

Raport de Planificare și Implementare

Proiect “Agora”

Echipa: Panaet Maria Alexandra, Drăgușoi Miruna

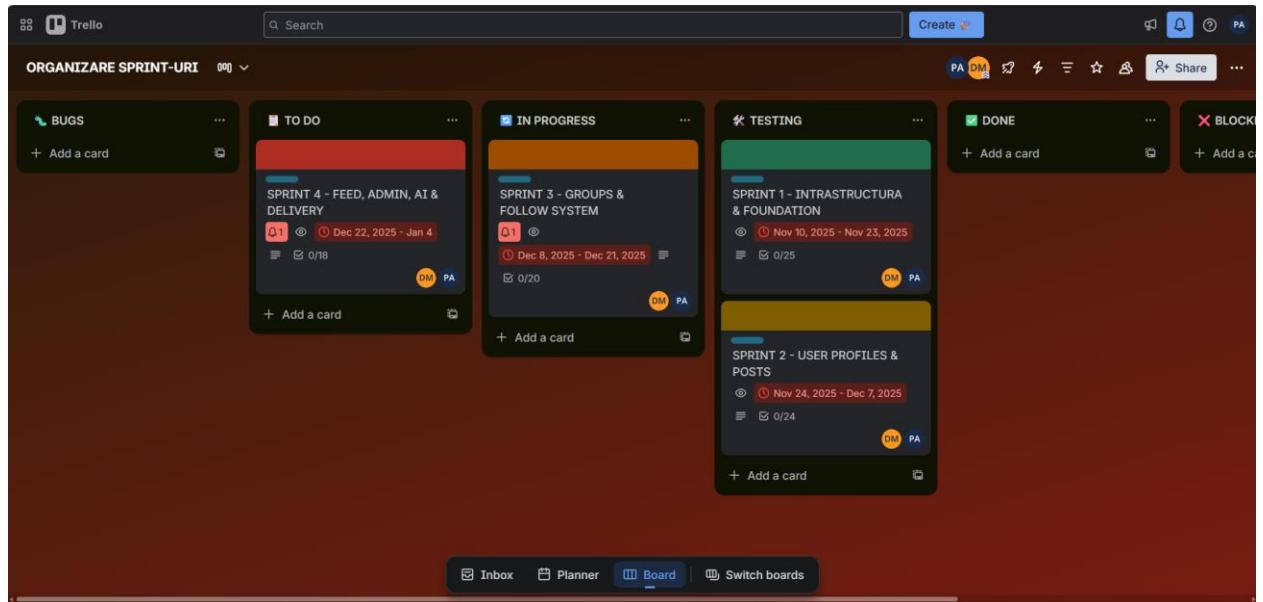
Grupa: 244

Proiect: “Micro-Social Platform”

1. Metodologia de Lucru și Organizarea

Pentru dezvoltarea platformei “Agora”, am adoptat o metodologie de lucru Agile, organizându-ne activitatea în Sprint-uri. Instrumentul principal pentru managementul task-urilor a fost Trello, care ne-a permis să avem o viziune clară asupra progresului. Am ales să lucrăm asincron, dar cu puncte de sincronizare frecvente pentru a evita suprapunerile de cod, în special pe fișierele sensibile precum Program.cs sau Controllerele principale.

Vedere de ansamblu a board-ului Trello cu listele (organizarea generală a Sprint-urilor):

