Requirements Document

Artists Land

Versiunea 1.0.0

Product Backlog cu Epics & User Stories: <https://github.com/users/allee15/projects/7>

Sprint 1 Backlog cu User Stories: <https://github.com/users/allee15/projects/8>

Sprint 2 Backlog cu User Stories: <https://github.com/users/allee15/projects/10>

**Cuprins**

1. [**Introducere**](#gjdgxs)
   1. [Scopul documentului](#30j0zll)
   2. [Domeniul aplicației](#1fob9te)
2. [**Descriere generală**](#3znysh7)
   1. [Contextul aplicației: De ce este necesară aplicația? Ce problemă rezolvă?](#2et92p0)
   2. [Obiective principale: Ce se dorește să realizeze aplicația?](#tyjcwt)
   3. [Stakeholderi: Cine sunt utilizatorii principali și părțile interesate?](#3dy6vkm)
3. [**Cerințe funcționale**](#1t3h5sf)
   1. [Funcționalități de bază](#4d34og8)
   2. [Scenarii de utilizare (Use Cases)](#2s8eyo1)
   3. [Diagrame funcționale](#17dp8vu)
4. [**Cerințe nefuncționale**](#3rdcrjn)
   1. [Securitate](#26in1rg)
   2. [Compatibilitate](#lnxbz9)
   3. [UI/UX](#35nkun2)
5. [**Arhitectura sistemului**](#1ksv4uv)
   1. [MVVM](#44sinio)
   2. [Structura generală](#2jxsxqh)
   3. [Integrare](#z337ya)
6. [**Planificare și livrabile**](#3j2qqm3)
   1. [Sprint 1](#1y810tw)
   2. [Sprint 2](#4i7ojhp)
7. [**Diagrama bazei de date**](#2xcytpi)
8. [**Diagrama Gantt**](#1ci93xb)
9. **Teste unitare**
   1. Descrierea testelor unitare
   2. Test Coverage
10. [**Test Driven Development (TDD)**](#3whwml4)
    1. [Avantaje](#2bn6wsx)
    2. [Dezavantaje](#qsh70q)
11. **Pair Programming vs individual:**
    1. Avantaje
    2. Dezavantaje
12. [**Sprint Daily Standups**](#3as4poj)
    1. [Sprint 1](#1pxezwc)
    2. [Sprint 2](#49x2ik5)
13. **Retrospectivă**
    1. Sprint 1
    2. Sprint 2
14. [**Raport munc**](#2p2csry)**ă** [**echip**](#2p2csry)**ă**
15. **Introducere**
    1. **Scopul documentului**

Scopul acestui document este de a detalia cerințele funcționale și nefuncționale pentru dezvoltarea aplicației mobile Artists Land, care promovează arta contemporană și susține artiștii în dezvoltarea carierei lor. Documentul oferă o descriere clară a funcționalităților aplicației, așteptările utilizatorilor și constrângerile tehnice, servind drept ghid pentru echipa de dezvoltare și stakeholderi.

Aplicația se adresează:

* **Simplilor utilizatori:** care pot explora operele artistice postate, interacționa cu artiștii prin chat și contribui la promovarea acestora.
* **Artiștilor:** care pot partaja opere de artă, interacționa cu publicul, edita conținut și accesa opțiuni suplimentare pentru promovarea lucrărilor printr-un sistem de nivel personalizat.

Principalele obiective ale aplicației includ:

* **Promovarea artei contemporane** printr-o platformă interactivă, accesibilă tuturor utilizatorilor, indiferent de vârstă.
* **Oferirea unui mediu sigur și intuitiv** pentru autentificare, navigare și gestionarea conținutului utilizatorilor.
* **Facilitarea interacțiunii între utilizatori și artiști** prin funcționalități precum feed-ul principal și chat-ul interactiv.
* **Susținerea artiștilor în promovarea lucrărilor lor** printr-un sistem de token-uri și nivele personalizate, prioritizând vizibilitatea acestora în feed-ul principal.

Acest document reprezintă o referință pentru planificarea, implementarea, testarea și lansarea aplicației, contribuind la o înțelegere comună între toate părțile implicate.

**1.2 Domeniul aplicației**

Aplicația *Artists Land* se încadrează în domeniul artei contemporane și al platformelor digitale dedicate promovării creațiilor artistice. Aceasta oferă o soluție inovatoare pentru conectarea artiștilor cu publicul larg, într-un mediu digital interactiv.

Într-un context în care accesul la artă este din ce în ce mai digitalizat, iar artiștii caută platforme prin care să își expună lucrările și să atragă audiențe noi, aplicația propune:

* *Promovarea artei* printr-un feed centralizat cu lucrări creative accesibile oricui.
* *Conectarea artiștilor cu publicul* prin intermediul funcționalităților interactive, precum chat-ul.
* *Susținerea noilor artiști* printr-un sistem de niveluri personalizate, care le oferă vizibilitate proporțională cu nivelul personal, asigurând o distribuție prioritizată a postărilor în feed.

Aceasta răspunde nevoilor:

* Utilizatorilor obișnuiți, care doresc să descopere și să aprecieze arta într-un mod accesibil și ușor de utilizat.
* Artiștilor profesioniști sau la început de carieră, care au nevoie de o platformă pentru a-și prezenta lucrările, a interacționa cu publicul și a-și dezvolta imaginea profesională.

Prin integrarea tehnologiei moderne, *Artists Land*își propune să redefinească modul în care arta este descoperită, apreciată și promovată în era digitală.

1. **Descriere generală**
   1. **Contextul aplicației: De ce este necesară aplicația? Ce problemă rezolvă?**

În contextul digital actual, există o lipsă de platforme dedicate exclusiv artei contemporane care să combine accesibilitatea cu funcționalități avansate de promovare. Aplicația noastră vine să umple acest gol, oferind artiștilor posibilitatea de a-și prezenta lucrările și de a interacționa direct cu admiratorii lor.

Aplicația *Artists Land* este necesară datorită creșterii cererii pentru platforme digitale dedicate promovării artei și conectării artiștilor cu un public global. În prezent, mulți artiști, în special cei aflați la începutul carierei, întâmpină dificultăți în promovarea creațiilor lor, fie din lipsă de resurse, fie din cauza limitărilor platformelor existente. De asemenea, publicul larg nu are mereu acces ușor la operele artiștilor contemporani.

Problema pe care o rezolvă aplicația:

* Crearea unui spațiu virtual care să permită artiștilor să își prezinte lucrările într-un mod profesionist și accesibil.
* Facilitarea interacțiunii directe dintre artiști și utilizatori prin funcționalități intuitive precum chat-ul.
* Promovarea egală a artiștilor, indiferent de nivelul lor de notorietate, printr-un sistem de vizibilitate bazat pe token-uri și niveluri personalizate.
  1. **Obiective principale: Ce se dorește să realizeze aplicația?**

Aplicația **Artists Land** își propune să atingă următoarele obiective:

* **Promovarea artei contemporane** printr-o platformă dedicată artiștilor, unde aceștia pot posta lucrările lor și pot atrage o audiență variată.
* **Conectarea artiștilor cu publicul larg** prin funcții interactive, oferind utilizatorilor o experiență captivantă.
* **Susținerea noilor artiști** prin oferirea unei platforme competitive, care le permite să concureze pentru vizibilitate cu artiști consacrați.
* **Asigurarea unui mediu sigur și accesibil** pentru toți utilizatorii, prin funcționalități precum autentificarea cu 2FA, resetarea parolei și gestionarea datelor personale.
* **Generarea de venituri pentru artiști** printr-un sistem de token-uri, care le permite să își promoveze lucrările în funcție de bugetul disponibil.
  1. **Stakeholderi: Cine sunt utilizatorii principali și părțile interesate?**

Utilizatorii principali:

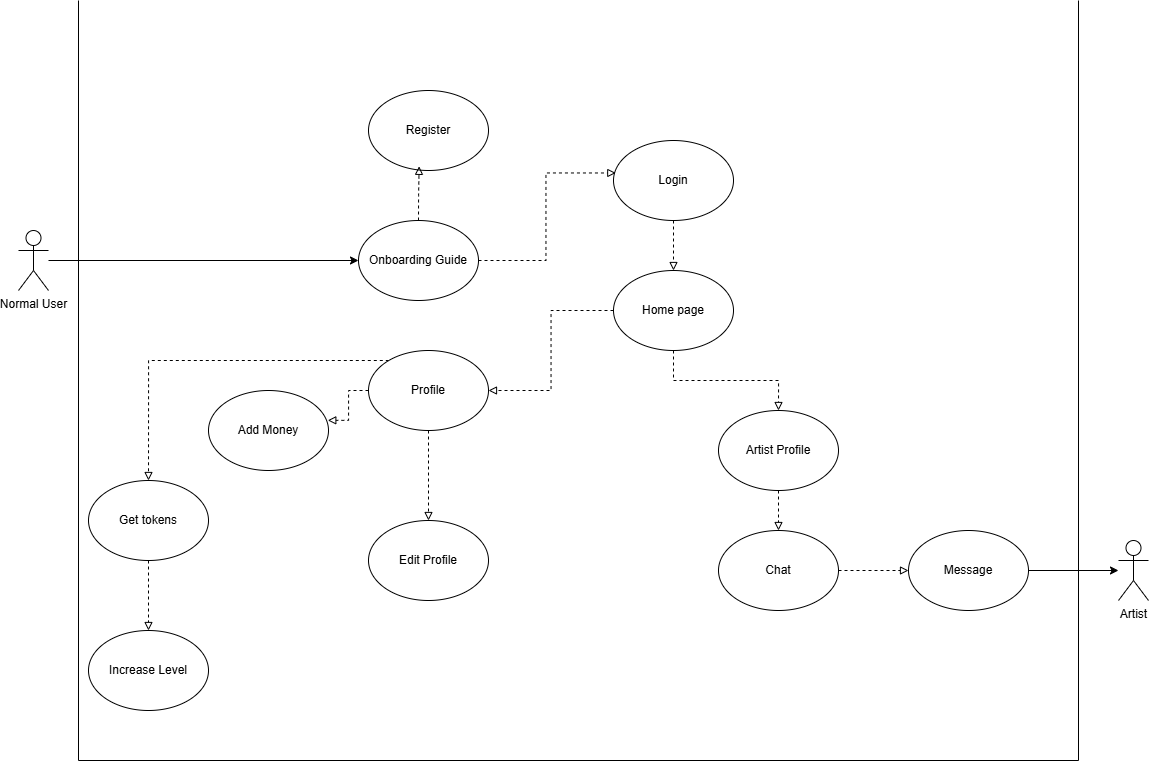
* Artiștii profesioniști: Cei care doresc să își promoveze lucrările și să interacționeze cu publicul.
* Artiștii aflați la început de carieră: Cei care caută să își construiască o imagine și să atragă o audiență nouă.
* Utilizatorii obișnuiți: Cei interesați de artă, care doresc să exploreze și să aprecieze lucrările artiștilor, având posibilitatea să comunice direct cu aceștia.

Părțile interesate (stakeholderii externi):

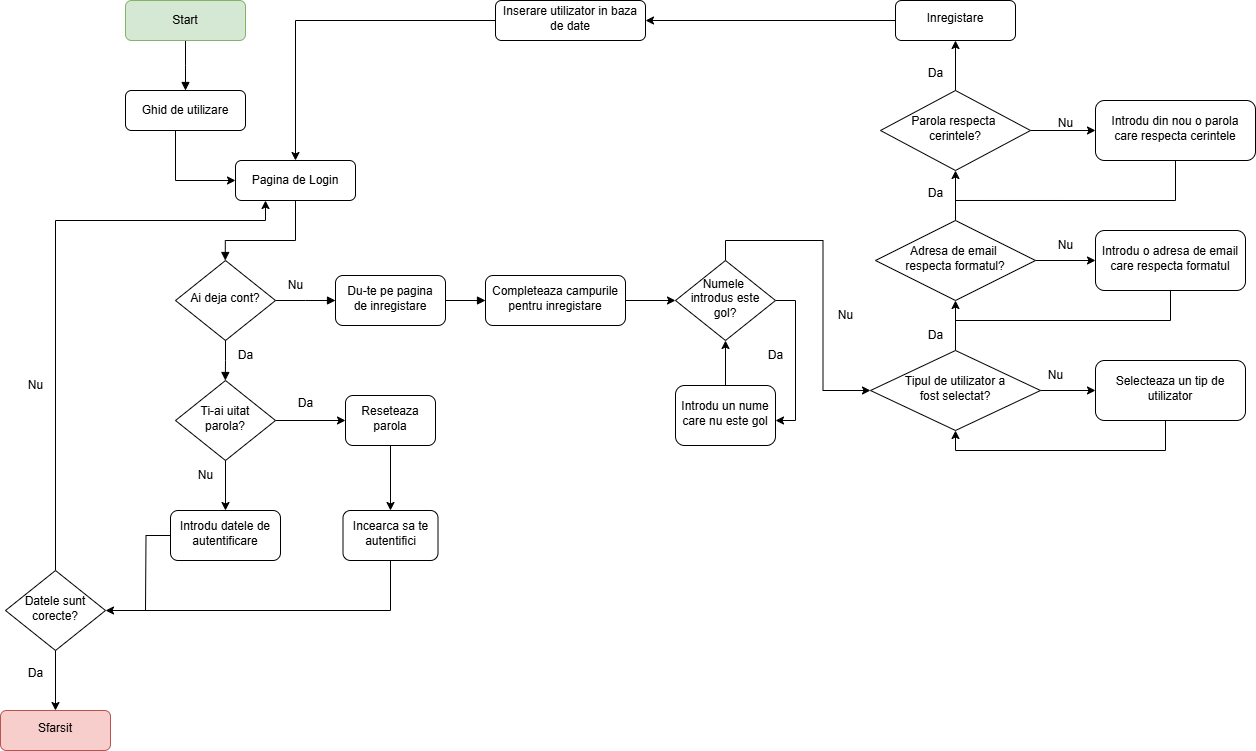
* Dezvoltatorii aplicației: Echipa responsabilă de implementarea și întreținerea funcționalităților.
* Investitorii sau sponsorii: Cei care susțin financiar dezvoltarea aplicației și doresc să obțină venituri din funcționalitățile premium.
* Promotori culturali: Organizațiile sau grupurile care promovează arta contemporană și sunt interesate să folosească aplicația ca mijloc de promovare.

1. **Cerințe funcționale**
   1. **Funcționalități de bază**

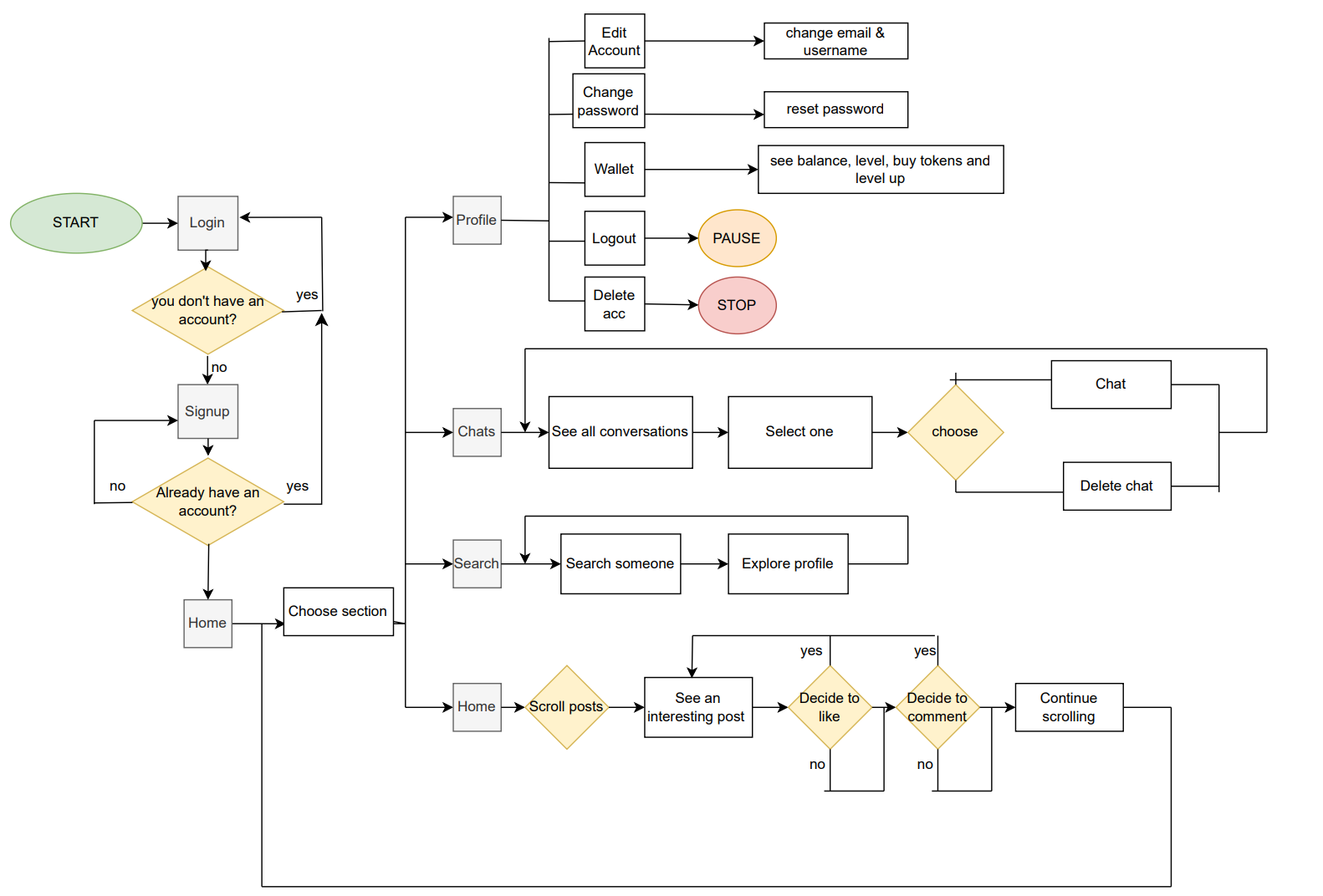
* Autentificare și crearea contului
  + Utilizatorii pot crea un cont selectând tipul de utilizator: obișnuit sau artist.
  + Autentificarea se face prin email, parolă și un cod 2FA pentru securitate sporită.
  + Resetarea parolei este disponibilă printr-un proces simplu și sigur.
* Feed principal (Home Page)
  + Afișează lucrările artistice postate de artiști, în funcție de nivelul de promovare și preferințele utilizatorilor.
  + Oferă opțiuni de interacțiune, precum aprecierea sau comentarea postărilor.
* Profilul utilizatorului
  + Utilizatorii obișnuiți își pot actualiza datele personale, șterge contul sau se pot deconecta.
  + Artiștii pot adăuga, edita sau șterge imagini artistice direct din profilul lor.
* Postare și management al conținutului artistic (pentru artiști)
  + Artiștii pot încărca imagini artistice cu descrieri personalizate.
  + Postările pot fi gestionate (editate sau șterse) din profil.
* Chat interactiv
  + Utilizatorii obișnuiți pot comunica direct cu artiștii, adresând întrebări printr-un chat integrat.
  + Artiștii primesc notificări despre noile mesaje și pot răspunde utilizatorilor.
* Promovarea lucrărilor artistice
  + Artiștii pot achiziționa token-uri pentru a crește vizibilitatea postărilor în feed.
  + Sistemul de *Personal Level* determină prioritatea postărilor în funcție de nivelul achiziționat.
* Ghid introductiv pentru utilizatori noi
  + Un tutorial interactiv, format din trei pagini, explică funcționalitățile aplicației. Acesta poate fi omis la alegerea utilizatorului.
* Administrarea financiară (pentru artiști)
  + Posibilitatea de a adăuga fonduri în cont, care sunt convertite în token-uri pentru promovarea lucrărilor.
  + Tranzacțiile sunt securizate și integrate direct în aplicație.
* Deconectare și ștergere cont
  + Toți utilizatorii au posibilitatea de a se deconecta sau de a șterge contul direct din setările aplicației.
  1. **Scenarii de utilizare (Use Cases)**



* 1. **Diagrame funcționale**
* Diagrama de flux pentru procesul de autentificare (include și validările datelor la înregistrare).



