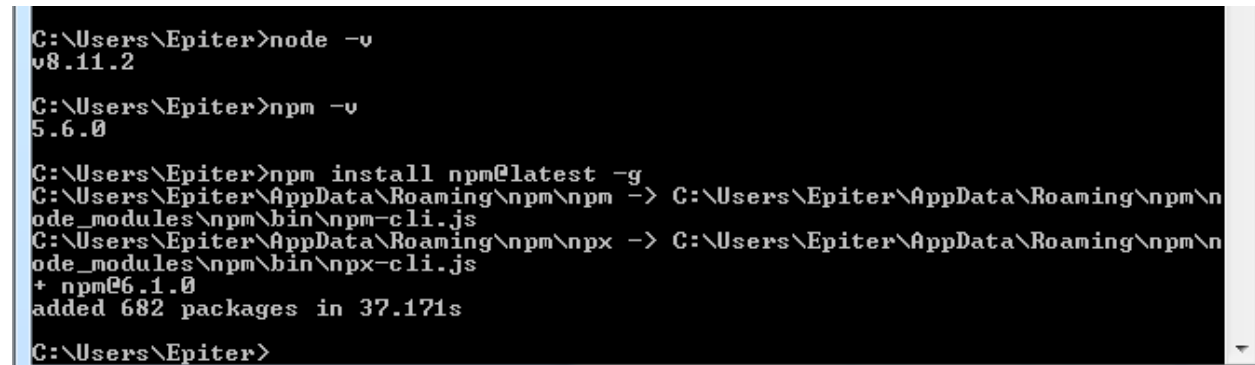
Fig.2. Install HTML – Template

În continuare am avut nevoie de NodeJS, deci l-am instalat și pe acesta [2], apoi npn [3].

Odată instalate, vom scrie în linia de comandă:

```
npm install npm@latest -g
```

de unde opțin ultima versiune npn, fig.3.



```
C:\Users\Epiter>node -v
v8.11.2

C:\Users\Epiter>npm -v
5.6.0

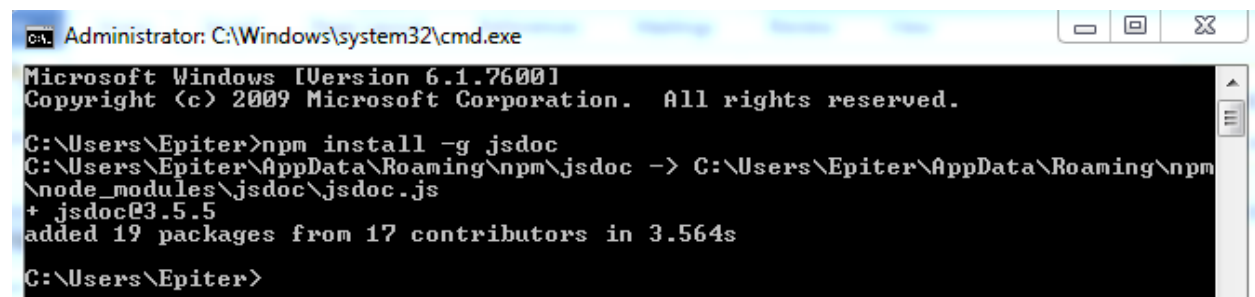
C:\Users\Epiter>npm install npm@latest -g
C:\Users\Epiter\AppData\Roaming\npm\npm -> C:\Users\Epiter\AppData\Roaming\npm\node_modules\npm\bin\npm-cli.js
C:\Users\Epiter\AppData\Roaming\npm\npx -> C:\Users\Epiter\AppData\Roaming\npm\node_modules\npm\bin\npx-cli.js
+ npm@6.1.0
added 682 packages in 37.171s

C:\Users\Epiter>
```

*Fig.3.* Ultima versiune npn

Acum că totul este pregătit, putem foarte ușor să instalăm JSDoc folosind linia de comandă:

```
npm install -g jsdoc
```



```
Administrator: C:\Windows\system32\cmd.exe

Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Epiter>npm install -g jsdoc
C:\Users\Epiter\AppData\Roaming\npm\jsdoc -> C:\Users\Epiter\AppData\Roaming\npm\node_modules\jsdoc\jsdoc.js
+ jsdoc@3.5.5
added 19 packages from 17 contributors in 3.564s

C:\Users\Epiter>
```

