* **HTTP**

**-Prezentare generala, (request, response, alte protocoale de comunicare, etc)**

**POST, GET, DELETE, etc ….**

* **Tehnologii folosite**

**HTML, CSS(Bootstrap Library), jQuery Javascript Library( Client-Side), PHP(Server-Side), summernote(WYSIWYG Editor ), baze de date relationale MySQL**

**-Client-side, Server-side**

**Prezentare generala, etc**

**-HTML**

Prezentare generala

HMTL5 -> S a folosit in proiect metode asociate HTML5: localStorage, enctype

**-CSS**

Prezentare generala

Bootstrap: Descriere, utilizarea bootstrapului in proiect(S a folosit grid-ul si clase de bootstrap)

CSS este separat in folderul /css si este inclus in header.php(header.php la randul lui este inclus in fiecare pagina)

Descriere Reguli CSS folosite in proiect(le gasesti in /css/styles.css)

**-AJAX**

Prezentare generala, tipul de date JSON

**-Javascript**

Prezentare generala

Javascriptul se gaseste in /js/scripts.js

**-jQuery Javascript library**

Ce este o librarie, prezentare generala, descrierea utilizarii jQuery in proiect,

metodele folosite in proiect:

**-PHP**

Prezentare generala, descrierea ultilizarii PHP-ului in proiect, metode folosite in proiect.

Programare Obiect Orientata si Programare Procedurala

Metodologie software: Separarea logicii de prezentare

**Metode PHP din proiect**:

require() si include(): Descriere, Diferente

mail() : Functia de mail inclusa in PHP, descriere

lucru cu fisiere: Metode PHP pentru lucru cu fisiere(file system)

Sesiuni si Cookie-uri: Descriere, Diferente

Functii PHP pentru baze de date: mysqli\_query(), mysqli\_fetch\_assoc()

Functii PHP de securitate : msqli\_real\_escape\_string(), htmlspecialcharacters(), get\_magic\_quotes\_gpc()

Functii generale PHP din proiect: trim(), stripslashes(), isset(), perechea json\_encode(), json\_decode(), etc

**-Summernote**

Prezentare generala, de ce l am folosit in proiect: API simplu deci usor de integrat in proiect,

speed up development speed, WYSIWYG

**-MySQL**

Prezentare generala, designul bazei de date, tipul de date, normalizare, foreign keys, etc

GENERAL DESPRE PROIECT

-S-a separat conexiunea de baza de date de restul aplicatiei, ea fiind inclusa doar cand este necesara in pagina( O gasesti in /config/ db-connect.cfg.php). O incluzi in pagini prin require(si nu include) “config/db-connect.cfg.php”

-HTMLul este spart in header, footer, si body ul paginii. In fiecare body de pagina este inclus headerul si footerul: include “includes/header.inc.php” si include “/includes/footer.inc.php”(include si nu require, descris de ce)

-CSS este prezent in /css folder si este inclus in header, iar headerul inclus in fiecare pagina.

-Javascriptul il gasesti in /js/scripts.js si este inclus in footer.php, footerul este inclus in fiecare pagina.

Aparent mai gasesti si un script micut in header.inc.php(resposabil pt functionalitatea de facebook like)

-Footer-ul are cod pentru inclus dependetele proiectul( atat local cat si de pe CDN(Ce este, avantaje , etc))

-Orice din proiect este spart in bucatele in functie de functionalitate(delete.php, vote.php, index.php, process.php)

Alternativa ar fi sa ai **doar un** fisier cu 9821347983214746 linii de cod.

Asta e un standard despre care n ar trebui sa mai precizez.

Mentiune: Trebuie treacut prin cod si sa dea search pe google(Se gaseste orice pe google:

WHAT IS CUTARE, WHAT IS CUTARE, WHAT THE FUCK IS CUTARE), daca sunt nelamuriri, explic fara nicio problema)

De fapt, orice iti trece prin cap se gaseste pe google.

**Exemplu**

linkuri care explica de s-a ales tehnologiile:

<http://www.sitepoint.com/why-choose-php/> -pt php

<https://www.ostraining.com/blog/coding/bootstrap/> -pt bootstrap

<http://www.javaworld.com/article/2078613/java-web-development/6-reasons-you-should-be-using-jquery.html> -pt jquery

<http://www.weblinkindia.net/blog/why-choose-javascript> - pt javascript(Well, Javascript pt web este omniprezent, ca si Dumnezeu, lolz)

<https://www.mysql.com/why-mysql/topreasons.html> - pt mysql