

Platformă pentru boardgame meetups

Bahrim Dragoş 406
Tache-Ioan Radu 406

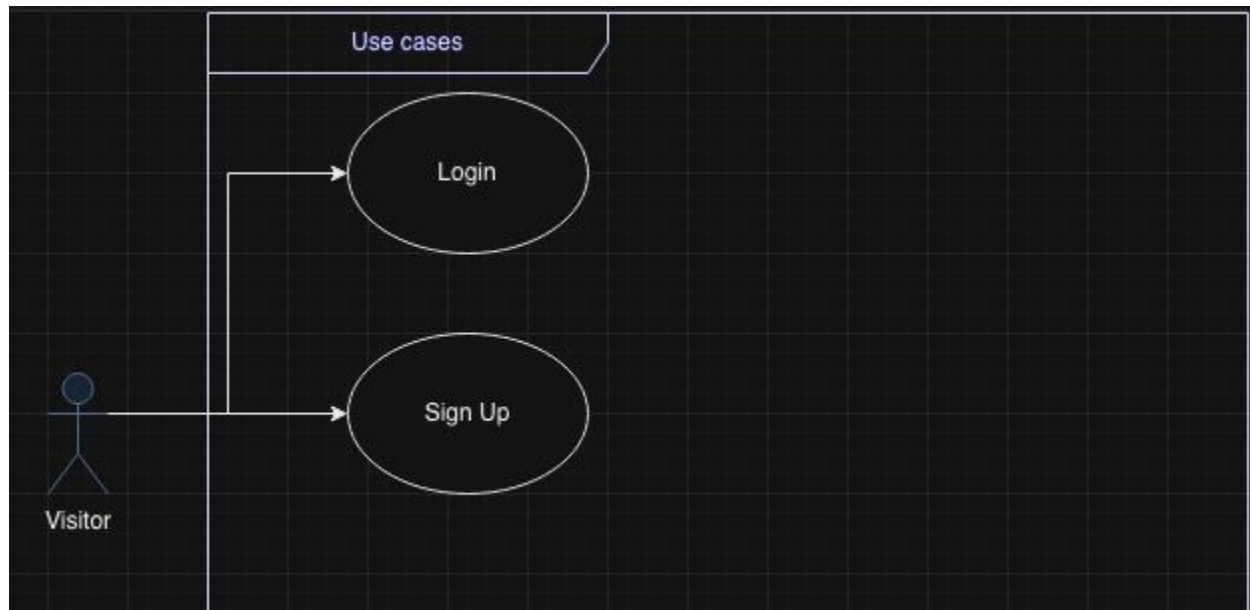
Cerință

Utilizatorii pot introduce locul de desfășurare a evenimentului de boardgame, specifica jocurile care sunt disponibile pentru a fi jucate și data și ora preferate. Utilizatorii se pot înregistra la evenimente, pot exprima preferințe pentru jocurile ce vor fi jucate și propune jocuri noi. Când un eveniment a adunat numărul necesar de jucători pentru unul sau mai multe jocuri, utilizatorul care a inițiat evenimentul poate alege un joc și anunța astfel decizia utilizatorii care s-au înregistrat (fie că au ales acel joc sau nu). Utilizatorii trebuie să facă checkin pentru a își valida participarea la eveniment. Aplicația/platforma ar trebui să aibă o bază de date cu jocuri și regulile lor (număr de participanți, reguli detaliate, etc.). Jucătorii pot câștiga badge-uri pentru fiecare joc (e.g. știe regulile, a jucat atâtea jocuri, helper, etc.). Utilizatorii pot alege cel mai bun jucător al unui eveniment și să raporteze comportamentul altor jucători (corect, nu știe să piardă, agresiv etc.).