* Diagrama de flux pentru fluxul general de utilizare a aplicației.



1. **Cerințe nefuncționale**
   1. **Securitate**

* Autentificare sigură
  + Utilizatorii trebuie să se autentifice folosind un email, o parolă și un cod 2FA pentru protecție suplimentară.
* Protecția datelor utilizatorilor
  + Aplicația trebuie să respecte standardele GDPR, asigurându-se că datele utilizatorilor sunt procesate și stocate în siguranță.
  + Comunicarea între aplicație și server va fi criptată folosind HTTPS.
* Prevenirea atacurilor de tip hacking
  + Tranzacțiile financiare pentru achiziționarea de token-uri vor fi procesate printr-un serviciu de plată certificat (Stripe).
  1. **Compatibilitate**
* Platforme suportate
  + Aplicația va fi disponibilă pentru dispozitive iOS.
  + Compatibilitatea va fi garantată pentru versiuni mai recente de sisteme de operare: iOS 17+.
* Rezoluții multiple
  + Interfața va fi optimizată pentru o gamă largă de dispozitive, inclusiv smartphone-uri și tablete.
  + Layout-ul va fi responsive și adaptabil la ecrane cu dimensiuni diferite de pixeli (DPI).
* Conectivitate minimă
  + Aplicația va funcționa optim pe conexiuni de internet 3G/4G, cu feed-uri actualizate în timp real.
  1. **UI/UX**
* Design intuitiv
  + Interfața utilizatorului trebuie să fie minimalistă, modernă și ușor de navigat, având un stil coerent și atractiv.
  + Aplicația va respecta principiile de design specifice pentru iOS: *Human Interface Guidelines*.
* Accesibilitate
  + Aplicația va include opțiuni de accesibilitate, precum contrast sporit, mărirea textului și suport pentru screen readers.
  + Elementele interactive vor fi ușor de utilizat chiar și pentru persoanele cu dificultăți motorii.
* Performanță UI
  + Navigarea între pagini și interacțiunile utilizatorilor trebuie să se desfășoare fără latențe vizibile (timp de răspuns < 300ms).

1. **Arhitectura sistemului**

Aplicația mobilă este construită cu SwiftUI, utilizând MVVM pentru separarea logicii de interfața grafică. Backend-ul, implementat în Express.js, gestionează autentificarea, stocarea imaginilor și funcționalitățile de chat și plăți.

* 1. **MVVM**

Aplicația va utiliza arhitectura **MVVM** pentru a separa logic funcționalitatea de interfața grafică, oferind un cod mai organizat și ușor de întreținut.

* Descriere generală a componentelor:
  + Model: Gestionează datele și logica aplicației. Include operațiuni precum autentificarea, managementul utilizatorilor și procesarea tranzacțiilor.
  + ViewModel: Intermediar între Model și View, transformând datele brute în informații pregătite pentru afișare. Utilizează framework-uri reactive, cum ar fi Combine (iOS).
  + View: Responsabil pentru afișarea interfeței utilizatorului și pentru captarea interacțiunilor utilizatorului.
* Avantajele arhitecturii MVVM:
  + Separarea clară a responsabilităților între logică și UI.
  + Scalabilitate și posibilitate de reutilizare a codului.
  + Ușurință în testarea unităților individuale (ViewModel).
  1. **Structura generală**

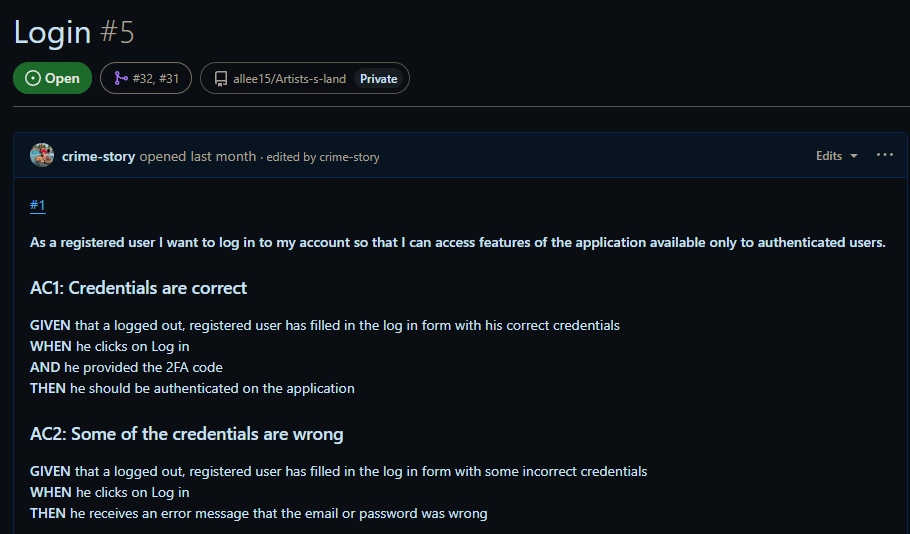
Aplicația va fi organizată pe module, fiecare având responsabilități specifice, pentru a încuraja reutilizarea și scalabilitatea:

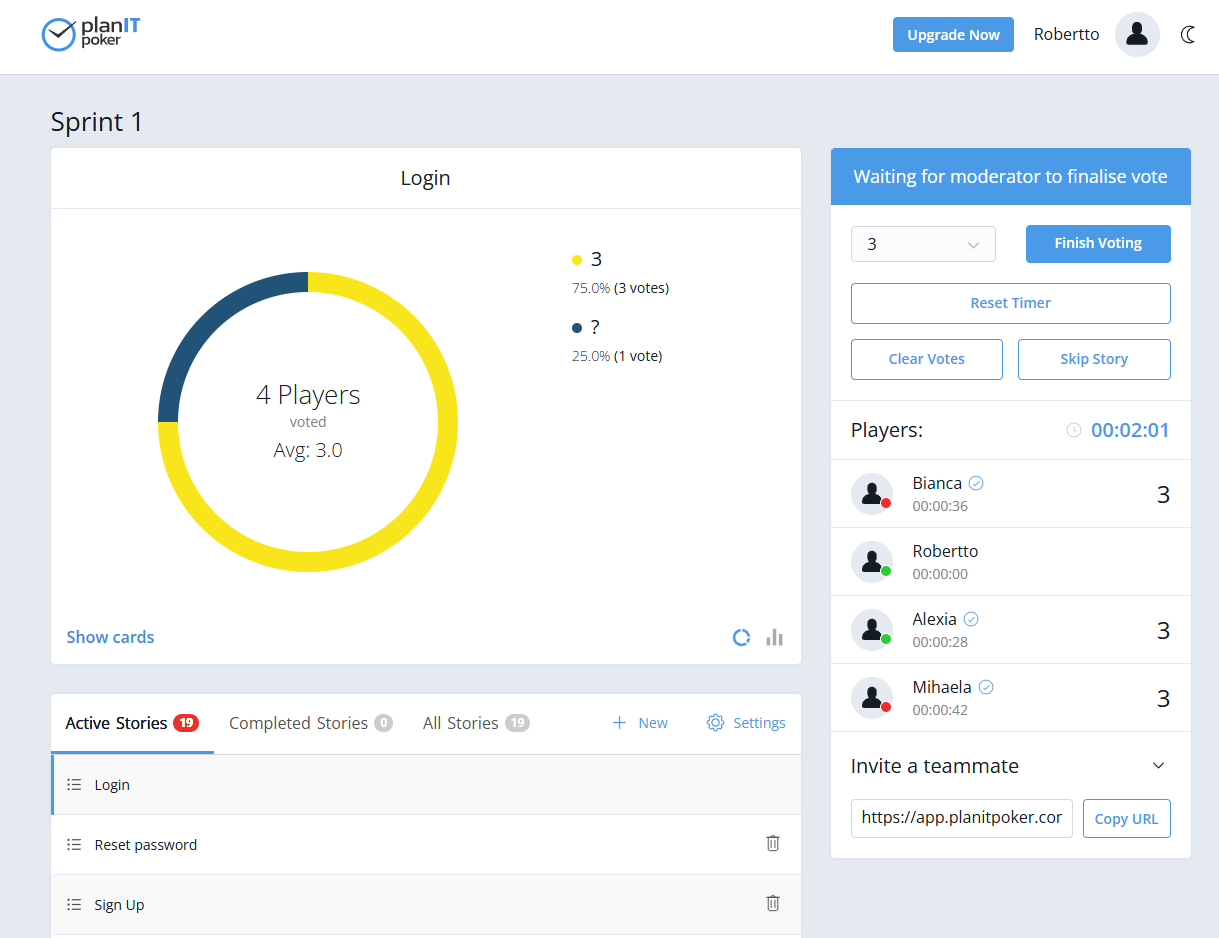
* **Autentificare**: Gestionarea proceselor de logare, înregistrare și resetare a parolei.
* **Feed**: Afișarea postărilor artistice și a conținutului dinamic.
* **Profil utilizator**: Gestionarea informațiilor personale și a setărilor contului.
* **Chat**: Comunicare directă între utilizatori și artiști.
* **Plăți**: Procesarea tranzacțiilor pentru achiziția de token-uri.
* **Fluxul de date**:
  + Utilizarea unui backend RESTful API, care va furniza date aplicației și va gestiona operațiunile critice (ex. autentificare, încărcare imagini).
  + Toate solicitările API vor fi gestionate printr-o clasă de tip \*\*Service Layer\*\*, care centralizează logica de comunicare cu serverul.
  1. **Integrare**
* Integrare parte mobilă
  + Aplicația mobilă va fi dezvoltată utilizând *SwiftUI* pentru iOS, un framework declarativ modern, care oferă o experiență de dezvoltare rapidă și o interfață utilizator intuitivă.
  + Avantajele utilizării *SwiftUI*:
    - Cod concis și ușor de citit datorită stilului declarativ.
    - Integrare nativă cu ecosistemul Apple, incluzând suport pentru funcții precum accesibilitate, Dark Mode și animații fluide.
    - Optimizare pentru performanță pe dispozitive iOS recente.
  + Aplicația va include componente modulare, precum:
    - **Feed-ul principal** pentru afișarea operelor artistice.
    - **Profilul utilizatorului** pentru gestionarea datelor personale.
    - **Chat-ul** pentru comunicarea în timp real.
    - **Search-ul** pentru a căuta artiști
* Integrare backend
  + Partea de backend va fi dezvoltată folosind *Express.js*, un framework minimalist și rapid pentru Node.js, pentru a gestiona eficient comunicarea între aplicația mobilă și baza de date.
  + Avantajele utilizării *Express.js*:
    - Structură flexibilă, ideală pentru aplicații moderne.
    - Suport excelent pentru crearea de API-uri RESTful.
    - Integrare rapidă cu alte biblioteci Node.js, precum middleware-uri pentru autentificare (ex. Passport.js) sau validarea datelor.
  + Responsabilități principale ale backend-ului:
    - Gestionarea utilizatorilor (înregistrare, autentificare, resetare parolă).
    - Stocarea și furnizarea imaginilor artistice.
    - Procesarea tranzacțiilor financiare pentru achiziția de token-uri.
  + Fluxul de date între *SwiftUI* și *backend*
    - Aplicația mobilă va comunica cu backend-ul folosind solicitări **HTTPS** și răspunsuri în format JSON.
    - Toate solicitările vor fi gestionate printr-un **Service Layer** în aplicația mobilă, care centralizează logica de rețea.
* Integrare cu servicii terțe
  + Plăți online: Tranzacțiile financiare vor fi gestionate printr-un gateway de plată precum Stripe, pentru securitate și fiabilitate.
* Testare și monitorizare
  + Integrarea între SwiftUI și Express.js va fi testată extensiv utilizând: Postman pentru verificarea API-urilor backend.

1. **Planificare și livrabile**

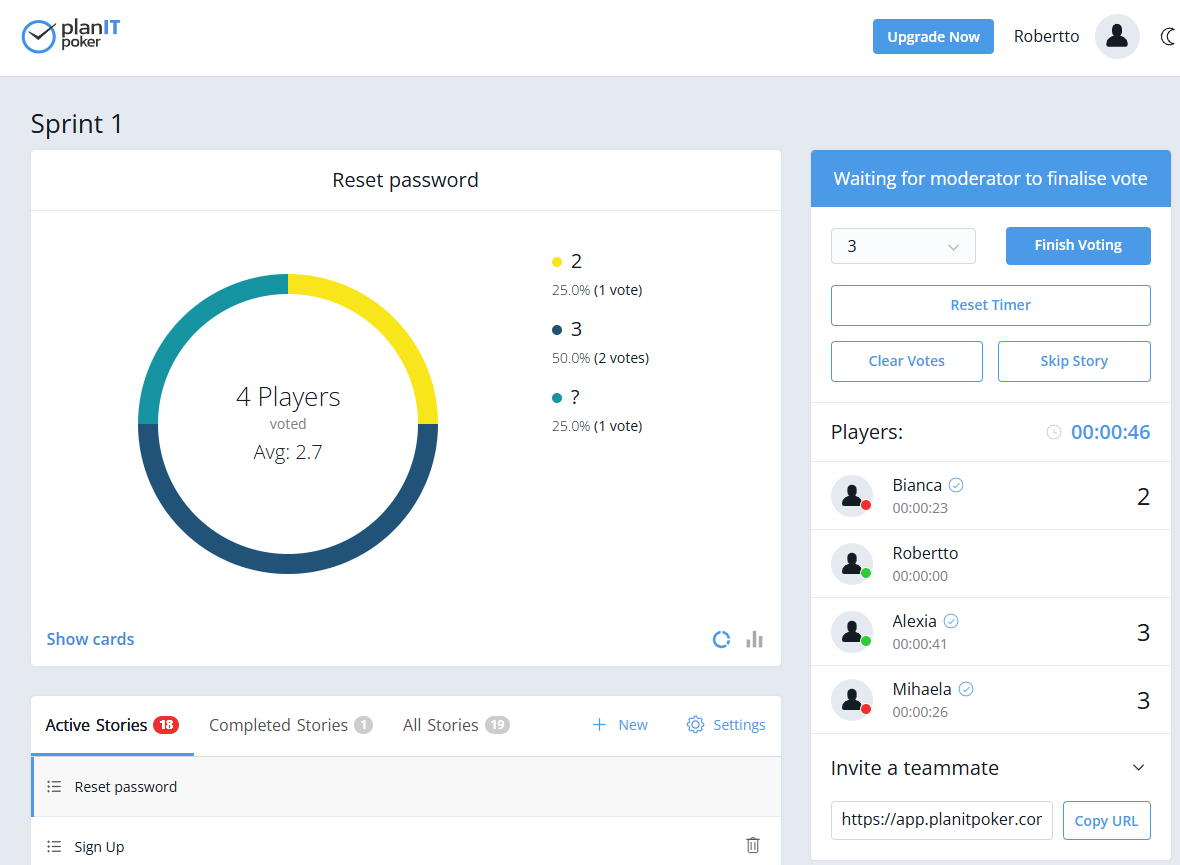
Proiectul nostru va fi împărțit în două sprint-uri:

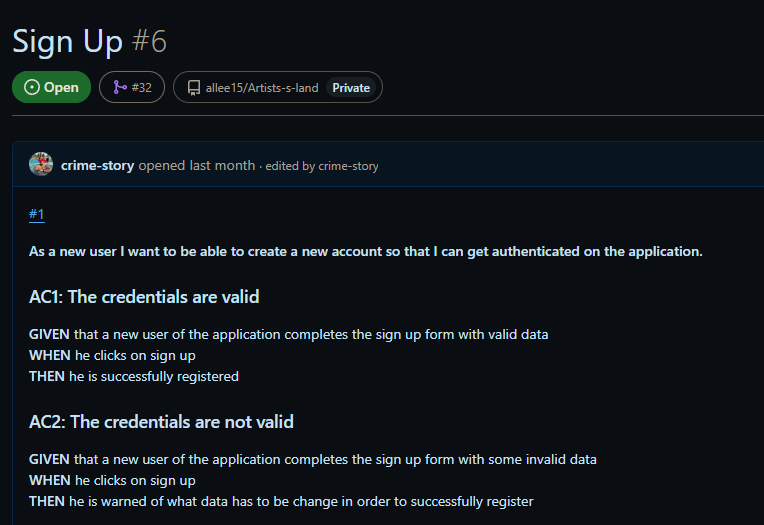
* Sprint 1 (20 Octombrie – 30 Decembrie 2024)
* Sprint 2 (1 Ianuarie – 19 Ianuarie 2025)
  1. **Sprint 1**