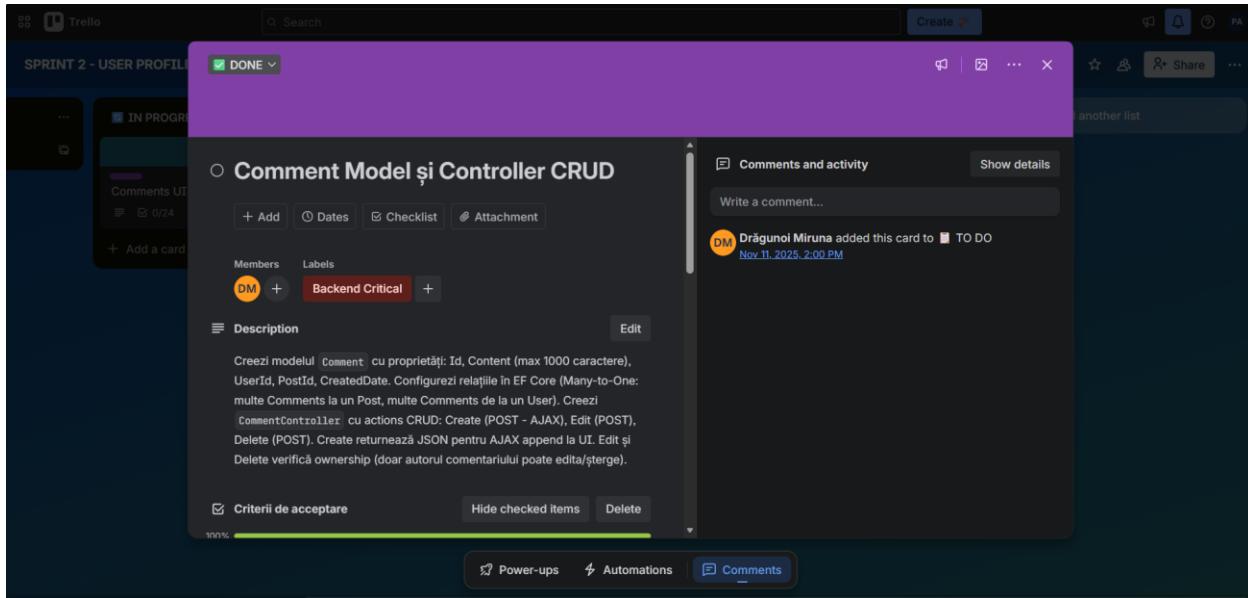


2. Planificarea și Împărțirea Task-urilor

Am realizat împărțirea task-urilor bazându-ne atât pe complexitate, cât și pe interdependențe.