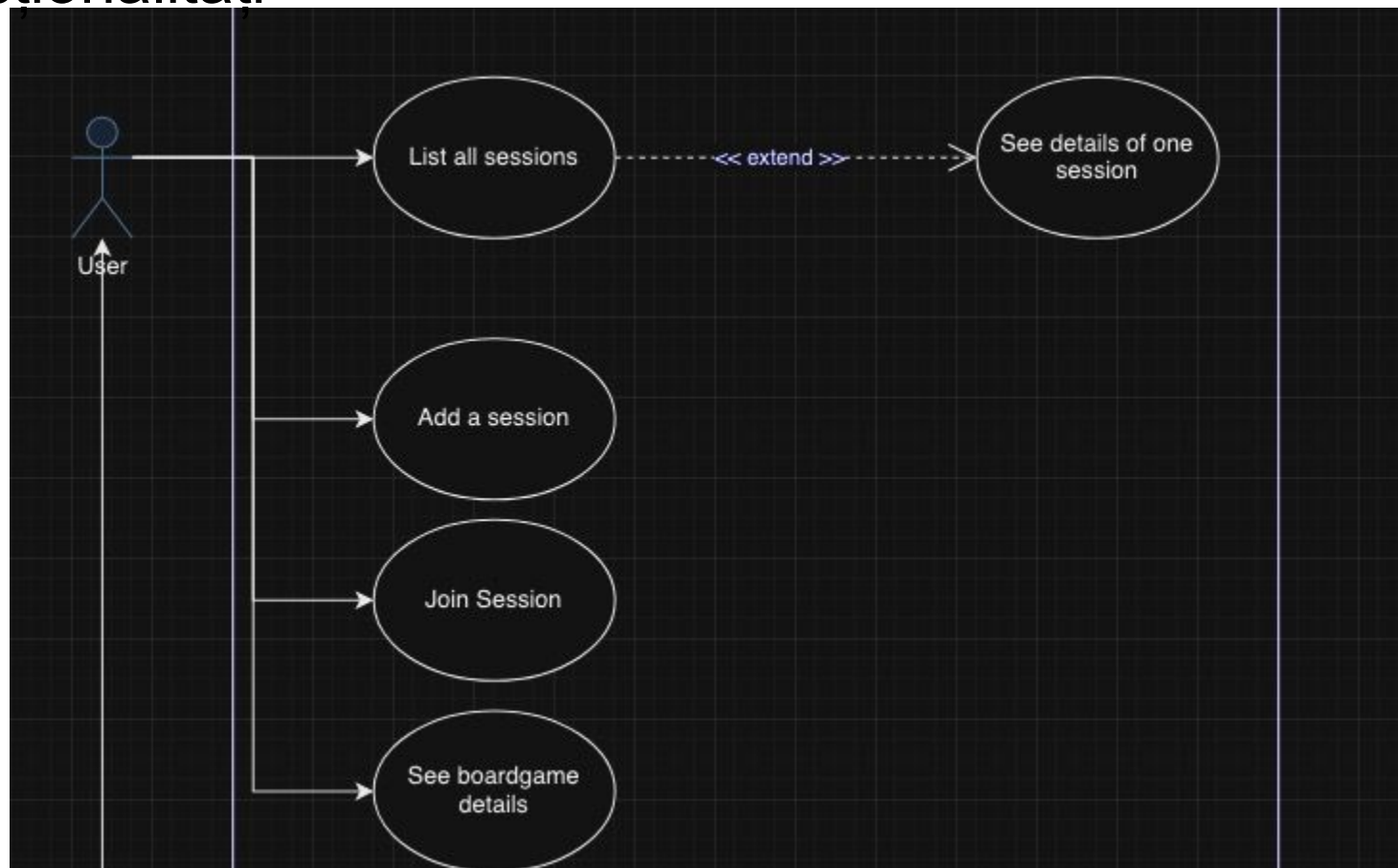
Componente

- aplicație în Flutter ce prioritizează platformele mobile
- server în Java folosind Spring Boot