*Fig.4.* Instalarea JSDoc prin linia de comandă

## 2. CREAREA AUTODOCUMENTAȚIEI

### 2.1. Necesitatea.

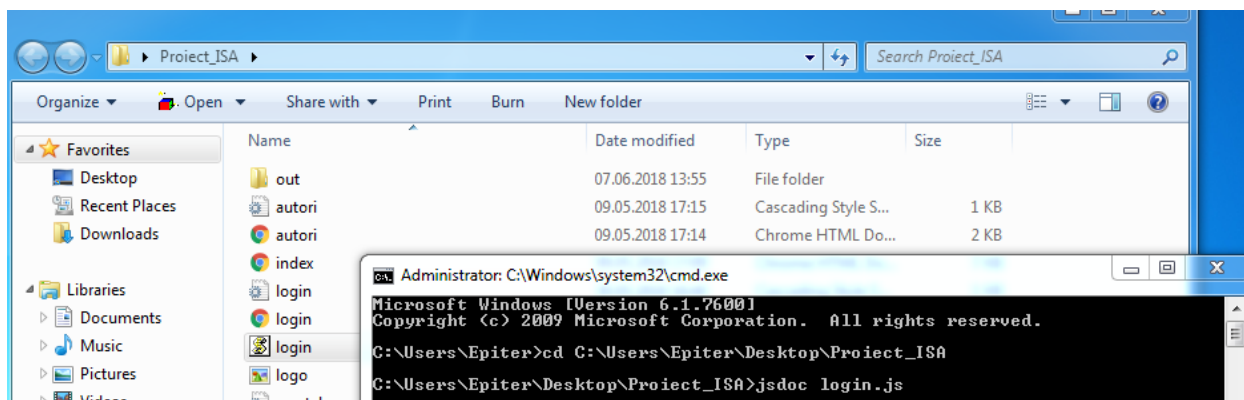
Oricare ar fi motivul, nedocumentarea unei aplicații nu este niciodată un lucru bun, chiar dacă este, de obicei, aceasta este complexă. Acest lucru este valabil, mai ales atunci când se dă codul sursa JavaScript clientului, care este, de obicei, neinițiat în programare. Interesant, totuși, răspunsul la problema cu documentația JavaScript provine de la una din aceleași tehnologii server-side – Java, de pildă JSDoc 3.

### 2.2. Realizarea automată a documentației JavaScript.

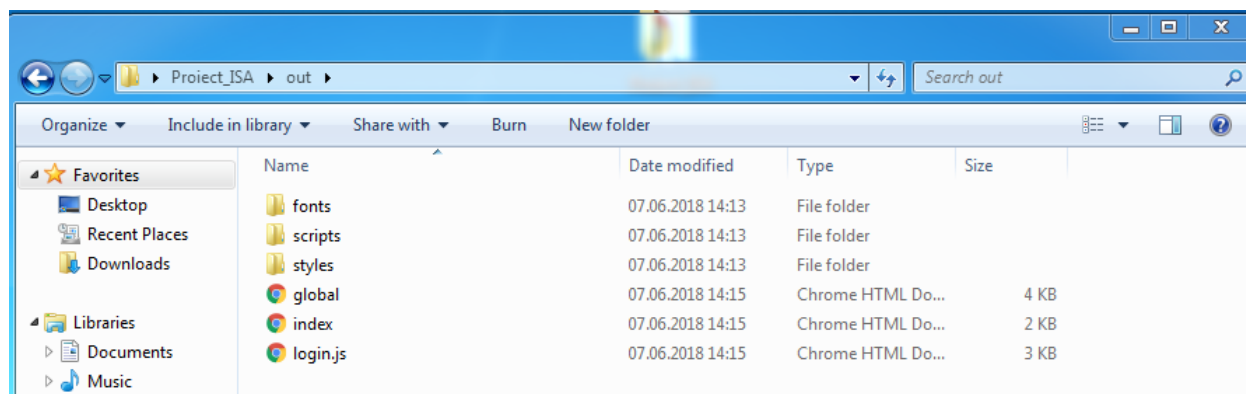
Odată ce avem fișierul .js, iar JSDoc 3 este instalat și actualizat, putem să trecem la ultimul pas al oricărui proiect, și anume autodocumentația acestuia.

Prima dată deschidem linia de comandă și ne mutăm în folderul în care avem fișierul js, după care introducem, în linia de comandă de mai jos și obținem autodocumentația, fig.5:

```
jsdoc numefisier.js
```



*Fig.5. Comanda JSDoc pentru crearea unui folder cu fișiere generate automat*



*Fig.6. Fișierul generat de JSDoc cu pagina de documnetate automată HTML*

### 3. CONCLUZII

În urma acestei lucrări am înțeles cât de importantă poate fi autodocumentarea unui proiect. Acest lucru ajută enorm în cadrul unei echipe pentru a înțelege rapid codul scris. Totodată, clientul poate înțelege proiectul doar deschizând o pagină HTML generată automat de către JSDoc 3, iar în momentul în care programatorul vrea să revină la o funcție ce a fost scrisă cu mult timp în urmă, acesta poate revizui documentația exact unde are nevoie, fără să piardă timp prețios căutând funcția ca apoi să ghicească fiecare parametru ce rol are și ce realizează.

În concluzie, autodocumentația este un pas foarte important pentru toți cei implicați într-un proiect, un pas ce este foarte ușor de făcut, cu efecte negative majore în cazul în care lipsește.



## 4. BIBLIOGRAFIE

- [1] <https://www.activestate.com/activeperl/downloads>
- [2] <https://nodejs.org/en/>
- [3] <https://www.npmjs.com/get-npm>