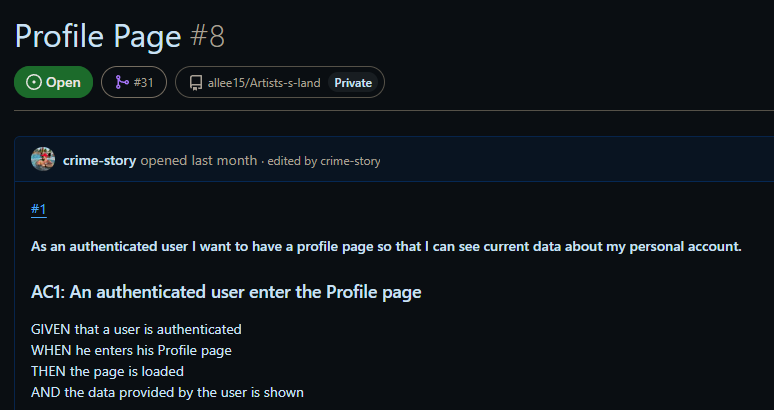




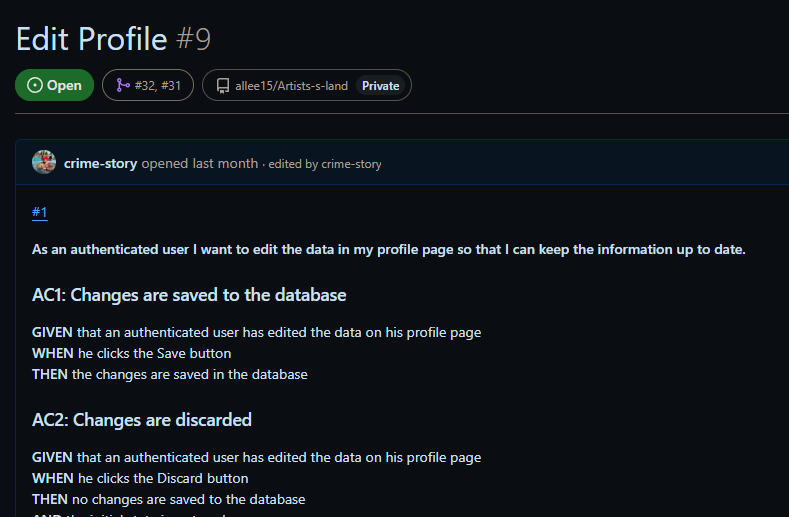


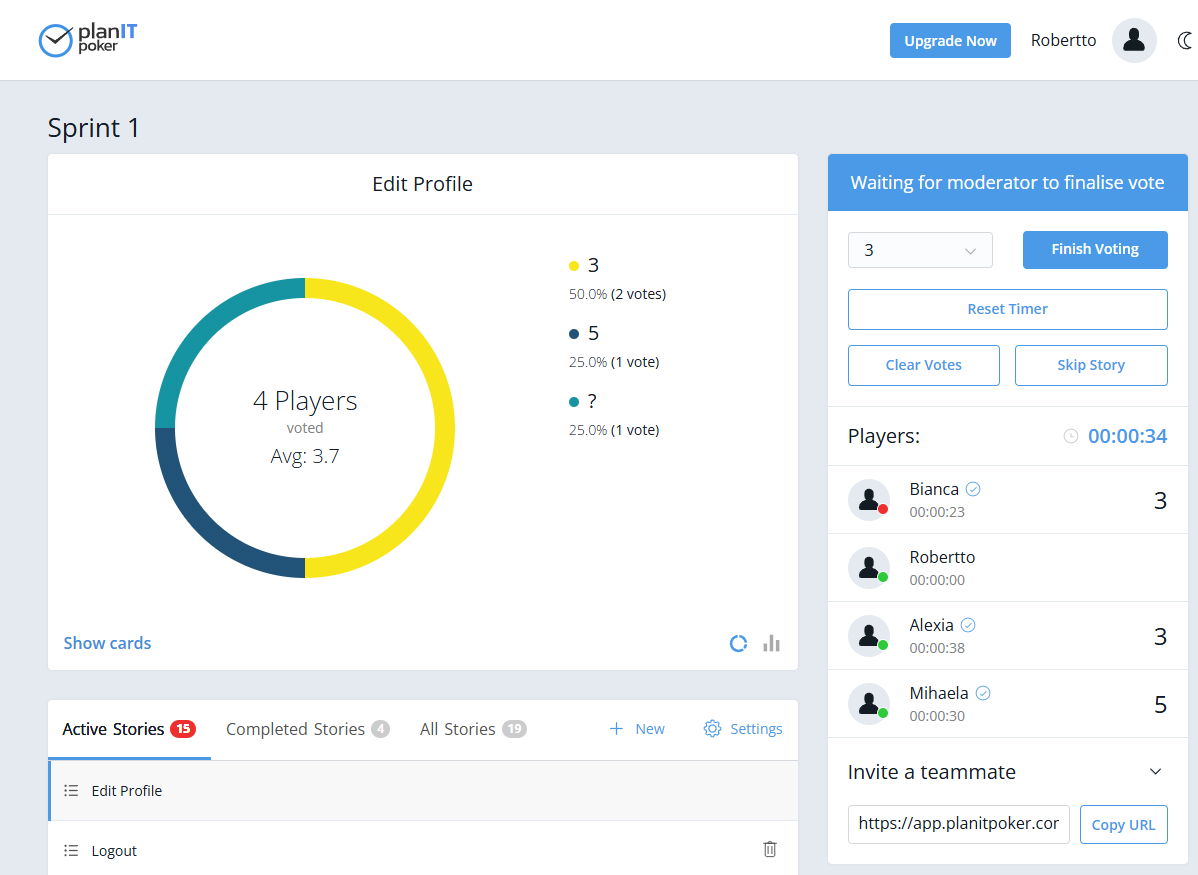


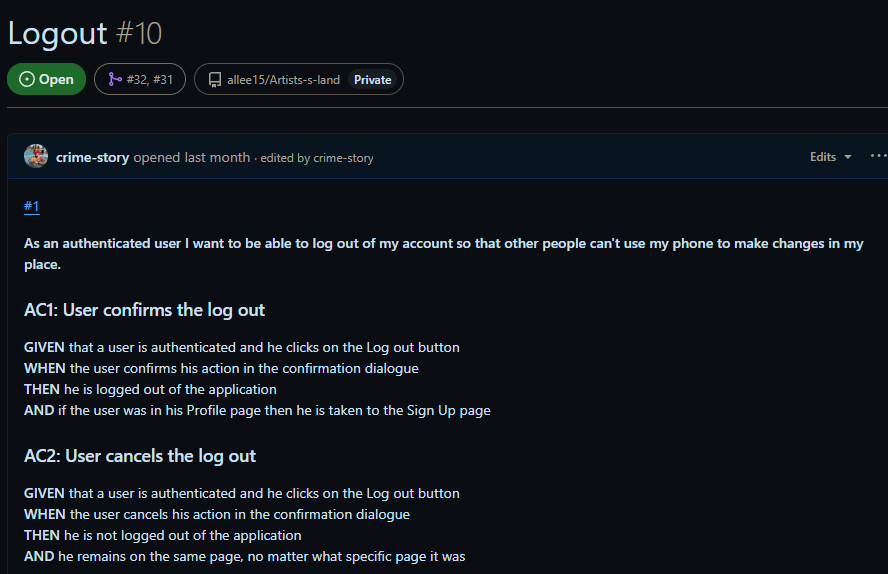


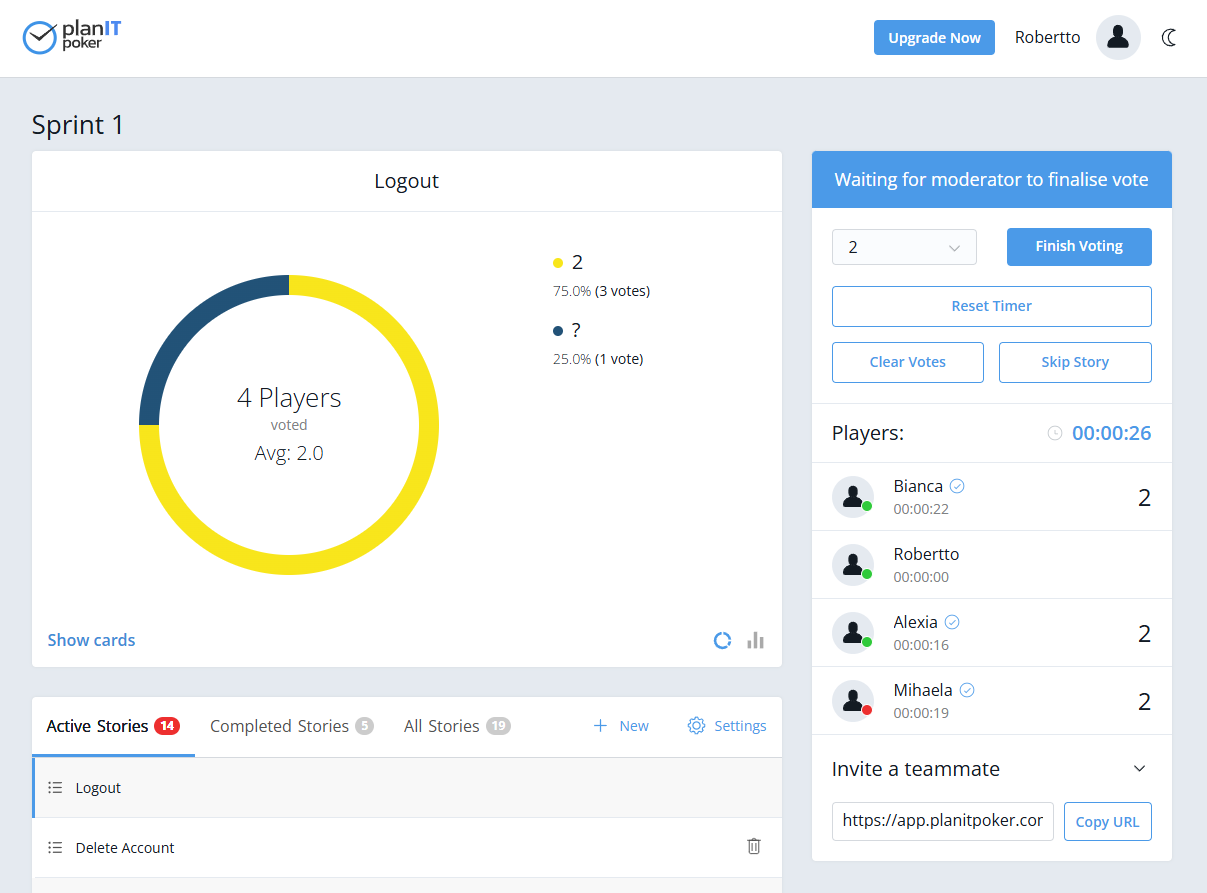


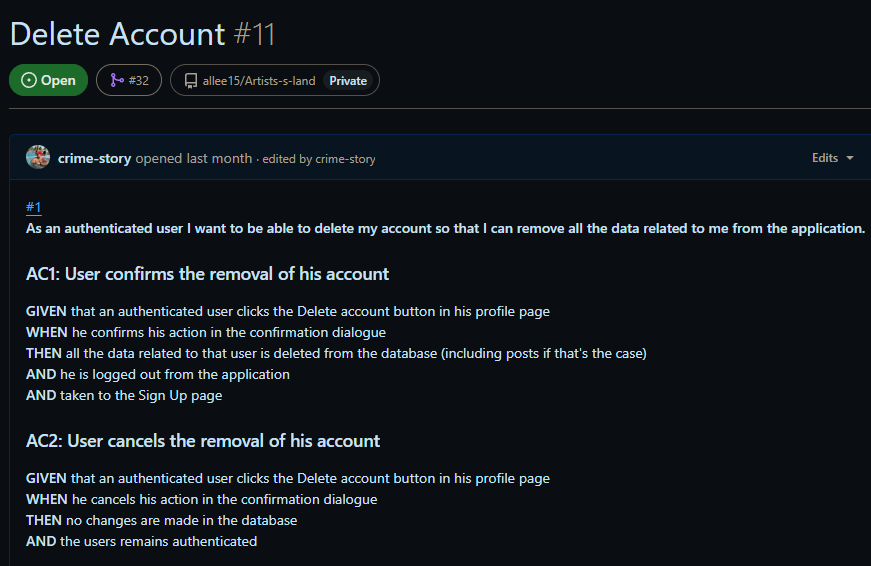


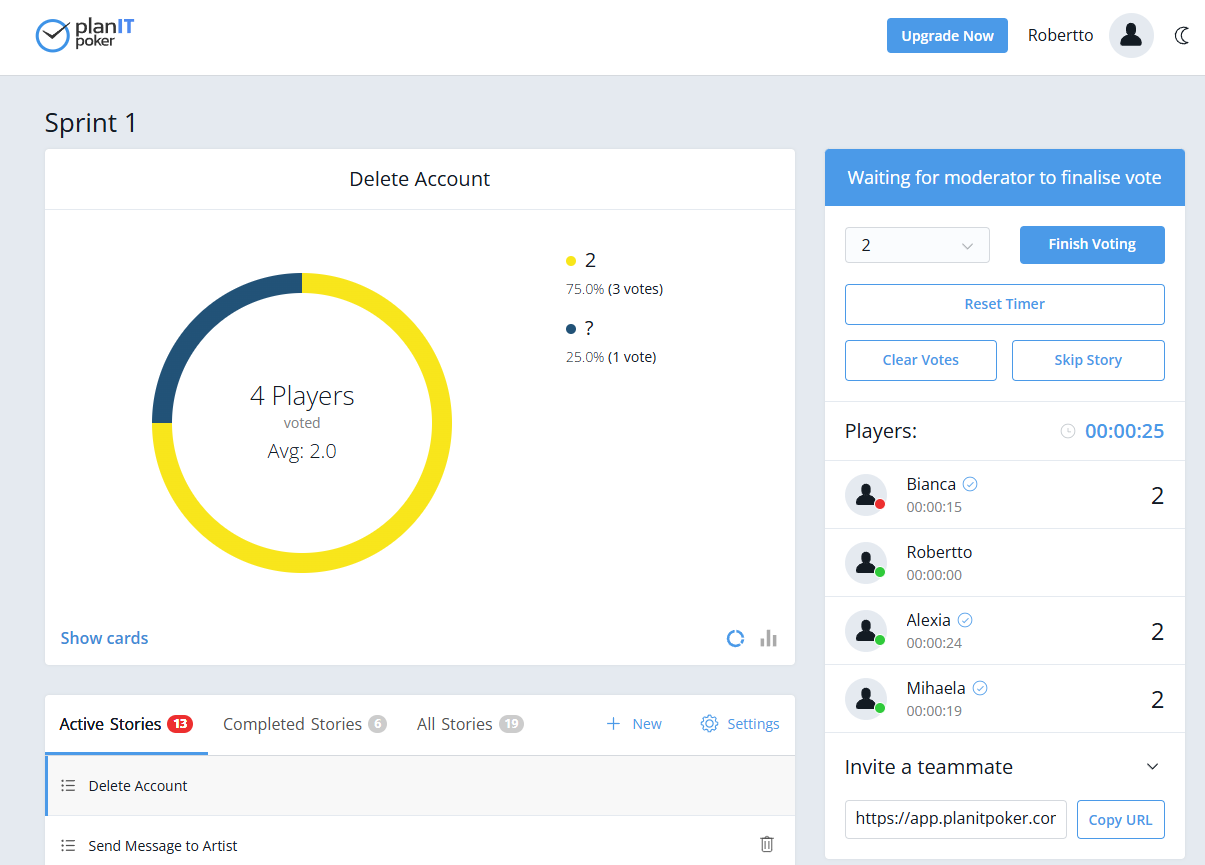




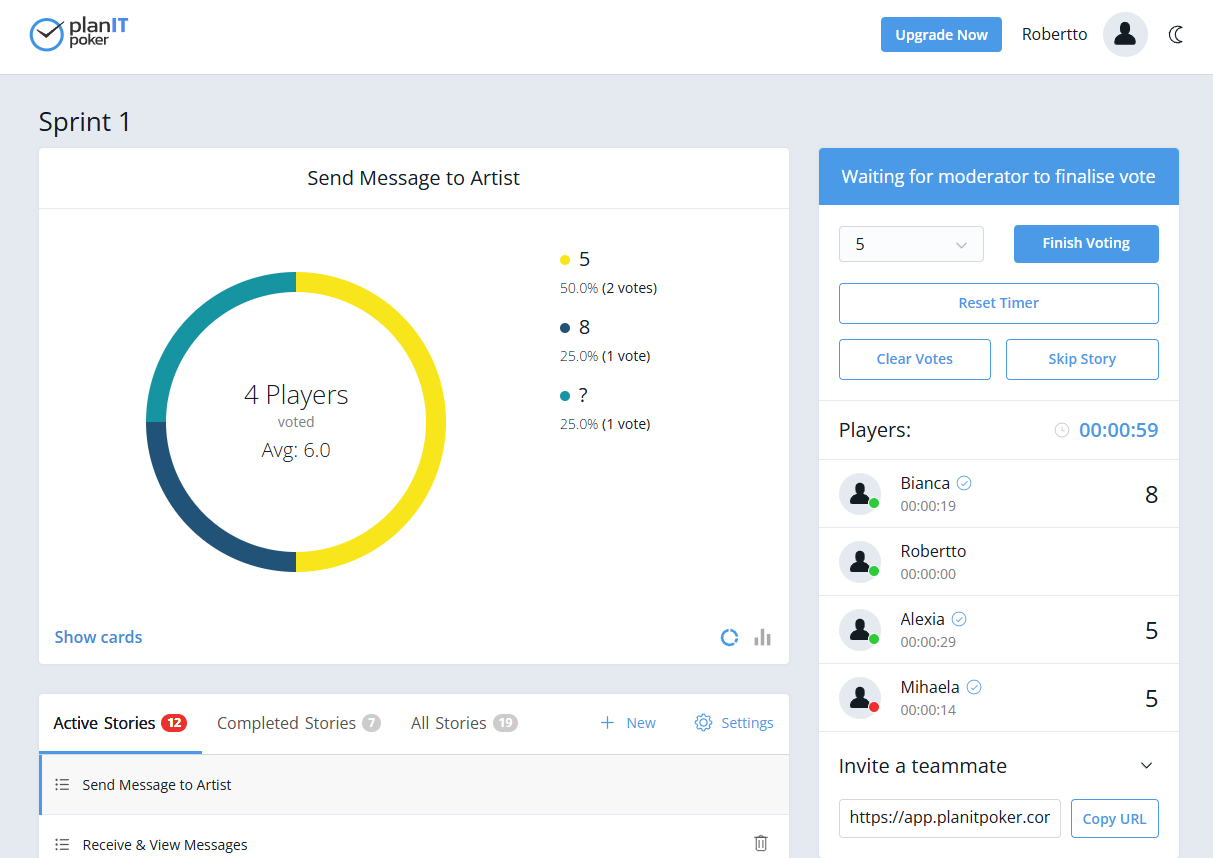




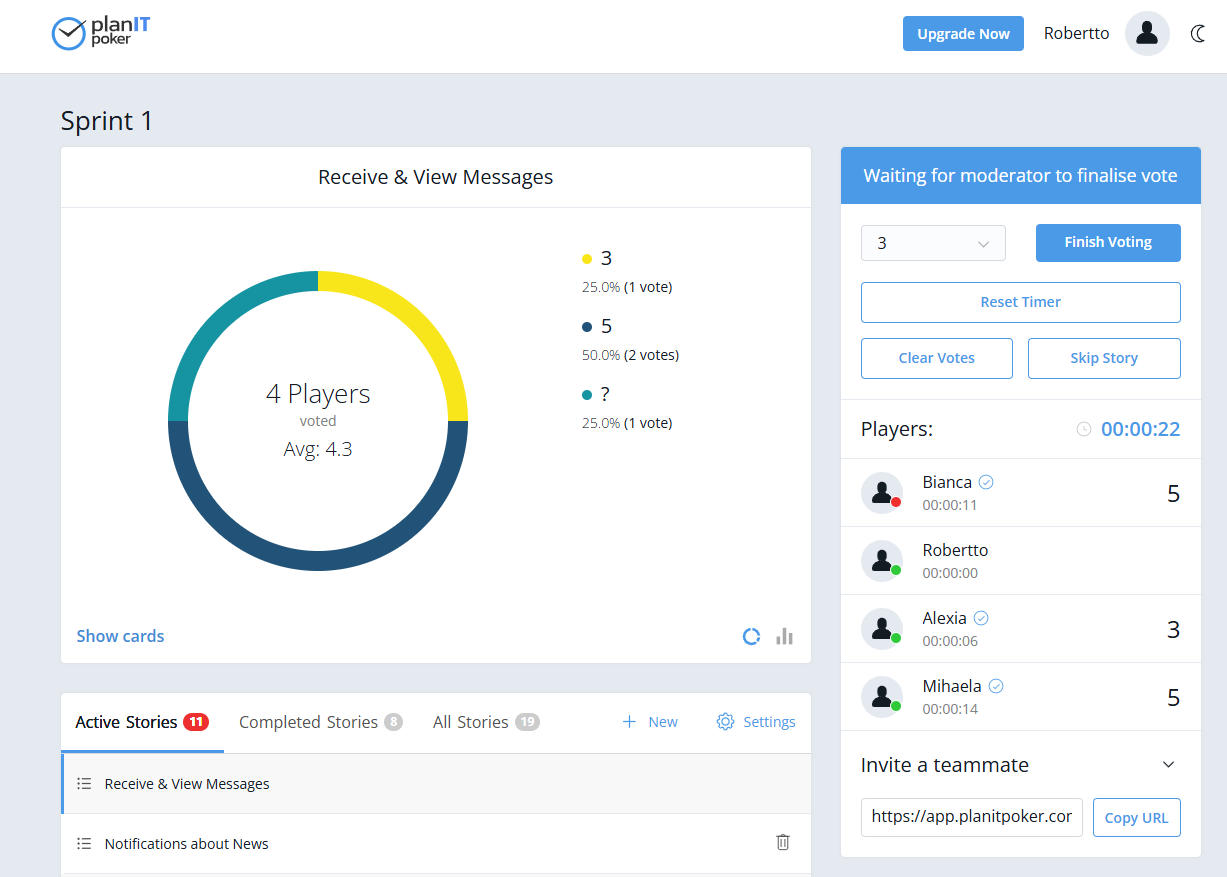






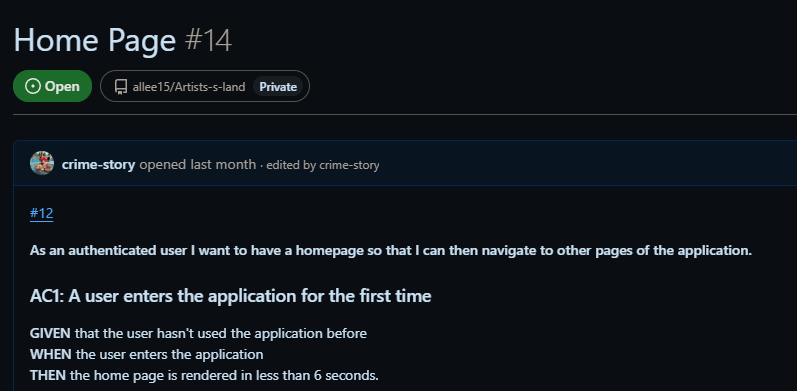


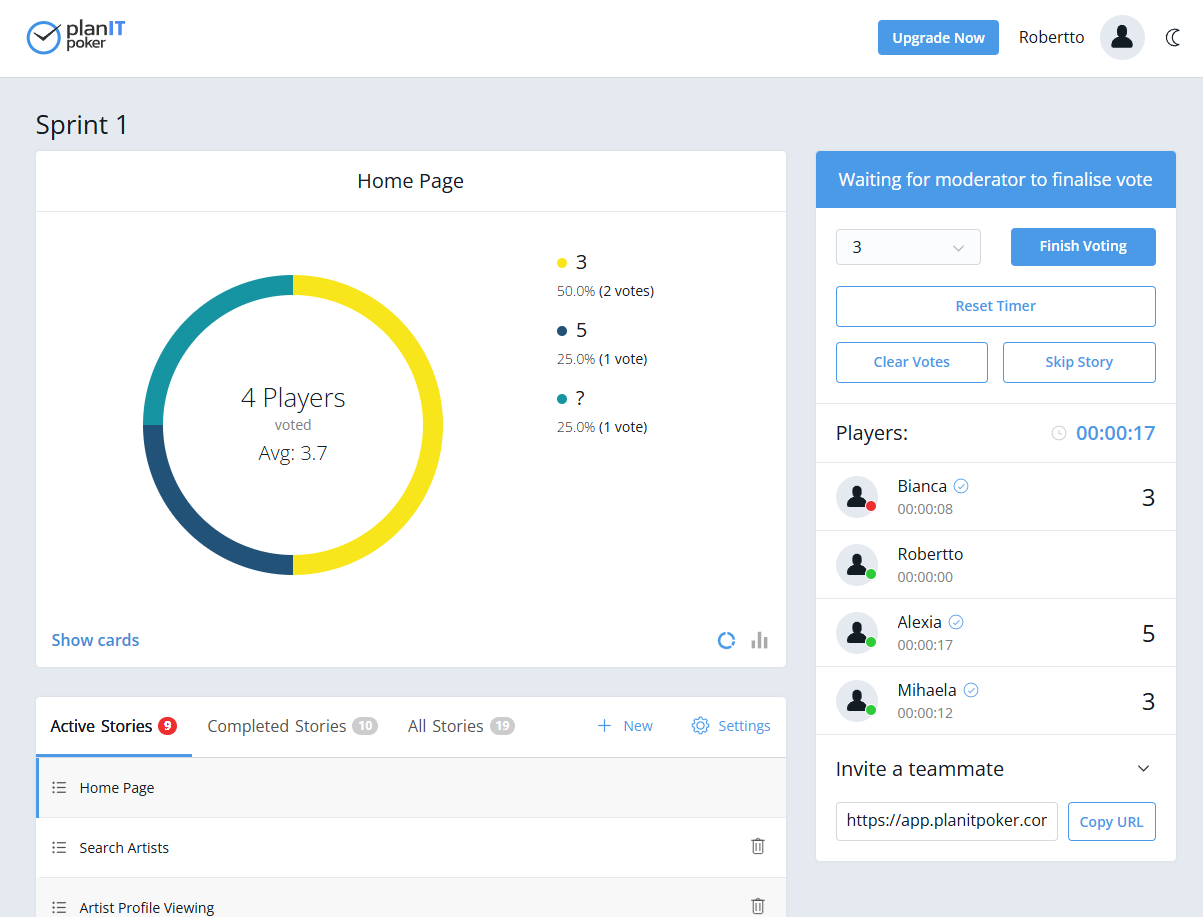








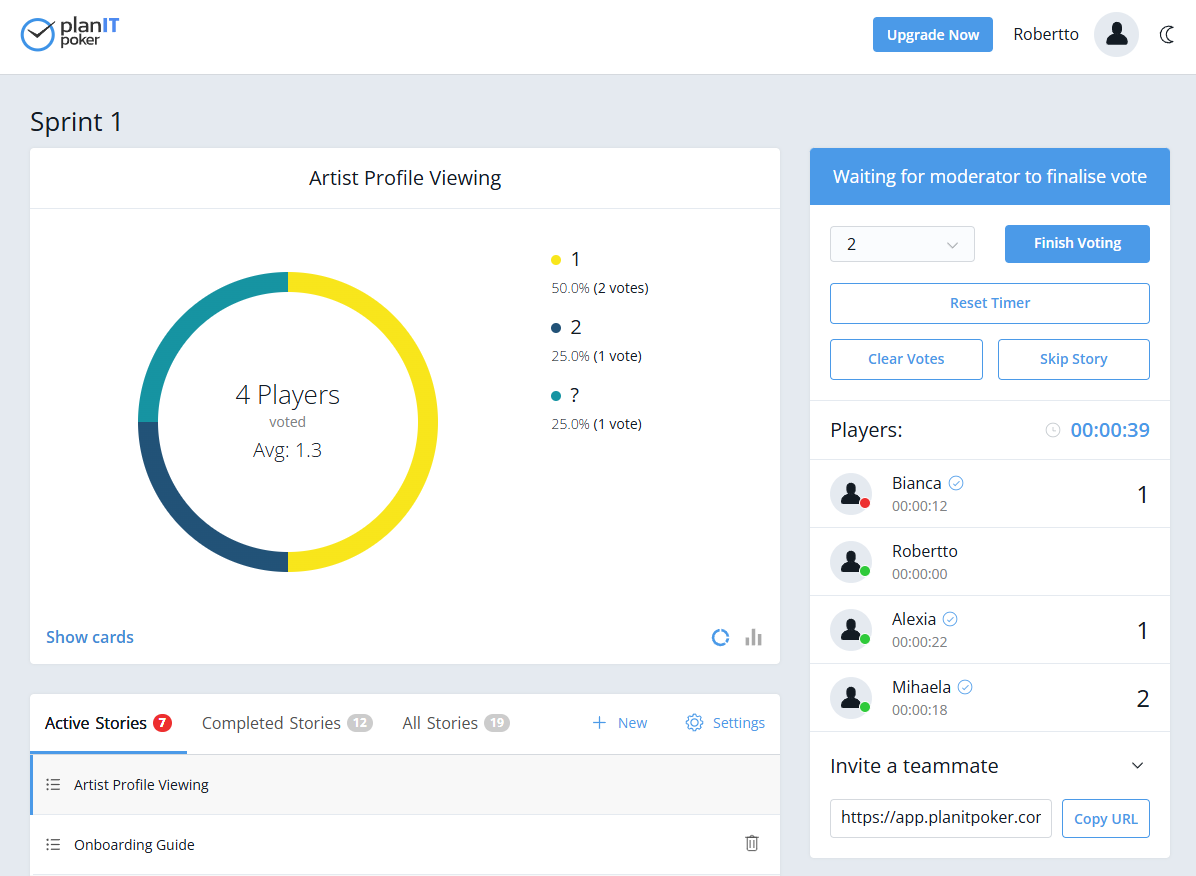


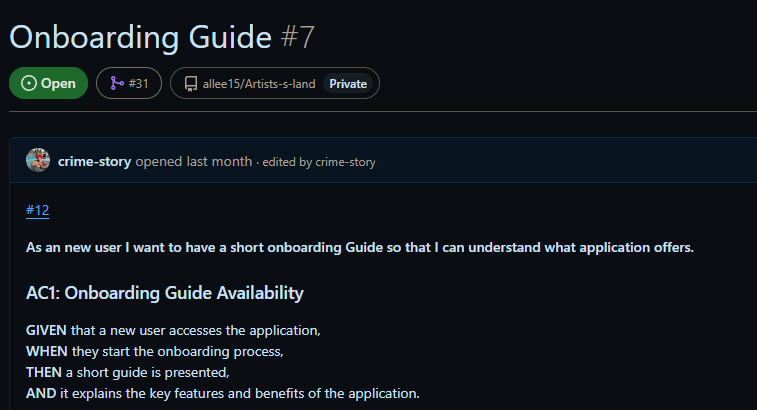


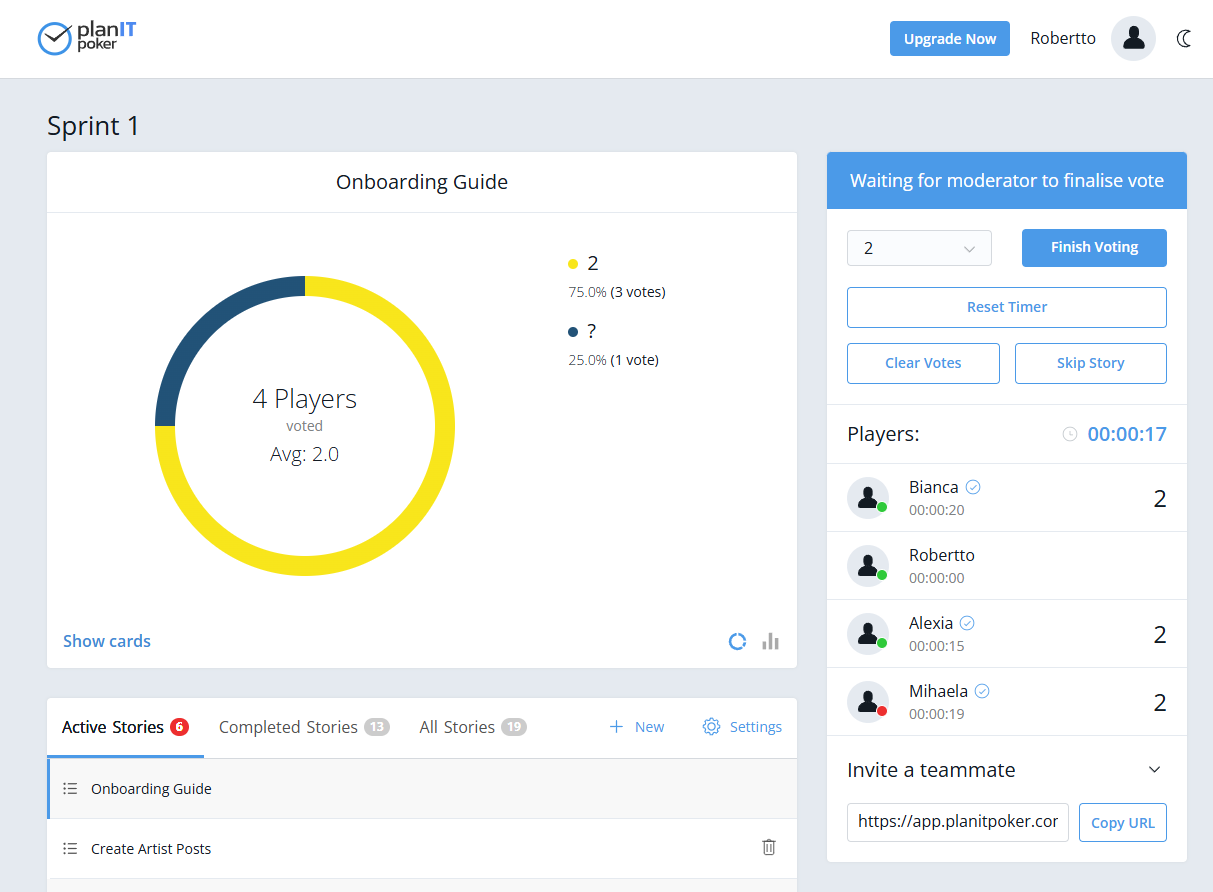


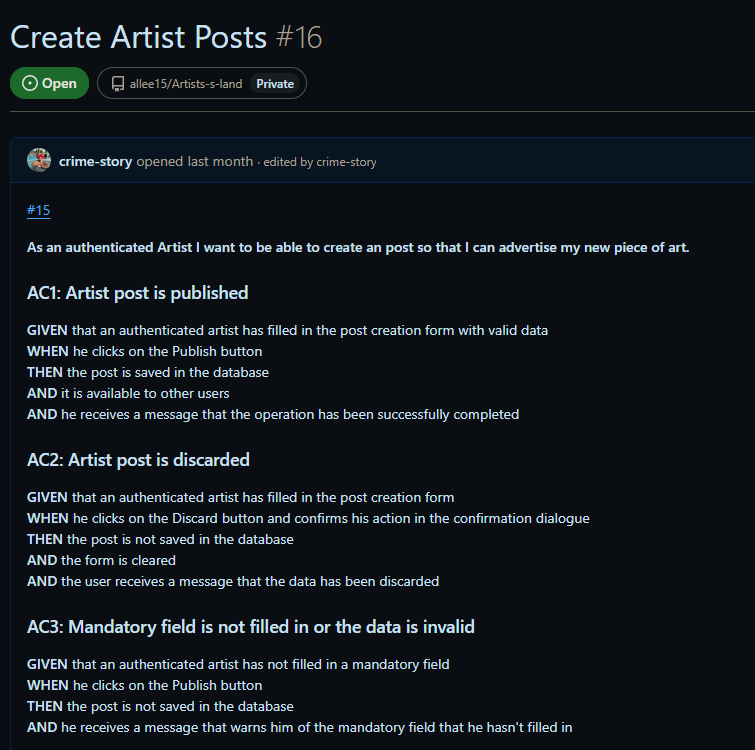


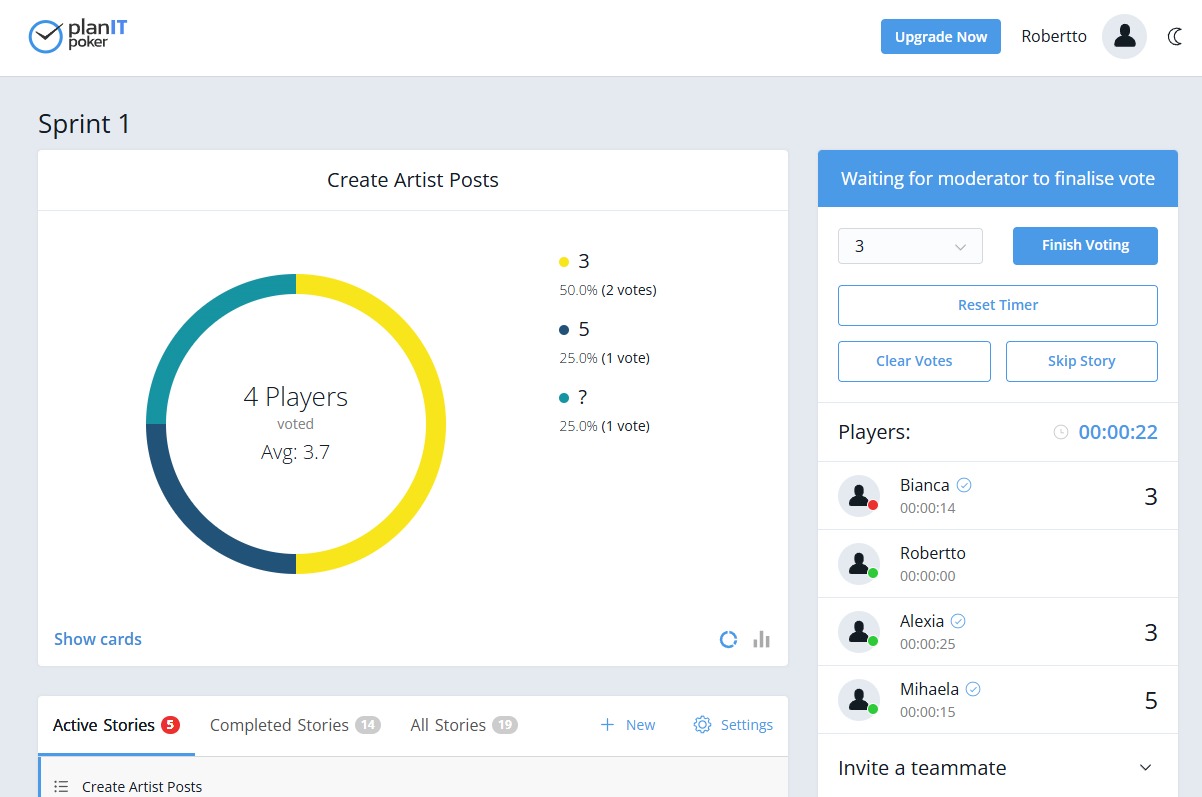




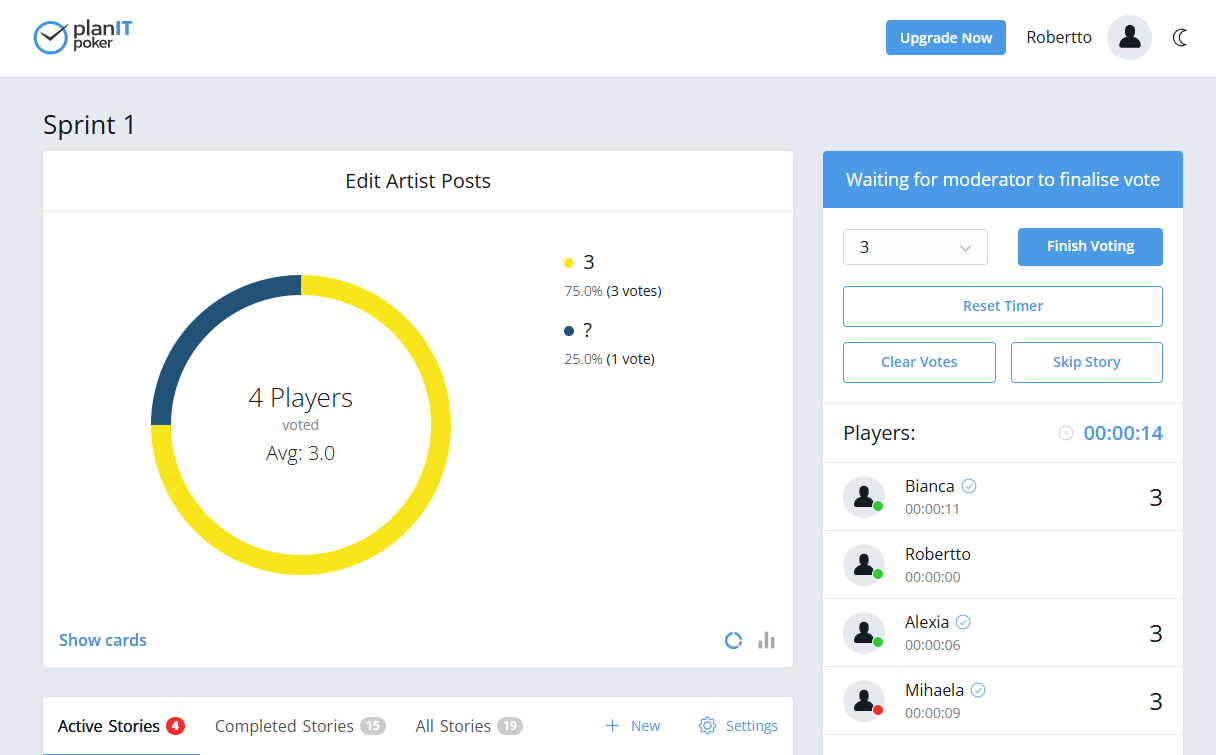




