

Tema 1

Proiectați și implementați o interfață cu utilizatorul folosind pagini web pentru o aplicație de evaluare și clasificare a medicilor și a spitalelor din România. Operații permise de aplicație:

1. Afișare informații despre unități medicale (nume, tip unitate, locație, sigla) **3p**
2. Buton/link către pagina detalii despre medicii ce activează într-o unitate medicală (nume, specializare, an absolvire, poza profil) **3p**
3. Selectare Limba (RO și EN). Salvarea preferinței utilizatorilor pentru limba în care se afișează informația se va face folosind HTTP Cookies sau HTML 5 Web Storage **3p**

Constrângeri:

- Pentru implementare folosiți limbajele HTML 5, CSS și JavaScript. Se accepta utilizarea oricărui framework existent care pune la dispoziție librării avansate pentru aceste limbaje (**Sugestii: Vue.js, Angular, React sau jQuery**).
- **Persistența datelor nu este în obiectivul temei.** Datele dummy despre spitale și medici se vor încărca folosind fișiere sau vor fi înglobate în paginile web dezvoltate (**Sugestie: fișiere XML și procesarea acestora**).
- La evaluare se vor considera următoarele aspecte legate de aplicație: meniuri, utilizare butoane pentru navigare, design pagina, paginare, font-uri adecvate, imagini, etc.

Tema 2

Plecând de la aplicația web dezvoltată la Tema 1, implementați următoarele noi funcționalități:

1. Încărcare date despre spitale și medici dintr-o bază de date **2p**
2. Înregistrare / logare utilizator (email & parola) **2p**
3. Pentru utilizatorii logați:
 - 2.1. Pagina de profil utilizator care conține următoarele informații ce pot fi modificate: Nume, Email, Parola, Afecțiune (selectare mai multe valori din listă), Imagine profil (posibilitate de încărcare imagine) **2.5p**
 - 2.2. Adăugarea de comentarii și evaluarea medicilor **2.5p**

Constrângeri:

- Se vor folosi tehnologii/limbaje AJAX și PHP pentru a implementa noile funcționalități (**Sugestie: folosiți servicii REST**). Se accepta utilizarea oricărui Framework existent care pune la dispoziție librării avansate pentru aceste tehnologii (**Sugestie: Laravel**).
- Persistența datelor se face folosind baze de date (**Sugestie: MySQL**).
- Pentru păstrarea stării utilizatorului după logare se vor folosi sesiuni.

Tema 3

Plecând de la aplicația web dezvoltată la Tema 2, implementați următoarele noi funcționalități:

1. Pagina acces administrator bazată pe autentificare **2p**
2. Administrarea informațiilor despre spitale și medici (operații CRUD). **3p**
3. Ștergerea contului unui utilizator anume **2p**
4. Ștergerea sau ascunderea conținutului pentru anumite comentarii **2p**

Constrângeri:

- Folosiți Python sau Ruby (alegeți cel puțin un limbaj) pentru implementare (**Sugestie: folosiți servicii REST**). Se accepta utilizarea oricărui Framework existent care pune la dispoziție librării avansate pentru aceste limbaje (**Sugestie: Ruby on Rails sau Django**).

Tema 4

Plecând de la aplicația web dezvoltată la Tema 3 considerați următoarele noi cerințe:

- Implementați și integrați în aplicație următoarele noi servicii
 - calculul ratingului unui spital bazat pe evaluarea medicilor făcută de către utilizatori și afișarea acestui rating pentru fiecare spital în parte ca și raport/grafic (**Sugestie: integrați un serviciu web pentru desenare de grafice**). **3p**
 - afișare spitale din vecinătatea utilizatorului și identificarea lor pe hartă (**Sugestie: integrați un serviciu web pentru hărți**). **3p**
- Definiți cazuri de testare pentru noile funcționalități. Folosiți unelte și tehnici de testare pentru a le valida. **3p**

Constrângeri:

- Noile funcționalități vor fi implementate folosind orice limbaj de programare web. Se accepta utilizarea oricărui Framework web existent.