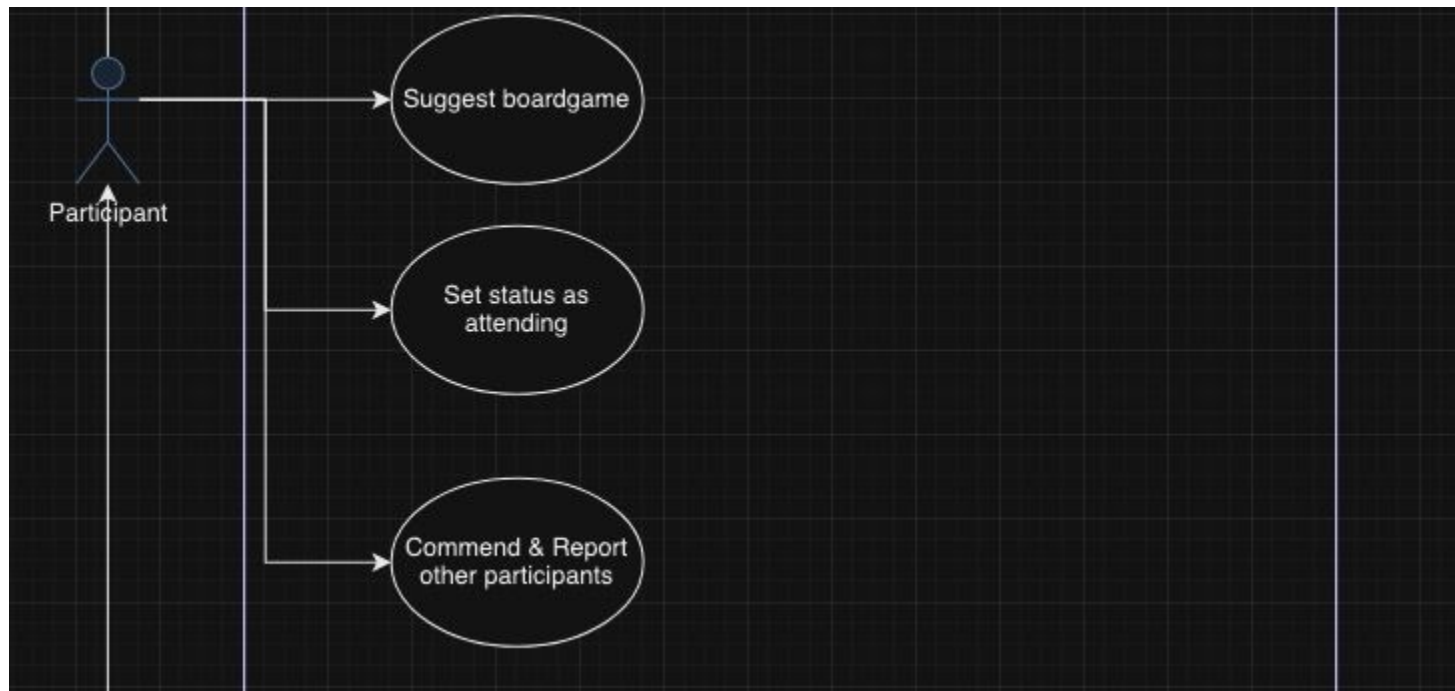
Funcționalități



Funcționalități



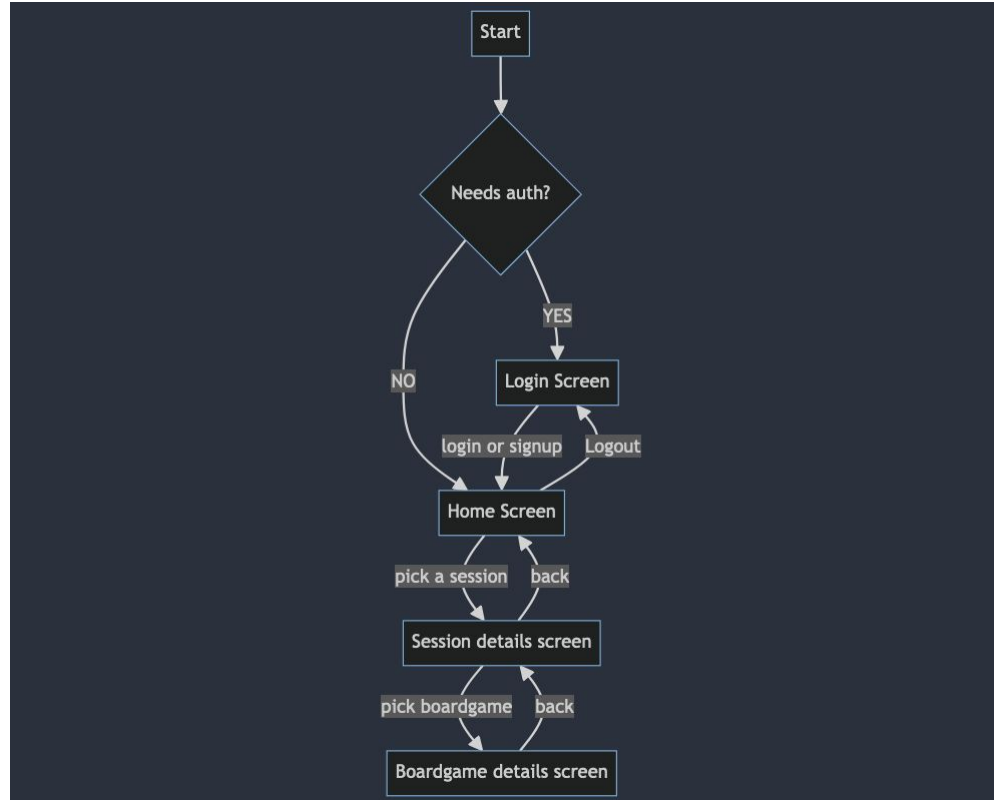
Funcționalități