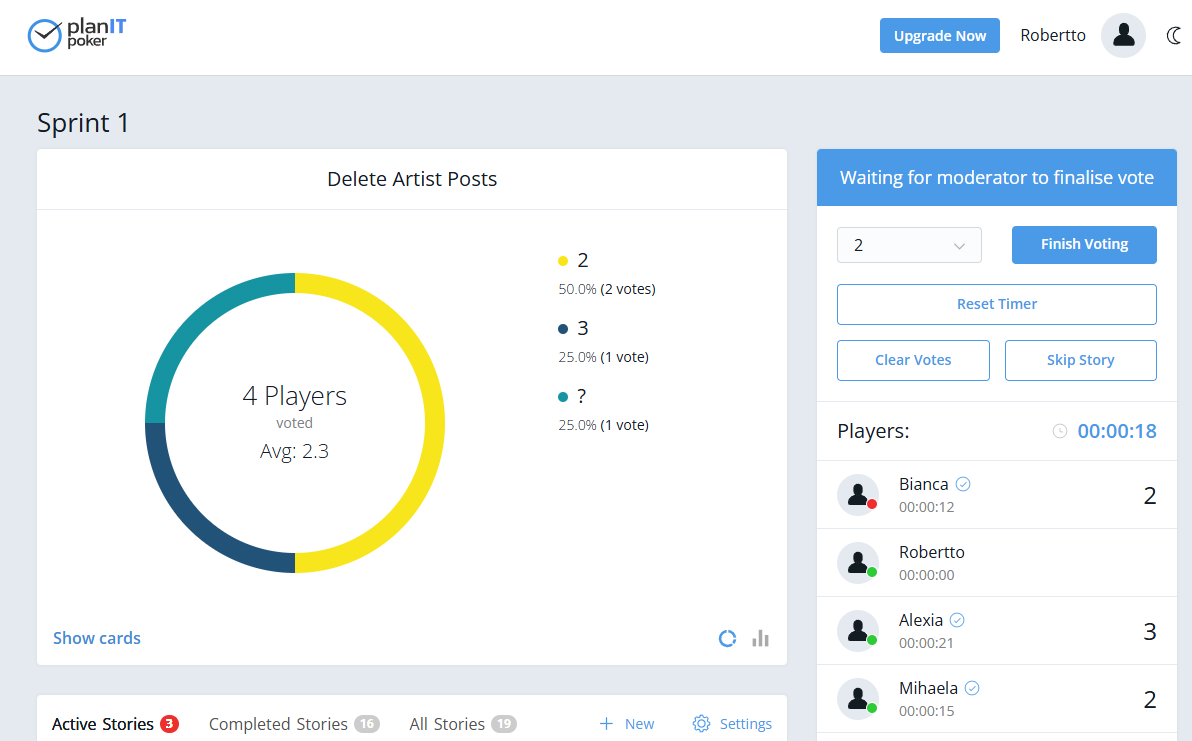


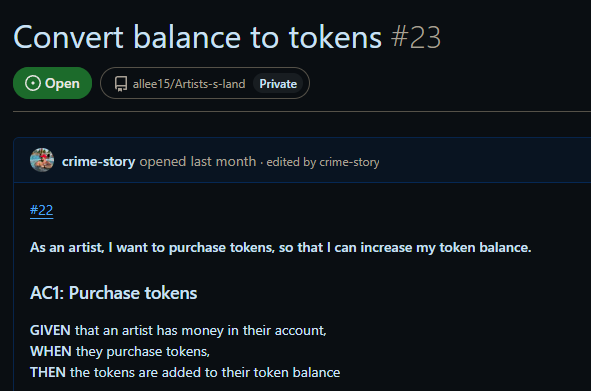


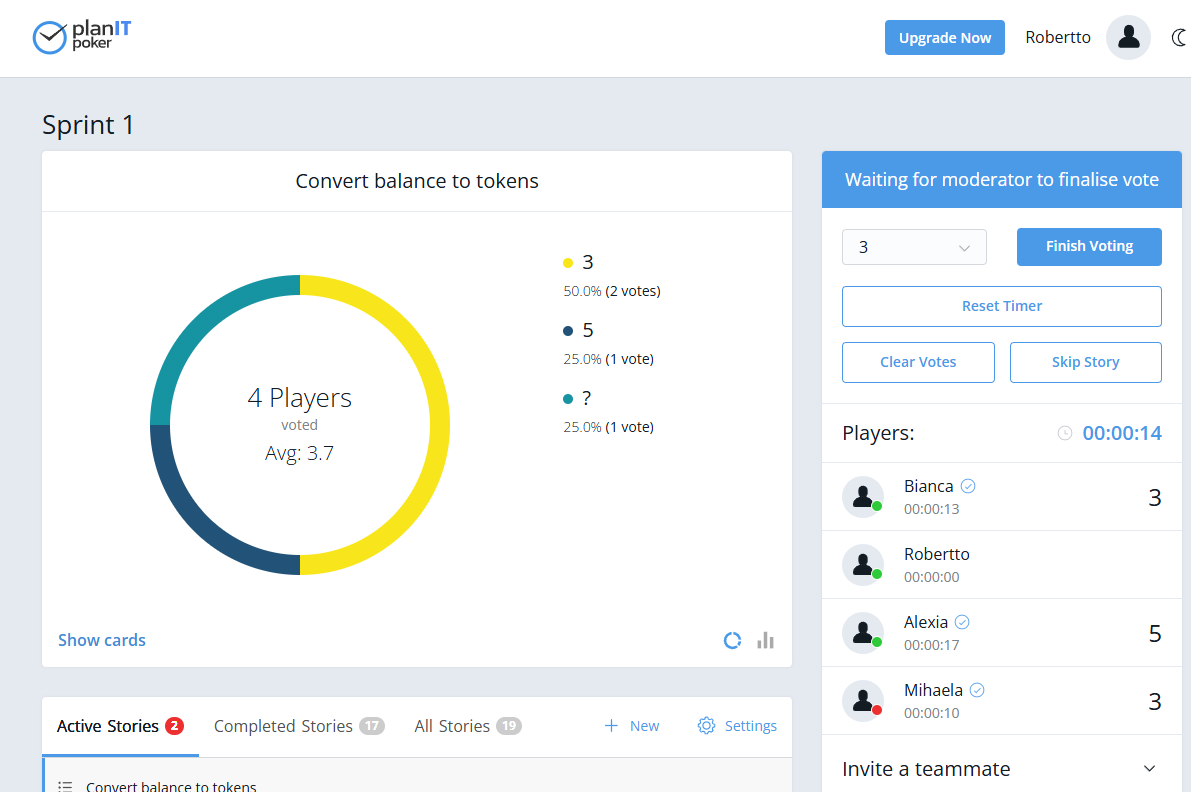


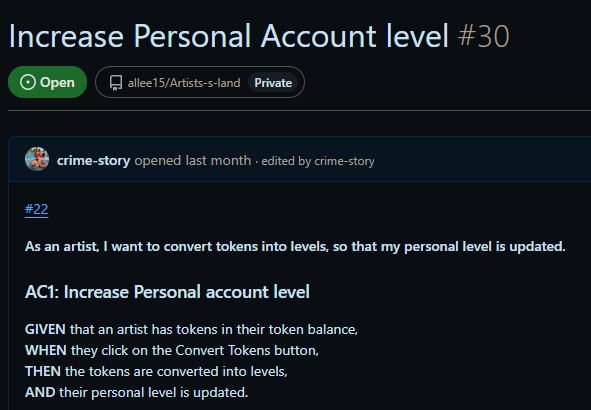


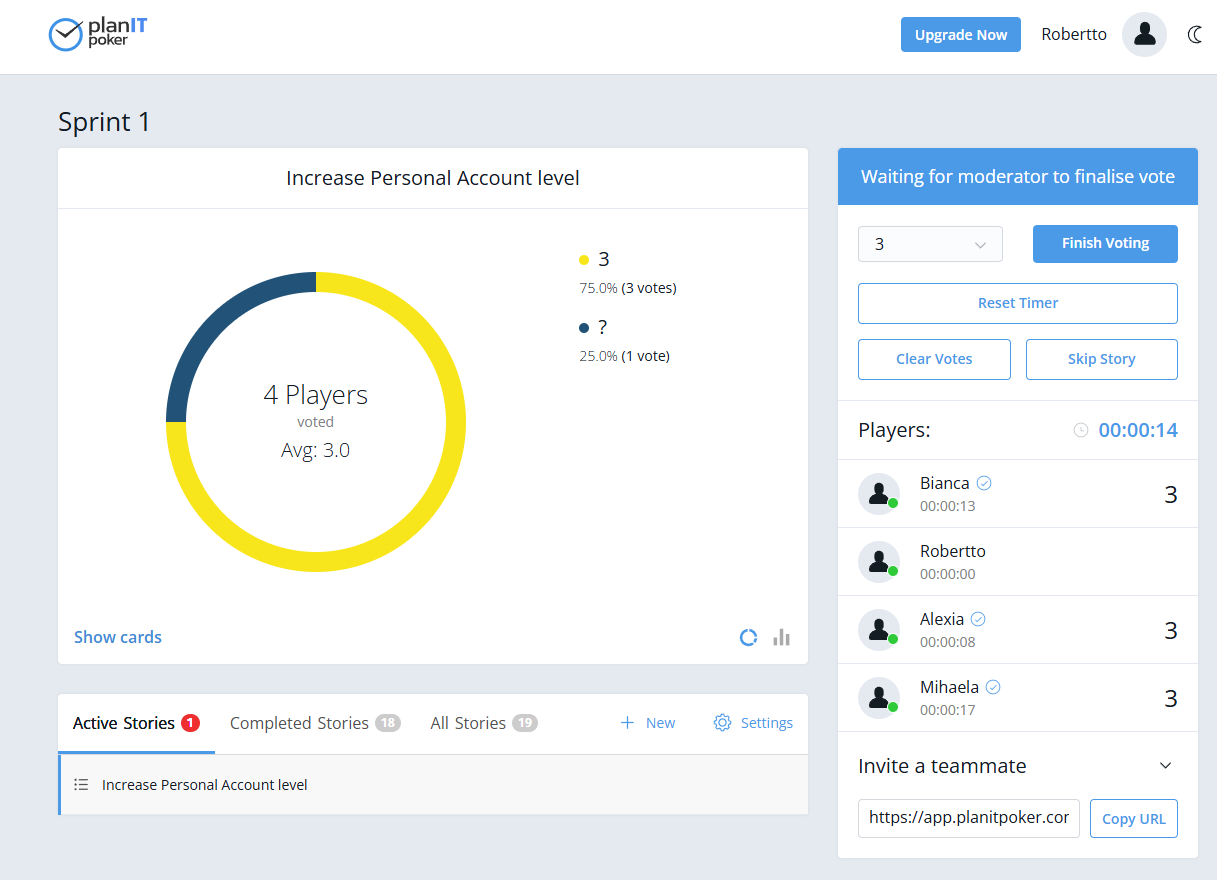






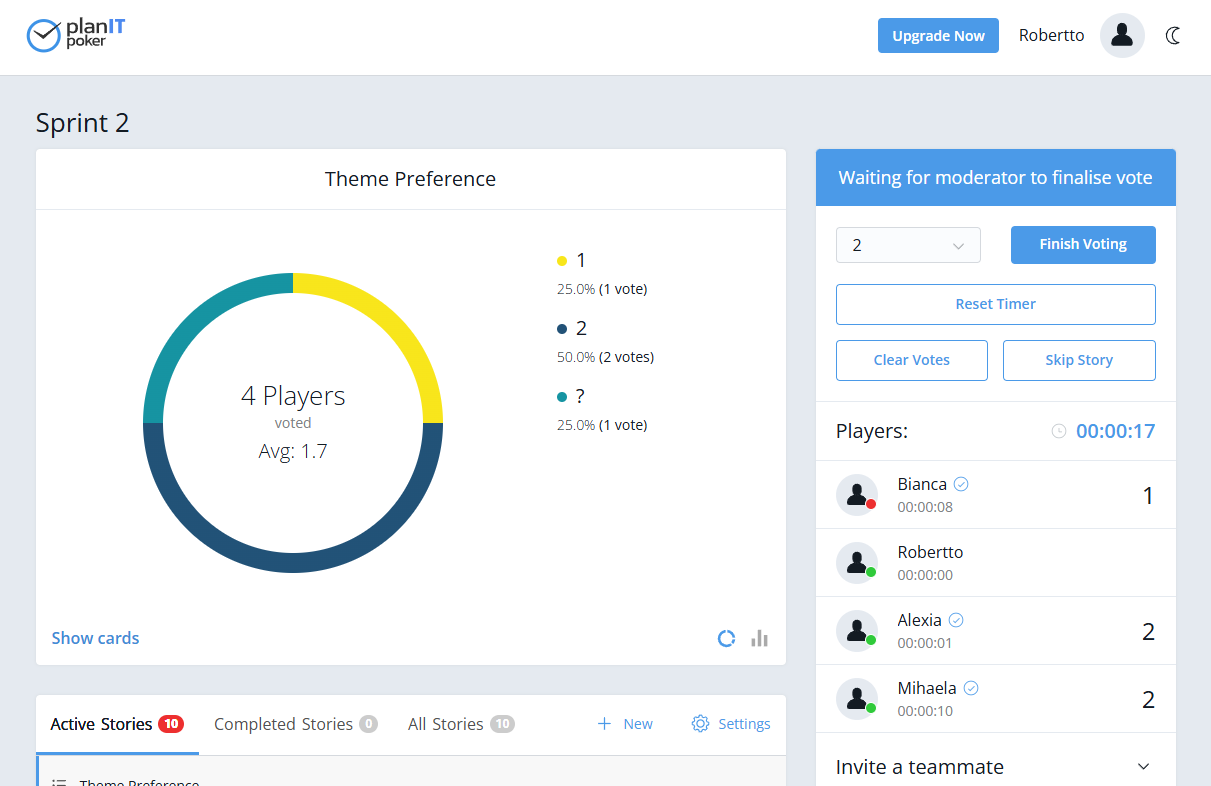


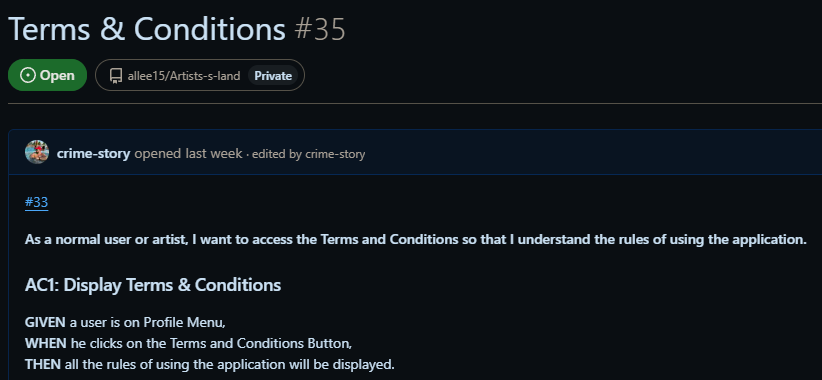




* 1. **Sprint 2**

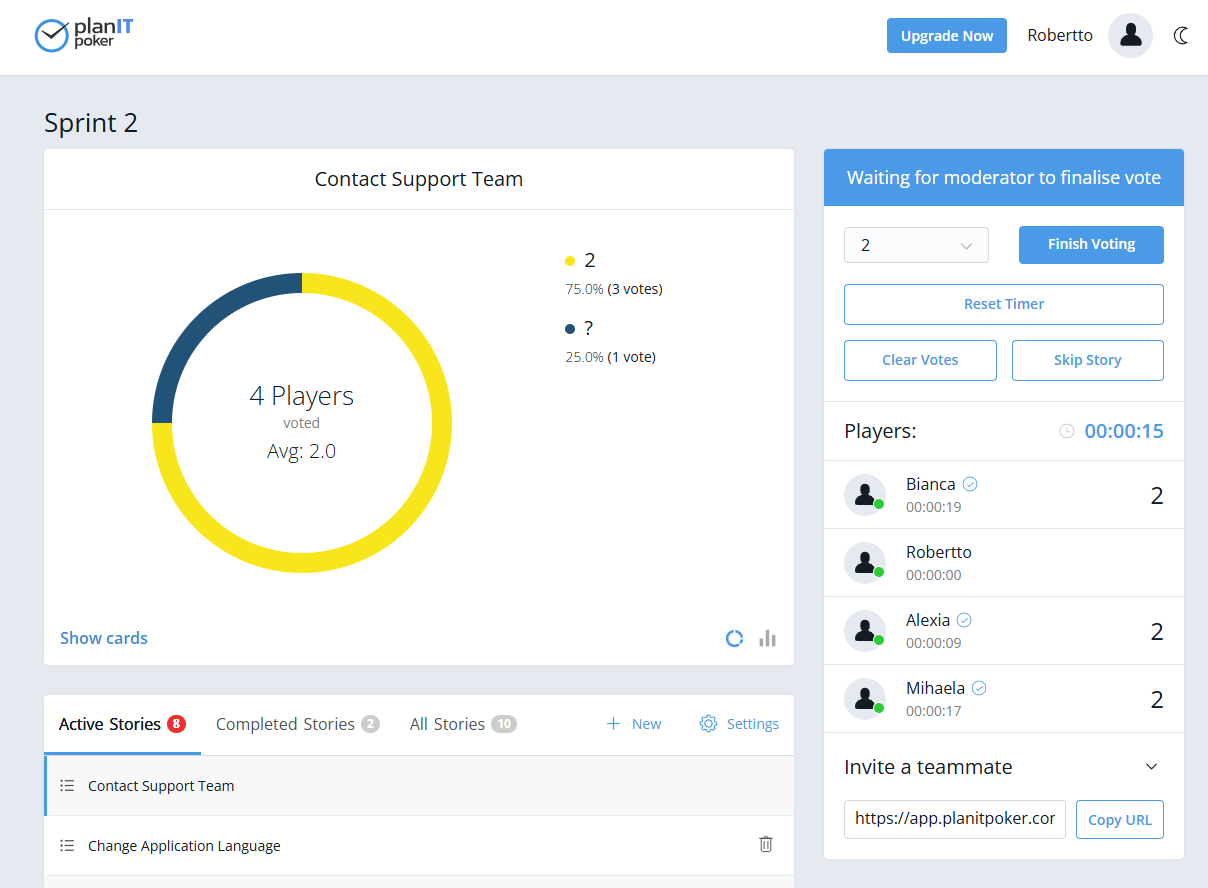


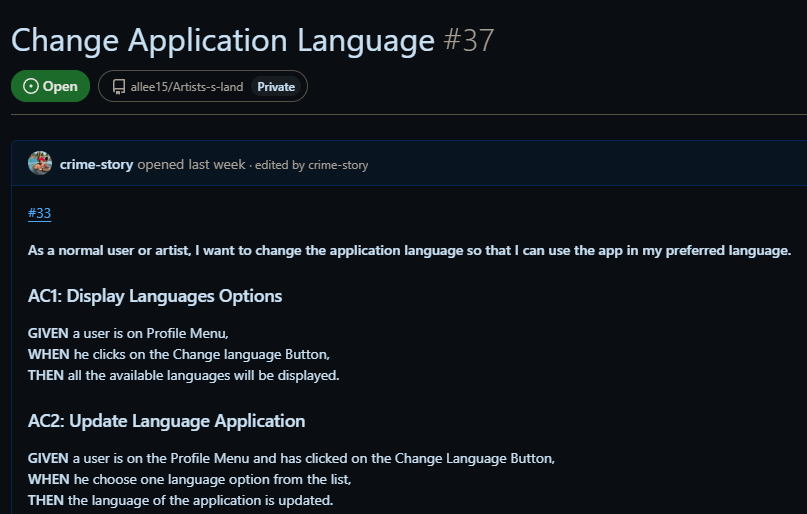


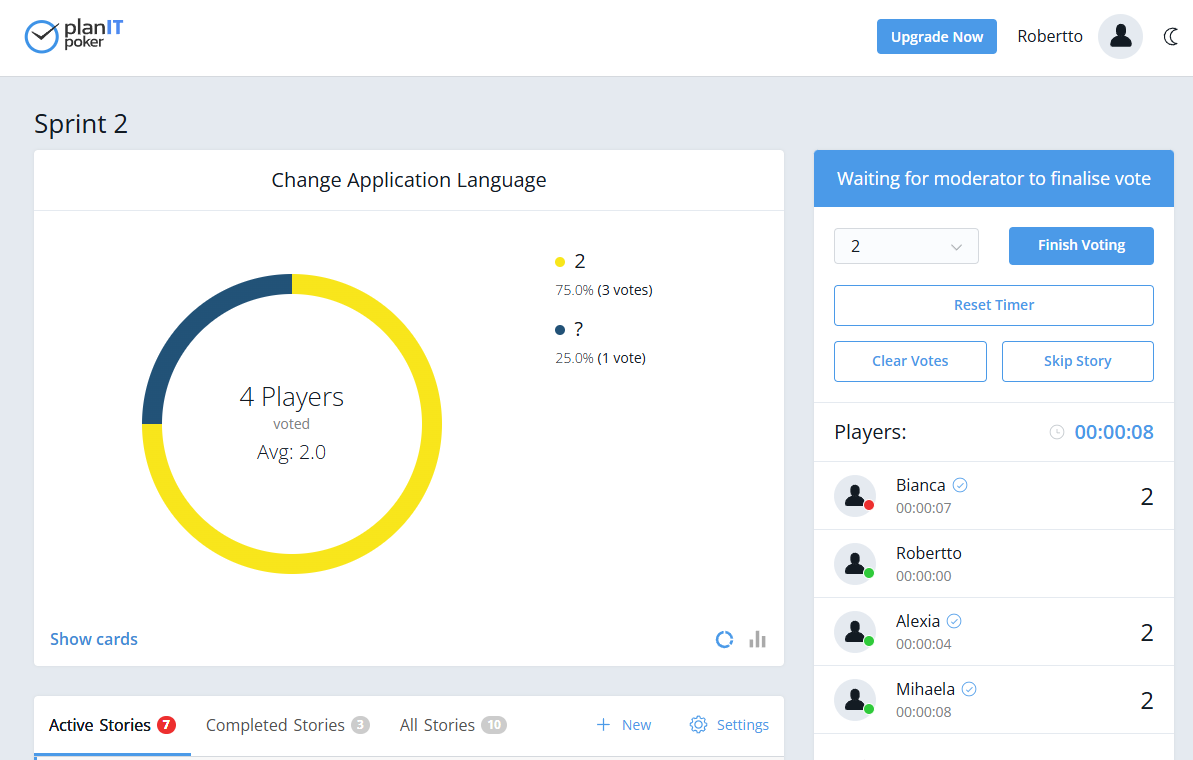


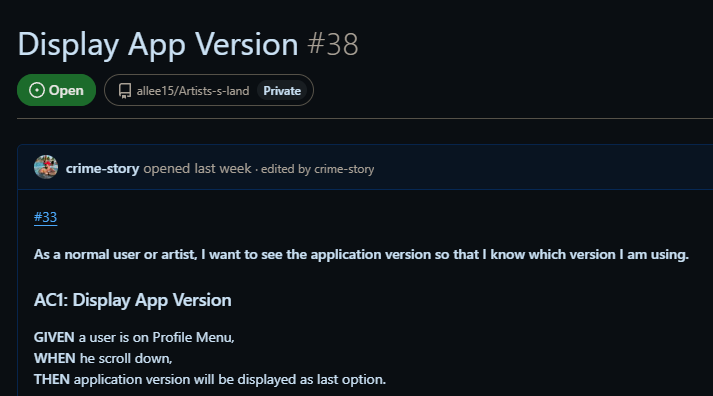


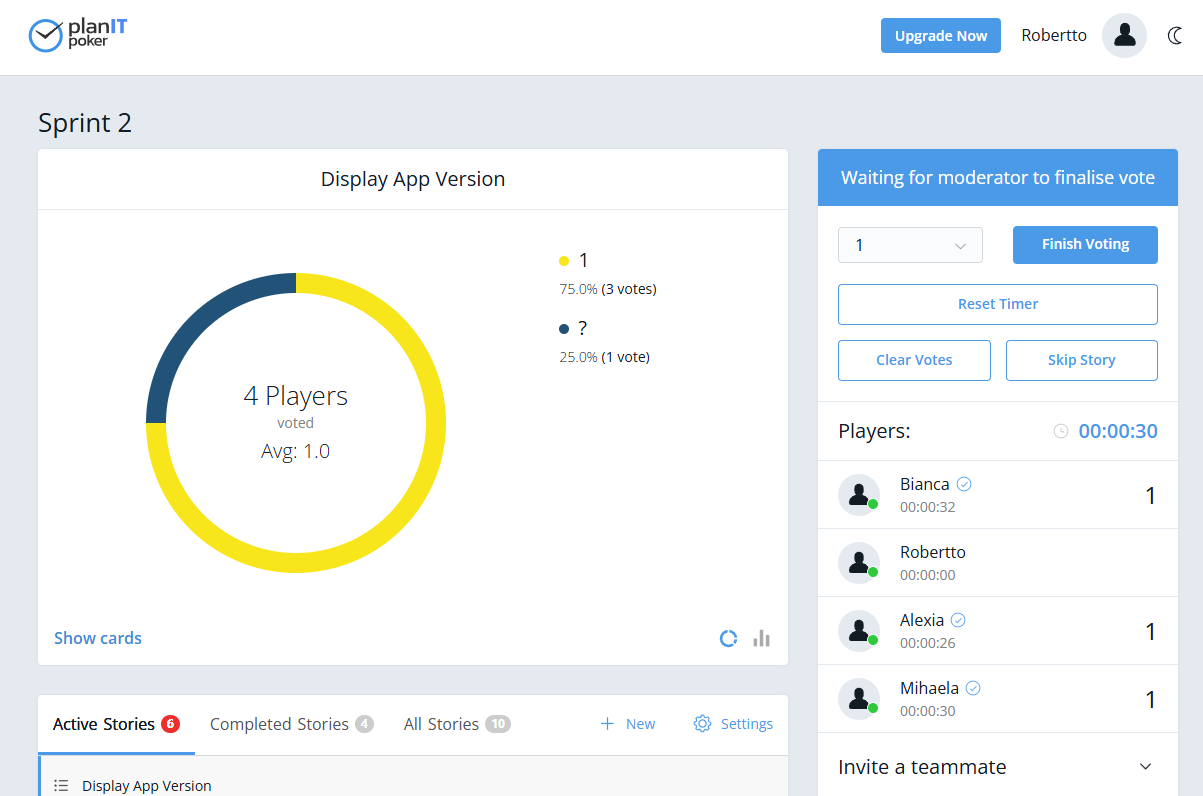


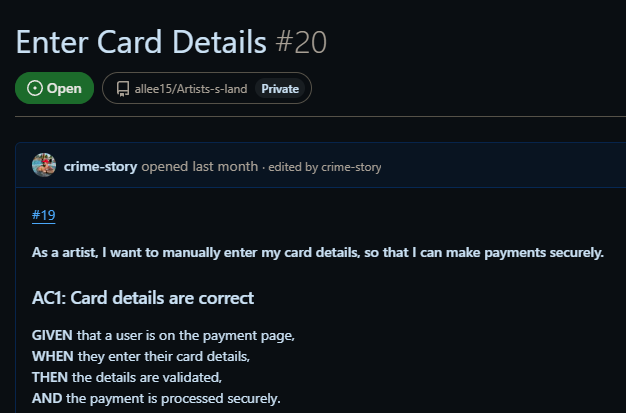


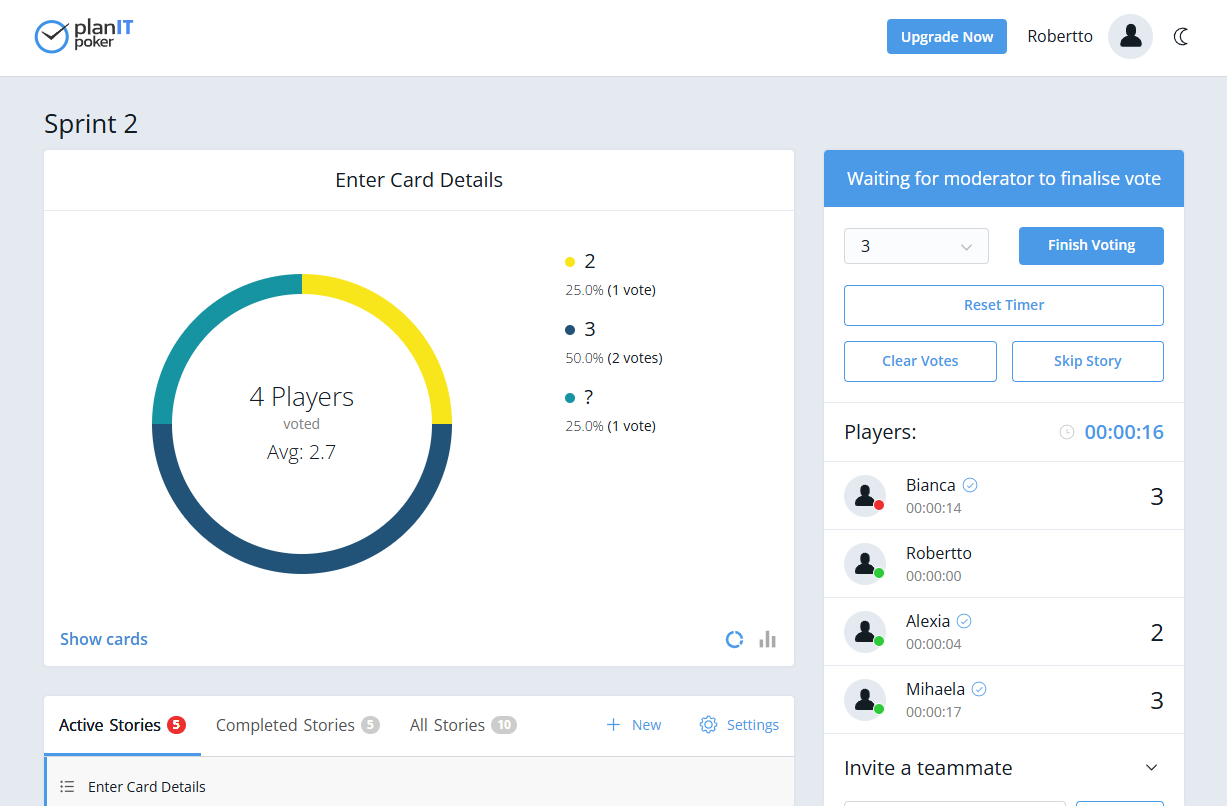


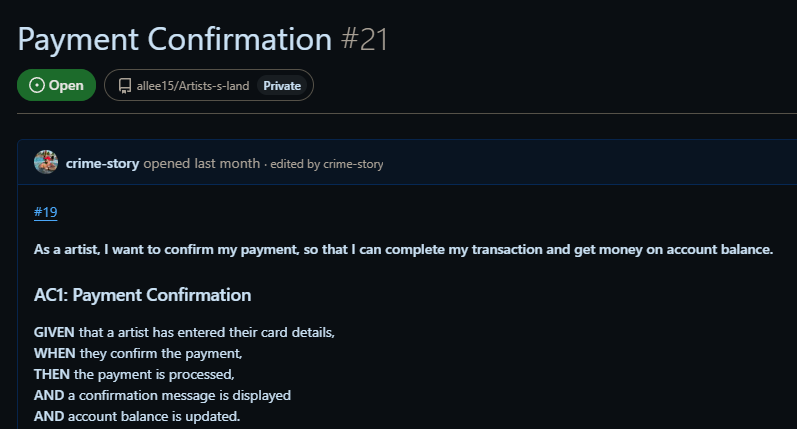




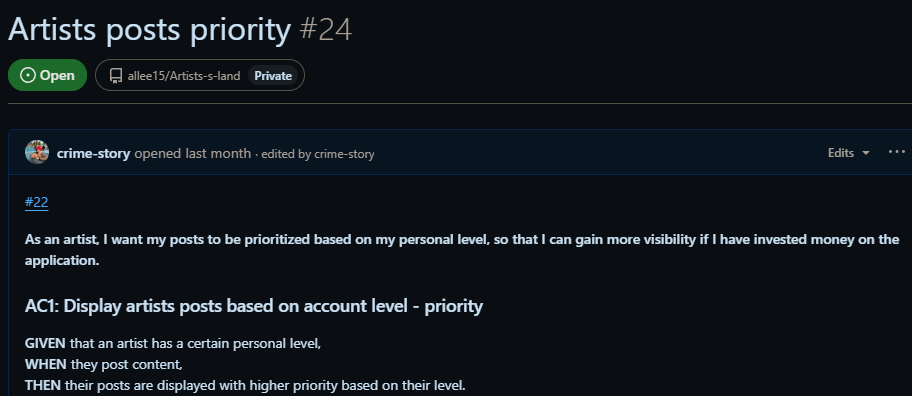


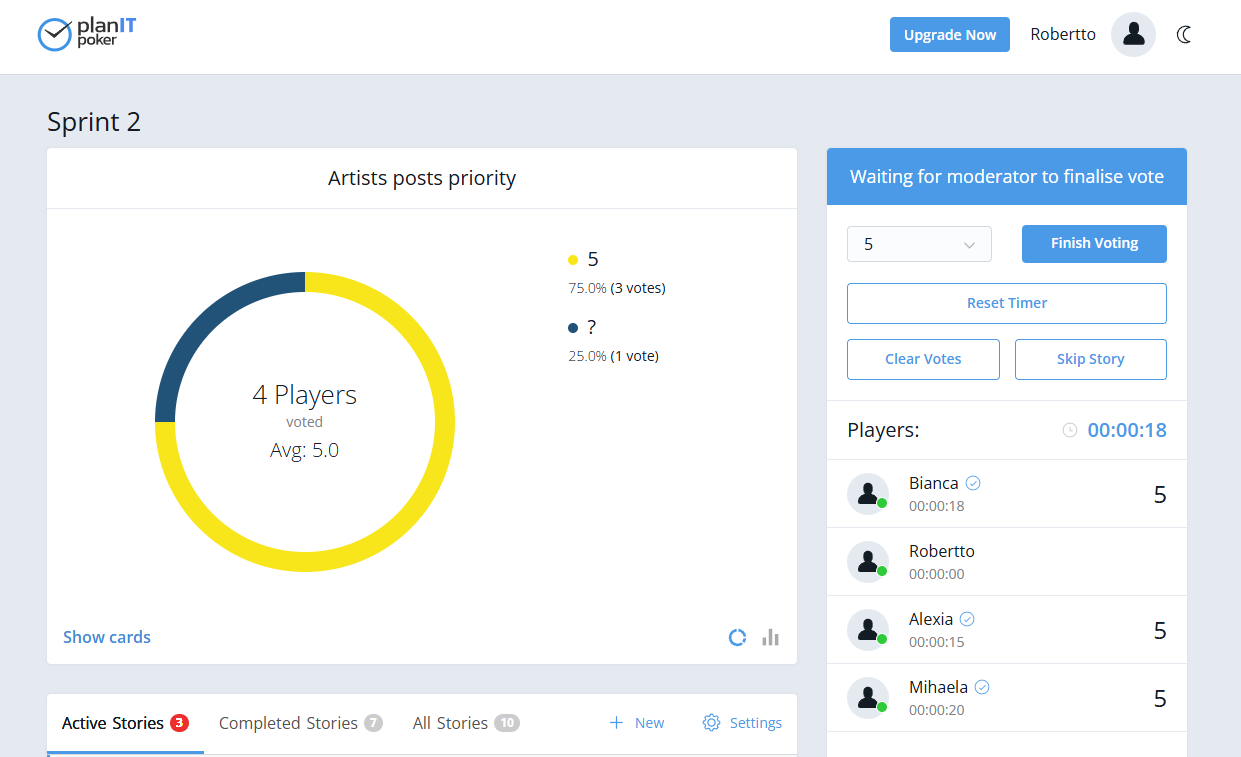




