Fisa cerintelor

1.Tipuri de utilizatori:

* 1. Utilizator
* Trebuie sa fie autentificat, daca nu are un cont isi poate crea unul sau se poate loga cu un cont existent pe retelele sociale.
* Acesta isi poate vedea detaliile contului, sa raspunda la intrebari, sa adauge propriile intrebari impreuna cu solutia si sa evalueze raspunsurile altor utilizatori la intrebari daugate. De asemenea poate alege din lista de livestream-uri unul pe care il va urmari, va adauga comentarii si va putea da feedback profesorului.
* Un utilizator poate crea livestreamuri daca i s-a dat dreptul de catre administrator.
  1. Administrator
* Are drepturi depline asupra aplicatiei. Se ocupa de mentenanta site-ului.
* Acesta acorda dreptul de adaugare de livestreamuri in functie de anumite criterii.
* Poate sterge intrebarile cu un raiting mic si lua dreptul de acrea livestream utiliztorilor cu feedback negativ.

2.Scenarii de utilizare:

* Un utilizator intra pe pagina de logare de unde alege sa se logheze, sa isi creeze un cont sau poate folosi un cont deja existent pe retelele sociale (Twitter/Github/Facebook).
* Odata logat acesta va ajunge pe pagina principala. Aici isi poate vedea datele in sectiunea „My account”.
* Poate alege sa raspunda la intrebari fiind directionat pe pagina de intrebari. Va putea alege in meniul din stanga o categorie de intrebari ce ii vor fi afisate. Apasand pe o intrebare va intra pe pagina dedicata acesteia in care va raspunde.
* O alta optiune din pagina principala este de a adauga un exercitiu. Acesta va trebui sa contina solutia, o explicatie succinta, o categorie si un nivel de dificultate.
* Utilizatorul poate vede in pagina principala lista de livestreamuri si sa aleaga unul pentru a-l vizualiza. In pagina de vizualizare poate adauga comentarii in timp real si sa adauge un feedback.

3.Arhitectura aplicatiei:

- Utilizatoii isi pot crea un cont in doua moduri, fie isi adauga datele in sectiunea Sign Up, acestea fiind adaugate in baza de date, parola va fi hashuit inainte da a fi stocata. A doua metoda este da a-si crea contul si a se loga folosind retele sociale via OAuth ( prin importarea librariilor)

- Livestreamul va fi realizat prin intermediul WebSocketurilor.

Pentru a persista utilizatorii logati vom utiliza un JSON Web Token. Header-ul va include tipul de token si algoritmul de criptare HMAC, in payload va fi usename-ul si emailul utilizatorului, iar pentru a crea signatura vom adauga secretul „ASSDF”. Acestia vor expira dupa 2 ore.

- Arhitectura aplicatiei are la baza design pattern-ul MVC:

Initial va fi rolul router-ului sa se asigure ca utilizatorul incearca sa aceceseze o pagina valida si sa il directioneze catre controllerul vizat.

Controllerele au rolul principal de directionare a datelor, ele vor face legatura intre partea de view sau browser si partea de logica a aplicatiei. Vor exista patru controllere utilizate pentru gestiunea uilizatorilor, intrebarilor, raspunsurilor si livestreamurilor. Question Controller se ocupa de gestiunea, afisarea, filtrarea intrebarilor in pagina; Answear Controller se ocupa de adaugarea raspunsurilor, evaluarea lor si afisarea; User Controller se ocupa cu autentificarea userilor, creearea tokenurilor, persitarea acestora; LiveStream Controller gestioneaza modul de creare a liveurilor, afisarea acestora, adaugare de comentarii si feedback ale utilizatorilor.

Modelele se ocupa de gestiunea datelor si de comunicarea directa cu baza de date. Fata de controllere acestea sunt 5 apare unul in plus pentru gestiunea comentariilor.Toate modelee: Question, Answer, User, LiveStream, Comment vor oferi functionalitati controllerelor specifice.

4.Repartizarea taskurilor:

Alexandru Denis:

- creare baza de date

- adaugare si gestionare intrebari

- adaugare raspuns

Bumbu Ana-Maria:

-logare/autentificare via OAuth

-persistare useri folosind JWT

-evaluare raspuns

Todireanu Valentin:

-creare livestreamuri

-vizualizare livestream

- gestionare comentarii