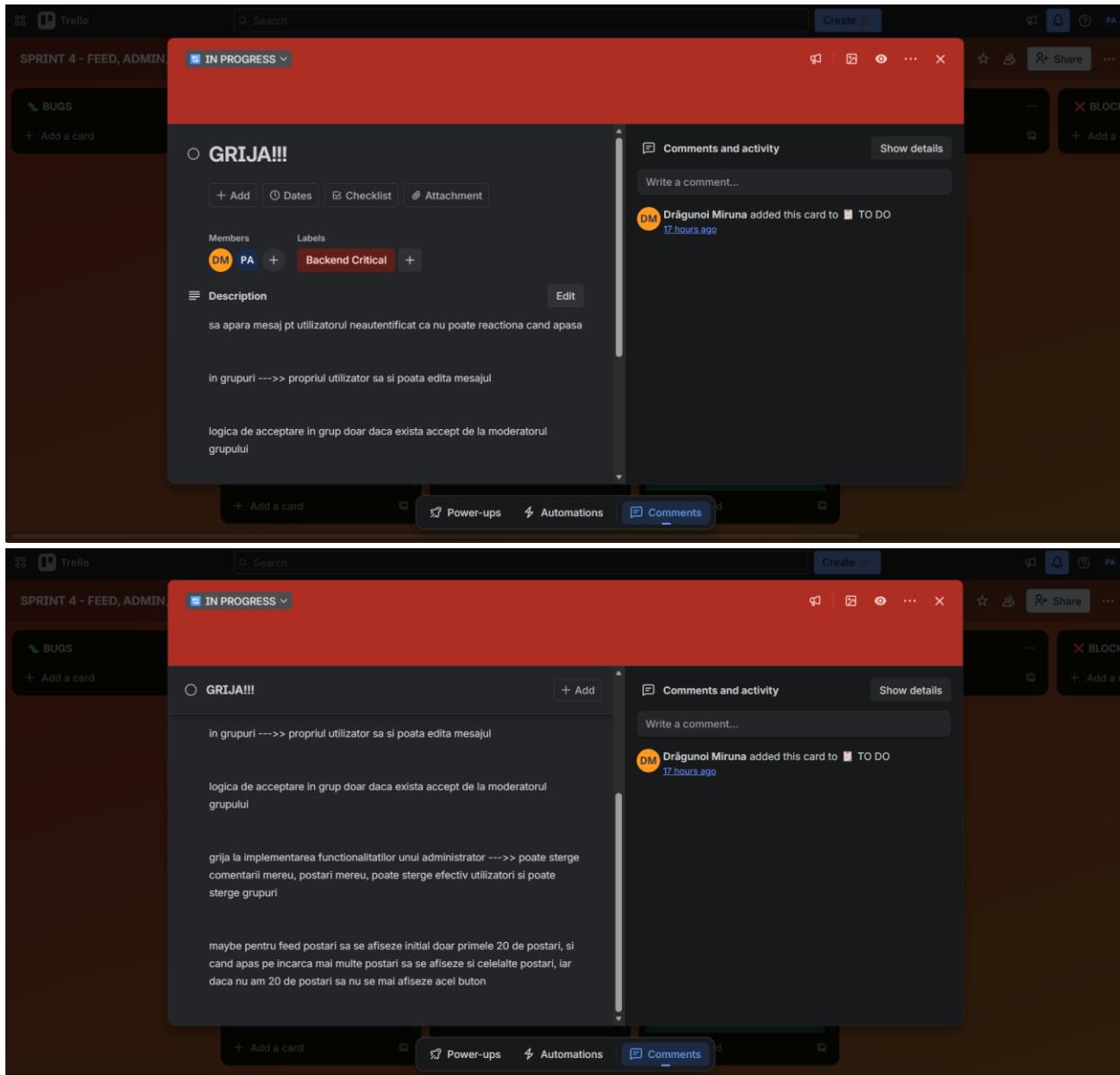
- Comunicarea:** Am folosit Trello nu doar pentru status, ci și pentru comunicare critică. Spre exemplu, am folosit etichete precum “Backend Critical” pentru a semnala funcționalități ce blochează alte dezvoltări.

Eticheta “Backend Critical”:



- **Sincronizarea în timp real:** Deși Trello a oferit o viziune de ansamblu, am menținut o comunicare continuă (prin mesaje instantanee și discuții directe) pentru a gestiona fluxul de lucru. Am adoptat o regulă de a ne anunța reciproc în momentul în care începeam lucrul, dar și imediat ce finalizam un task. Această abordare a eliminat aproape complet riscul de a suprascrie codul celeilalte colegi.
- **Iterația și Îmbunătățirea Continua:** Cardurile marcate special în Trello (ex.: lista "GRIJA!!!" sau "Backend Critical") nu au fost doar pentru bug-uri, ci au servit ca un jurnal de testare. Pe măsură ce rulam aplicația, notam observații de UX sau funcționalități lipsă (ex. necesitatea ca un utilizator să își poată edita propriul mesaj într-un grup). Acest proces ne-a permis să transformăm observațiile din timpul rulării în task-uri concrete de dezvoltare, rafinând aplicația de la o etapă la alta.

Lista de observații și îmbunătățiri:



3. Evoluția Proiectului și Productivitatea

Productivitatea echipei a crescut constant:

- **Faza incipientă:** Ne-am concentrat pe structura bazei de date și pe sistemul de autentificare.
- **Faza intermediară:** Am implementat funcționalitățile Core: postări, comentarii, sistemul de grupuri și cel de follow.
- **Sprint-ul final:** Ne-am axat pe rafinare UX, sistemul de notificări complexe și panoul de administrare.