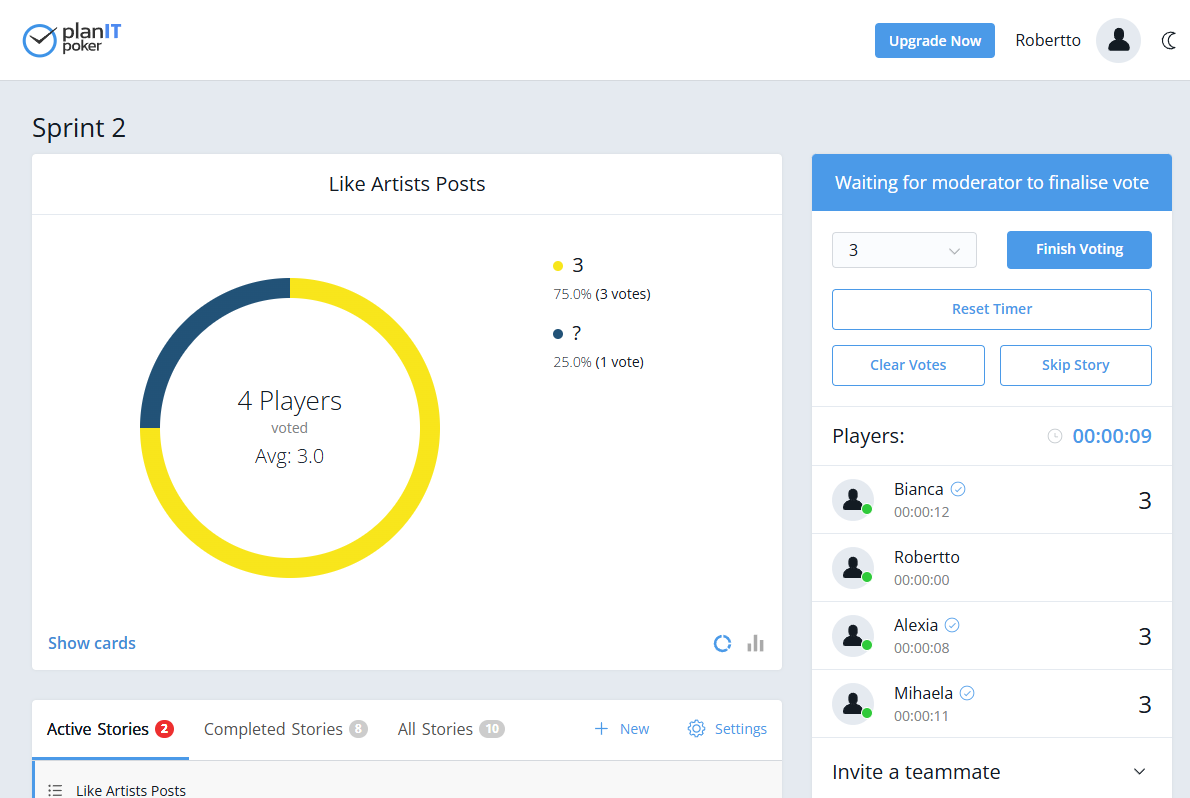




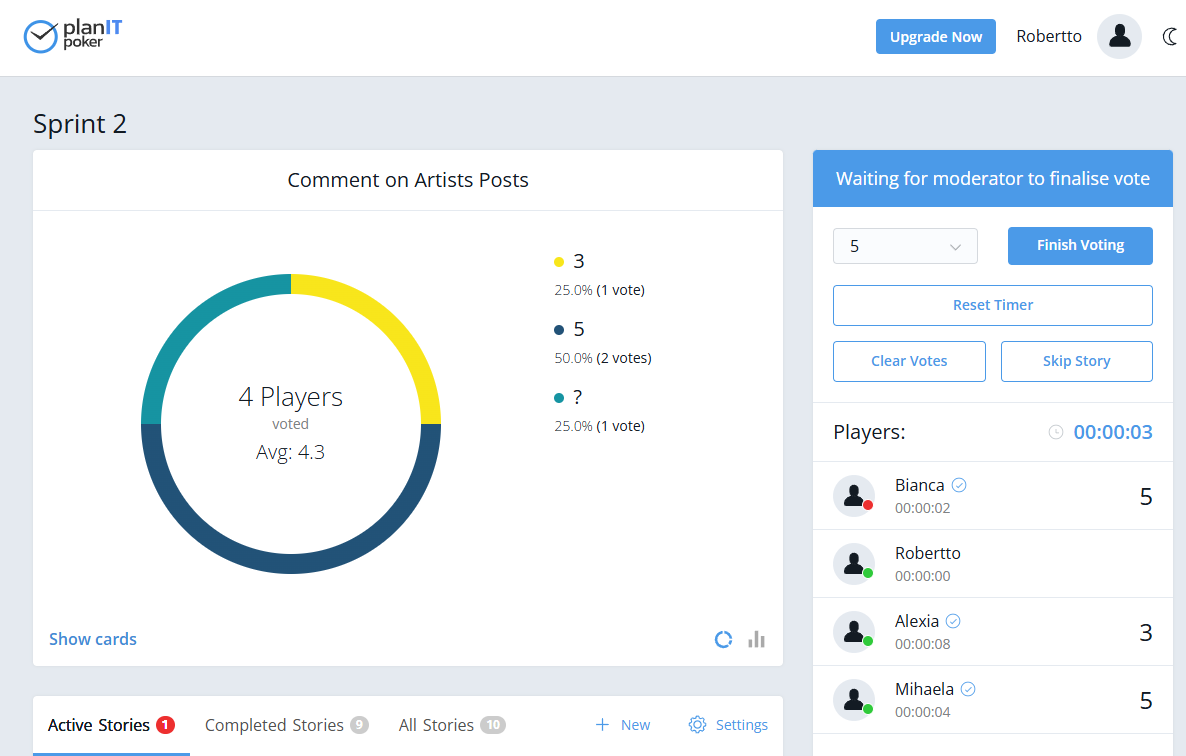




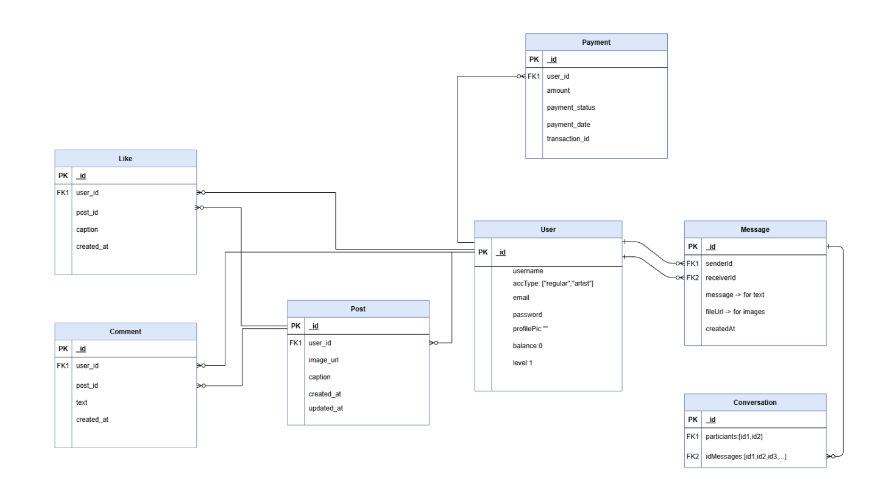








1. **Diagrama bazei de date**



1. **Diagrama Gantt**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Epice** | **25 Octombrie** | **Noiembrie** | **Decembrie** | **Ianuarie** | **15 Februarie** |
| Account creation, management and authentication |  |  |  |  |  |
| Artist Chat |  |  |  |  |  |
| Content viewing |  |  |  |  |  |
| Artist posts creation and further changes |  |  |  |  |  |
| Token System |  |  |  |  |  |
| App Settings  & Information |  |  |  |  |  |
| Payment System |  |  |  |  |  |
| Artists posts interactions |  |  |  |  |  |



**Aldea Alexia** 

**Cristea Bianca Stefania**

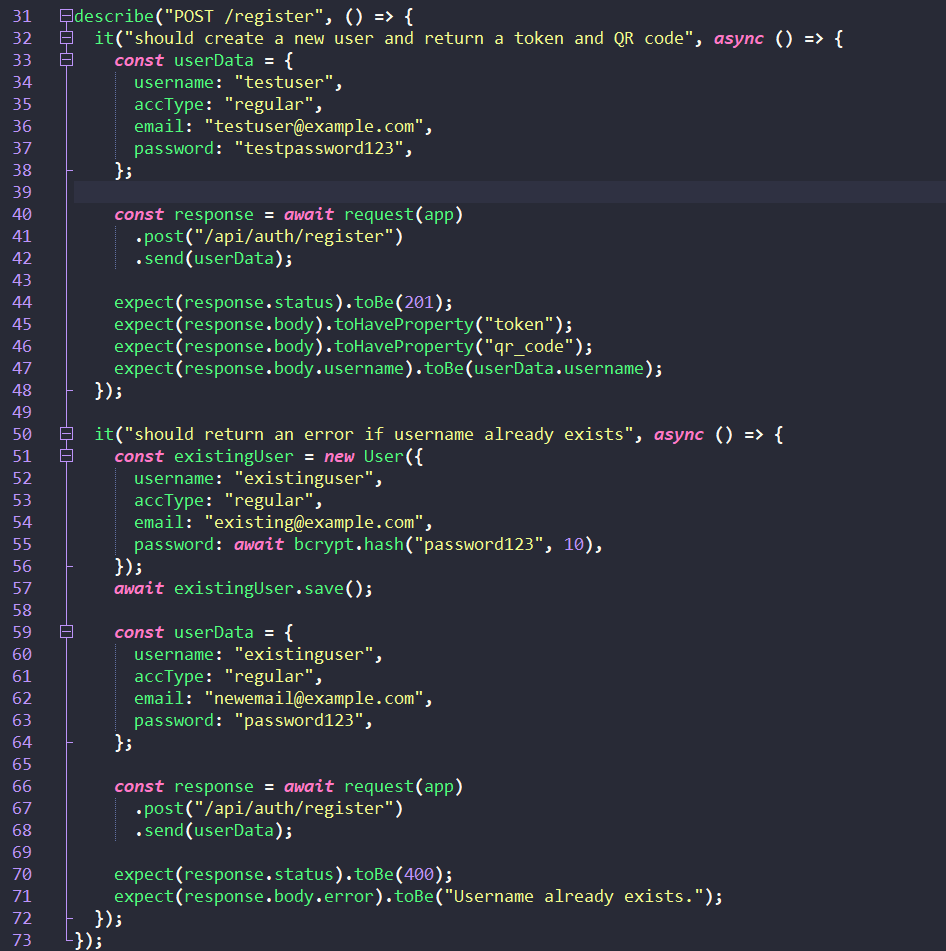
**Agafiței Iuliana Mihaela**

1. **Teste unitare**
   1. **Descrierea testelor unitare**

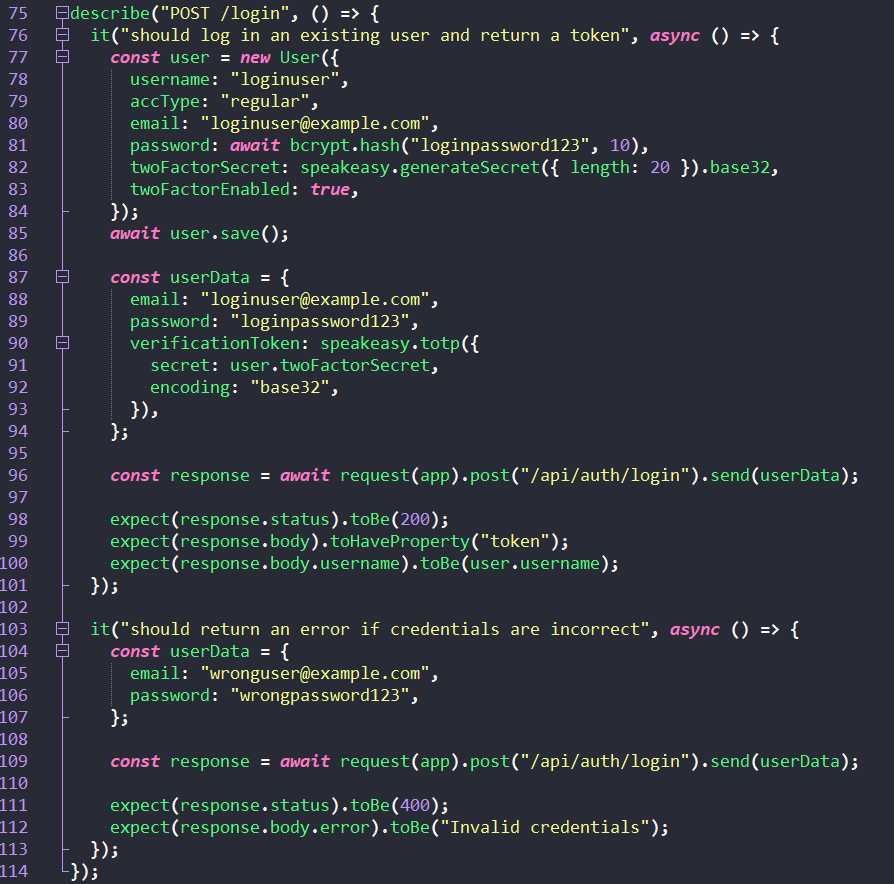
Teste unitare pentru API-ul de autentificare și 2FA

Pentru a asigura calitatea și funcționalitatea corectă a API-ului de autentificare, am utilizat **Supertest** împreună cu **Jest** pentru scrierea și rularea testelor unitare. Testele acoperă următoarele funcționalități critice ale sistemului:

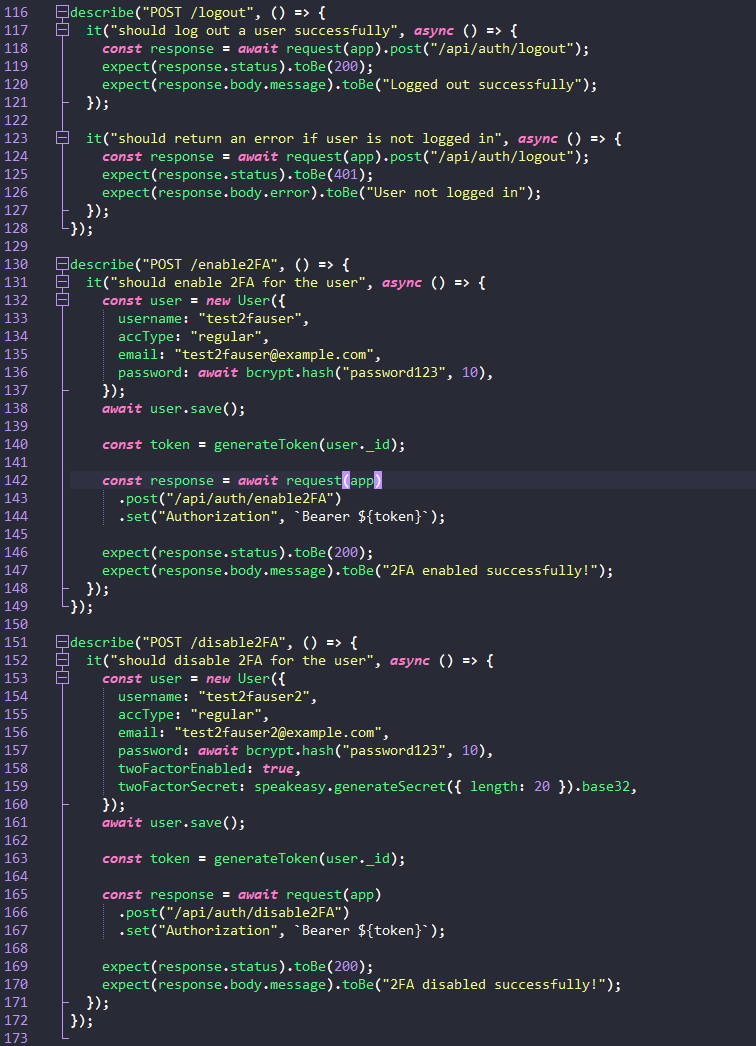
* **Validarea înregistrării unui utilizator nou**: Testarea procesului de creare a unui cont nou, generarea token-ului JWT și a codului QR pentru autentificarea în doi pași (2FA).