Pentru a menține un ritm de dezvoltare constant, am aplicat o strategie de prioritizare dinamică Funcționalitățile cu un grad mai ridicat de complexitate care riscau să blocheze progresul (precum opțiunea de a edita propriul mesaj într-un grup) au fost amâname strategic pentru etapa finală de rafinare. Această abordare ne-a permis să ne concentrăm inițial pe consolidarea nucleului esențial al aplicației, revenind ulterior asupra elementelor de finețe care îmbunătățesc experiența utilizatorului.

4. Impedimente și Soluții

Pe parcursul dezvoltării, am întâlnit mai multe provocări tehnice și de design:

i. Sincronizarea Frontend-Backend (Modalele de Urmăritori):

- **Impediment:** Implementarea listelor dinamice pentru “Următori” și “Urmăriți” a generat inițial erori de afișare (liste goale, layout suprapus) și probleme de rutare (redirecționare greșită către profilul utilizatorului din listă).
- **Soluție:** Am rescris logica de populare a modalelor, acordând o atenție sporită structurii HTML și parametrilor transmiși prin rutele ASP.NET Core (asp-route-username), asigurând o navigare fluidă.

ii. Gestiunea Permisii Complex (Grupuri și “Super-Admin”):

- **Impediment:** Integrarea rolului de “Administrator Global” peste logica existentă a grupurilor a creat conflicte de permisiuni. Inițial, sistemul nu permitea “Super-Adminului” să eliminate moderatori din grup sau să gestioneze membrii.
- **Soluție:** Am revizuit complet logica din GroupsController, implementând verificări ierarhice (ex.: User.IsInRole(Administrator) || isGroupOwner), astfel încât Adminul platformei să aibă prioritate asupra regulilor locale ale grupului.

iii. Gestionarea Conținutului Multimedia (Upload & Display):

- **Impediment:** Încărcarea și afișarea corectă a imaginilor și a videoclipurilor în feed și pe profil a fost o provocare de durată. Imaginile se distorsionau, iar videoclipurile nu se adaptau containerului.
- **Soluție:** Am ajustat stilurile CSS și clasele Bootstrap pentru a forța elementele media să respecte dimensiunile cardului, indiferent de rezoluția fișierului încărcat.

iv. Gestiunea Stării și Feedback-ul Vizual (Sistemul de Alertă):

- **Impediment:** Gestionarea mesajelor de sistem (succes/eroare/informare) prin TempData a generat probleme de persistență neașteptată a datelor. Am identificat un caz critic de utilizare: la deconectarea unui utilizator și conectarea imediată a altuia, mesajele de confirmare din sesiunea anterioară (ex.: postare ștearsă) erau afișate utilizatorului curent. De asemenea, alertele standard Bootstrap (verde/roșu intens) creau o discrepanță vizuală puternică față de tema aplicației.
- **Soluție:** Am implementat o logică nouă de curățare a sesiunii la Logout și am standardizat afișarea mesajelor într-un singur loc, eliminând afișarea haotică în pagini diferite. Am înlocuit stilurile implicite cu un design personalizat care utilizează paleta cromatică a brandului “Agora”, asigurând o experiență a utilizatorului fluidă și coerentă.

5. Concluzii

Proiectul “Agora” a reprezentat pentru noi mai mult decât o simplă temă. A fost o simulare realistă a dezvoltării unui produs software complet și complex. Am reușit să livrăm o aplicație funcțională, stabilă și coerentă vizual, atingând toate obiectivele propuse în etapa de planificare.

Un aspect definitoar al acestui proiect a fost atenția la detalii și inițiativa. Nu ne-am rezumat la cerințele minimale, ci am implementat funcționalități complexe, precum sistemul de notificări în timp real, element esențial într-o rețea socială modernă.

Dincolo de aspectele tehnice (ASP.NET Core, SQL, Framework), acest proiect ne-a consolidat abilitățile de lucru în echipă. Am învățat că o comunicare eficientă și o împărțire clară a task-urilor sunt la fel de (dacă nu și mai) importante ca și codul propriu-zis. Deși am întâmpinat blocaje tehnice, am reușit să le depășim prin documentare și depanare colaborativă, livrând un produs de care suntem mândre.