* **Autentificarea utilizatorilor existenți**: Verificarea login-ului utilizatorilor, inclusiv inclusiv 2FA.



* **Gestionarea erorilor**: Testarea scenariilor de eroare, cum ar fi utilizator existent la înregistrare sau date incorecte la autentificare.
* Testara activării și dezactivării 2FA, precum și procesul de logout.



Middleware-ul de autorizare

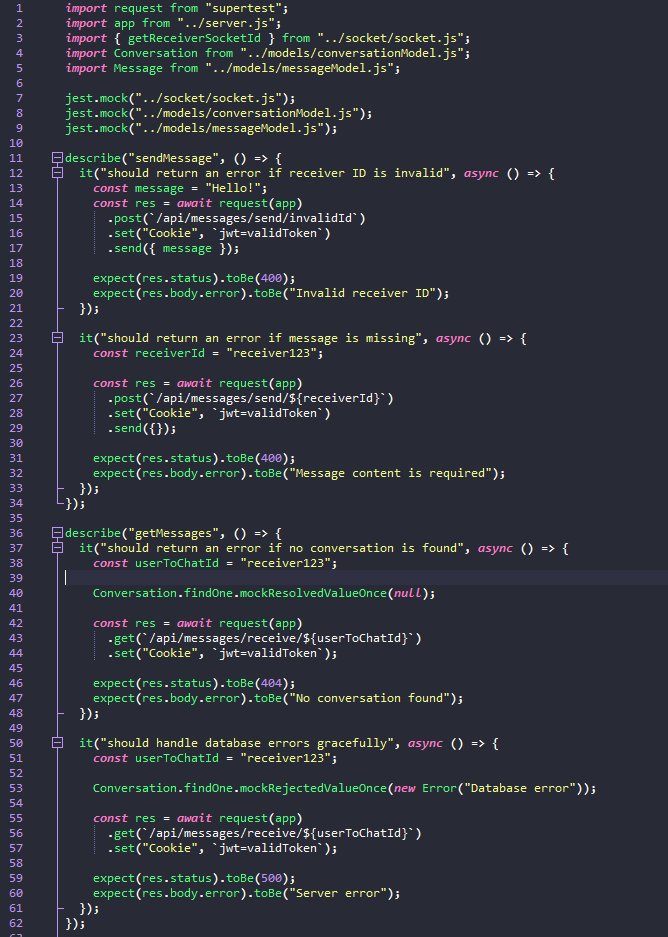
Testele acoperite:

* Răspunsul middleware-ului în cazul lipsei token-ului.
* Validarea token-urilor invalide.
* Autorizarea cu un token valid.



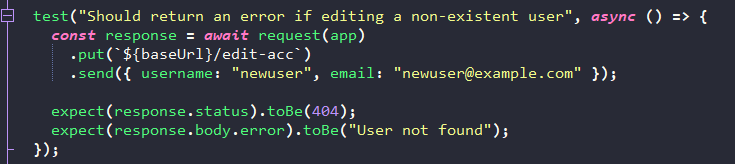
Teste pentru trimiterea mesajelor:

* Validarea ID-urilor invalide ale destinatarilor.
* Validarea mesajelor incomplete.
* Gestionarea erorilor pentru baza de date și cazurile în care nu poate fi găsită conversația.

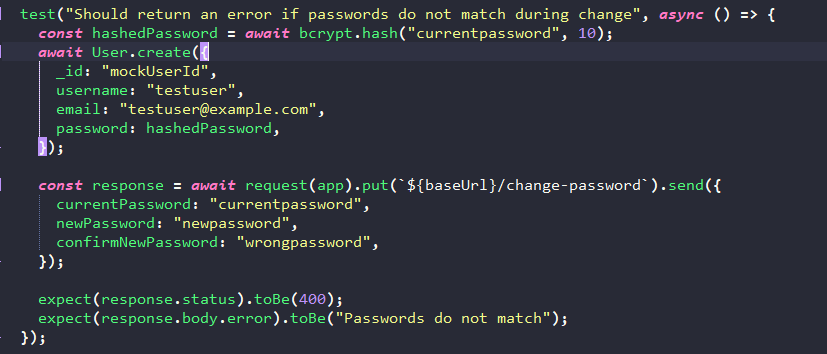
****

Teste pentru User Controller:

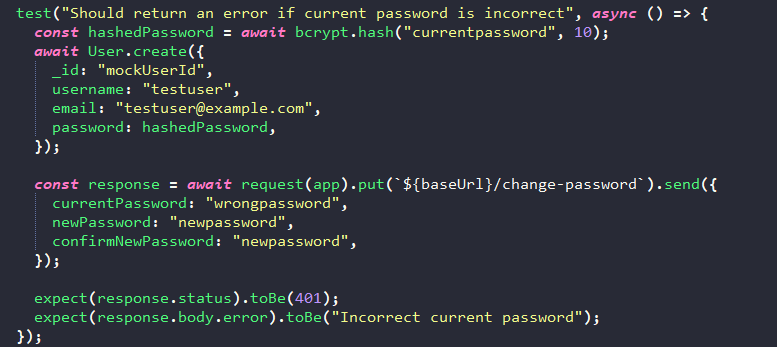
* Schimbarea datelor unui utilizator care nu există.



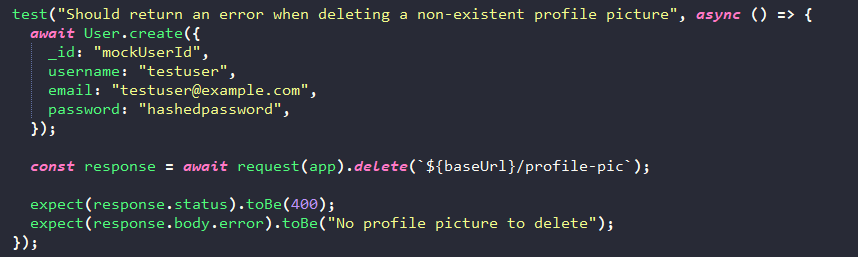
* Schimbarea parolei, dar când se completează confirmarea parolei noi, parolele nu sunt identice.



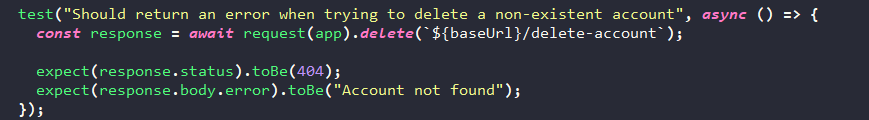
* Schimbarea parolei cu una care este incorectă.



* Ștergerea unei poze de profil care nu există.



* Ștergerea unui cont care nu există.



* 1. **Test Coverage**

****

1. **Test Driven Development (TDD)**
   1. **Avantaje**

* Clean Code & organized: testele au fost scrise înainte de implementarea funcționalităților, pentru a reduce timpul necesar refactorizării pe termen lung și pentru a ne concentra exact pe cerințele impuse, evitând astfel adăugarea unor funcționalități complexe și inutile.
* Prin scrierea testelor înainte de implementarea propriu-zisă, am reușit să evităm erorile care ar fi putut apărea în timpul implementării directe a task-ului. De exemplu, dacă nu am fi scris mai întâi testele pentru autentificare, nu ne-am fi gândit să validăm și formatul standard al adresei de email.
* Design modular: fiecare funcționalitate a fost testată individual prin teste unitare, facilitând astfel o întreținere eficientă a codului pe termen lung.
* Suport implicit pentru noii membri ai echipei: Testele vor servi drept ghid pentru noii programatori, demonstrând modul în care ar trebui să funcționeze aplicația, chiar dacă nu cunosc întregul cod.
* Prin utilizarea testelor automate, ne putem asigura că funcționalitățile aplicației noastre respectă cerințele și că modificările ulterioare aduse codului nu vor introduce erori noi.
  1. **Dezavantaje**
* Timpul de dezvoltare al task-urilor a crescut ușor față de estimarea inițială, deoarece am prioritizat scrierea testelor înainte de implementarea propriu-zisă.
* Procesul de familiarizare cu metodologia Test Driven Development nu a fost ușor la început. A trebuit să învățăm cum să scriem teste eficient și să ne obișnuim cu scrierea testelor înainte de implementarea codului.

1. **Pair Programming vs individual:**
   1. **Avantaje**

* Creșterea calității codului: Fiind două persoane pe același task, potențialele erori care pot apărea sunt găsite mai rapid. Astfel, codul este mai curat și eficient, iar timpul de testare este redus.
* Dacă o persoană are mai puține cunoștințe tehnice, iar celălalt mai multe, se poate face un transfer de cunoștințe între cei doi, pentru îmbunătățirea competențelor tehnice.
  1. **Dezavantaje**
* Costuri mai mari de resurse: Dacă două persoane stau împreună pe același task, în unele cazuri poate duce la un consum mai mare de timp și resurse în comparație cu munca individuală, ceea nu este întotdeauna rentabil pentru task-urile simple.
* Ambele persoane care lucrează pe task-ul respectiv trebuie să se concentreze în permanență, să analizeze împreună înainte deciziile, să facă un schimb de idei, ceea ce poate duce la o epuizare psihică.

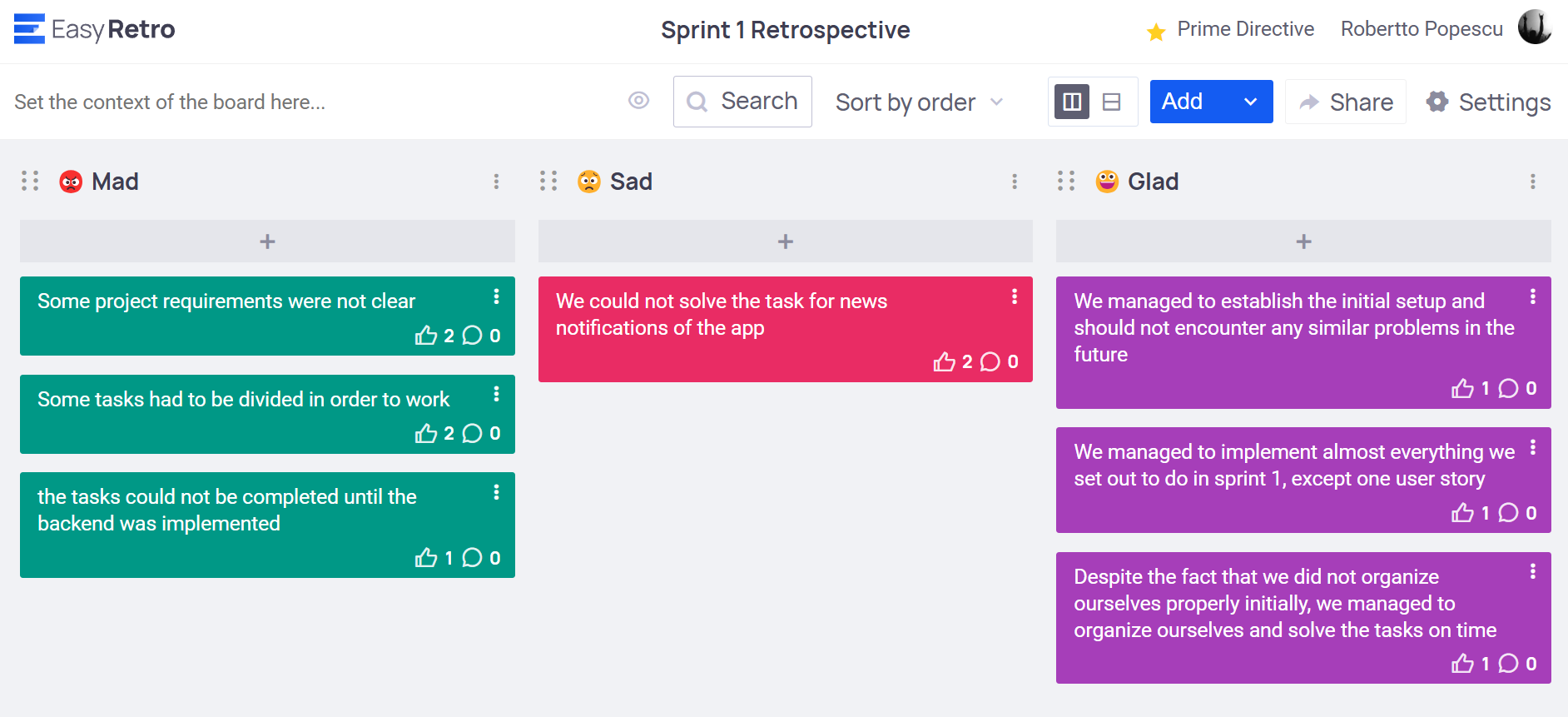
1. **Sprint Daily Standups**
   1. **Sprint 1**

* **Probleme de organizare**: În timpul primului Sprint, am întâmpinat probleme de organizare, deoarece cerințele proiectului nu au fost clare pentru toți membrii echipei de la început sau au fost modificate ulterior. De asemenea, au apărut dependențe între membrii echipei, deoarece task-urile includeau cerințe atât de backend, cât și de mobile, iar inițial ne-am organizat cu o persoană pentru frontend și două pentru backend (task-urile nu pot fi validate ca fiind DONE, dacă doar Mobile-ul sau Backend-ul este finalizat).
* **Comunicare deficitară între membrii echipei:** O altă problemă a fost comunicarea deficitară între noi, ceea ce a dus la întârzieri în livrarea unor funcționalități.
* **Probleme tehnice:** Funcționalitate de a trimite notificări despre noutățile din aplicație a fost abandonată, datorită unor probleme tehnice cu baza de date.
  1. **Sprint 2**
* În cadrul celui de-al doilea Sprint, ca echipa am planificat următoarele activități principale: implementarea funcționalităților de payment, interacțiunea cu postările artiștilor și afișarea informațiilor despre aplicație.
* **Rolul dezvoltatorului Mihaela**: Ne așteptam ca colega noastră Mihaela, având rolul de dezvoltator, să contribuie la implementarea acestor funcționalități. Cu toate acestea, din cauza unor constrângeri personale și tehnice, ea nu a putut să intervină activ în realizarea task-urilor alocate.
* **Redistribuirea sarcinilor**: Ceilalți doi dezvoltatori, Alexia și Bianca, au preluat task-urile care inițial erau alocate Mihaelei. Astfel, au fost nevoite să-și reorganizeze prioritățile pentru a asigura livrarea funcționalităților planificate, fără întârzieri.
* **Implementarea sistemului de payment**: Aceasta a durat mai mult decât era estimat inițial, întrucât au apărut probleme tehnice neprevăzute. Aceste probleme au necesitat soluții suplimentare care au prelungit timpul de finalizare.

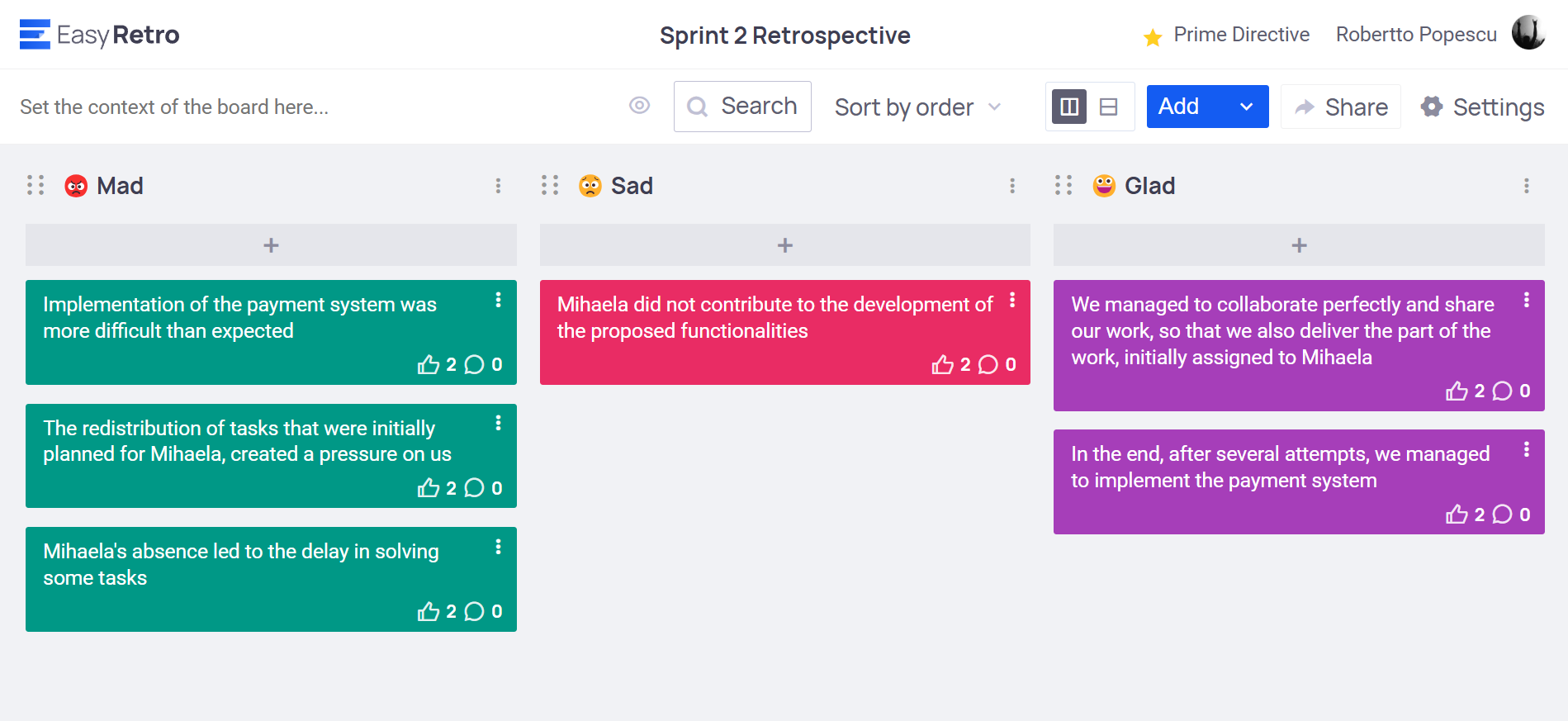
1. **Retrospectivă**

Pentru a realiza retrospectivele aferente Sprinturilor 1 și 2, am utilizat tool-ul online **EasyRetro**, care ne-a oferit o platformă intuitivă și eficientă pentru organizarea și analizarea feedback-ului echipei.

* 1. **Sprint 1**

****

* 1. **Sprint 2**

****

1. **Raport muncă echipă**

* Popescu Paullo Robertto Karloss - Product Owner
  + Contribuție: Asigurarea cadrului de lucru pentru întregul proiect, prin redactarea documentației și definirea cerințelor funcționale și tehnice necesare dezvoltării aplicației.
    - Redactarea documentației proiectului:
      * Elaborarea scenariului principal ce descrie cerințele sistemului (Requirements Document).
      * Derivarea și rafinarea celor 15+ User Stories inițiale în conformitate cu principiile INVEST (Independent, Negotiable, Valuable, Estimable, Small, Testable).
      * Justificarea prioritizării User Stories și contribuția la crearea backlog-ului proiectului.
    - Diagrame și structură tehnică:
      * Crearea **Use Case Diagram**, care ilustrează cazurile de utilizare ale aplicației.
      * Elaborarea diagramei procesului de autentificare și a fluxului general de utilizare.
      * Dezvoltarea unei **diagrame Gantt** pentru planificarea și monitorizarea progresului proiectului.
    - Test-Driven Development (TDD): Am realizat o analiză detaliată a avantajelor și dezavantajelor TDD față de dezvoltarea tradițională, evidențiind:
      * Beneficii precum creșterea calității codului, detecția timpurie a erorilor și claritatea cerințelor.
      * Limitări, cum ar fi timpul suplimentar necesar pentru scrierea testelor și dificultățile în adoptarea metodei de către dezvoltatori cu experiență limitată în TDD.
    - Pair Programming: Am evaluat avantajele și dezavantajele acestei tehnici de colaborare, subliniind:
      * Beneficii precum îmbunătățirea calității codului, transferul de cunoștințe între colegi și creșterea productivității în echipă.
      * Dezavantaje, inclusiv necesitatea de resurse suplimentare și posibile diferențe de stil de lucru între programatori.
    - Validarea implementărilor:
      * Participarea activă la **Planning Meeting** pentru a asigura alinierea obiectivelor echipei la cerințele proiectului.
      * Validarea funcționalităților dezvoltate de echipă în cadrul Review Meetings, cu identificarea potențialelor îmbunătățiri.
    - Meetinguri de retrospective:
      * Facilitarea **meetingurilor de retrospective** la finalul fiecărui sprint, utilizând **EasyRetro**, pentru a reflecta asupra progresului echipei și a identifica soluții pentru provocările întâmpinate.
    - Colaborare cu echipa:
      * Asigurarea clarității și detaliilor cerințelor pentru dezvoltatori, facilitând comunicarea între membrii echipei.
      * Utilizarea platformei de project management pentru urmărirea progresului și rezolvarea impedimentelor. **(Github projects)**
  + Procent din proiect: 30%
* Cristea Bianca - Scrum Master
  + Contribuție: diagrame
    - Workflow diagram, diagrama bazei de date
    - Ajutor activ în realizarea documentației
    - Ajutor în crearea principalelor task-uri
  + Contribuție: Realizarea backend-ului.
    - Crearea și configurarea proiectului de ExpressNode.js cu toate pachetele și fișierele de început necesare
    - Sistem complet de autentificare, 2FA
    - Resetare parola, adăugare/editare poza de profil, editare email/username, ștergere cont
    - Căutarea utilizatorilor în aplicație
    - Sistem complet de chat -> crearea unei conversații (chat), trimitere mesaje/poze
  + Procent din proiect: 35%
* Aldea Alexia (dezvoltator)
  + Contribuție: Integrare completă în aplicația mobilă pentru platforma iOS
    - Ajutor activ în realizarea documentației
    - Ajutor în crearea principalelor task-uri
    - Toate rutele create in backend, implementate în aplicația mobilă
    - Interfața utilizator, toate funcționalitățile aplicației
    - Crearea și configurarea proiectului de XCode cu toate pachetele și fișierele de început necesare
    - Asigurarea unui produs complet funcțional, ce prezintă un comportament adecvat unei aplicații mobile
  + Contribuție: Backend
    - Integrare completa Stripe pentru plăti
    - Postări -> regula de afișare a acestora în pagina principala (Home -> în funcție de nivel), adăugare comentariu/apreciere (like/unlike), șterge/raportare postare;
    - Funcționalități pentru pagina artistului: adăugare/ștergere postare, adăugare comentariu/apreciere (like/unlike)
  + Procent din proiect: 35%