Funcționalități

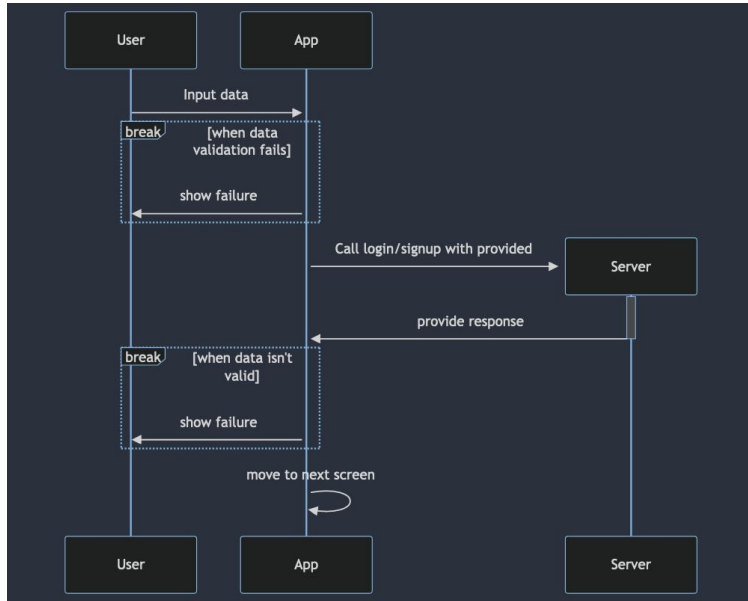


Interacțiuni



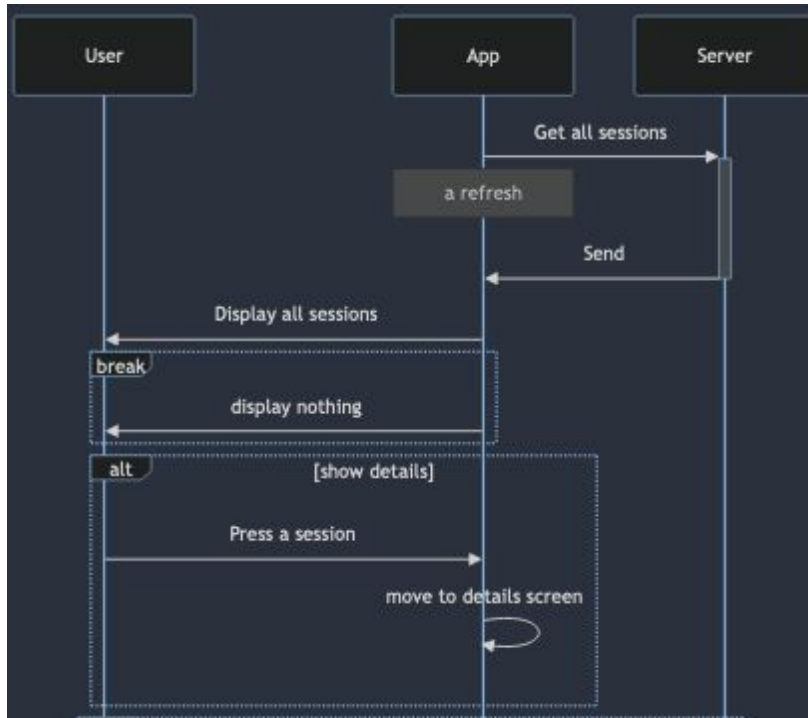
Interacțiuni

- login



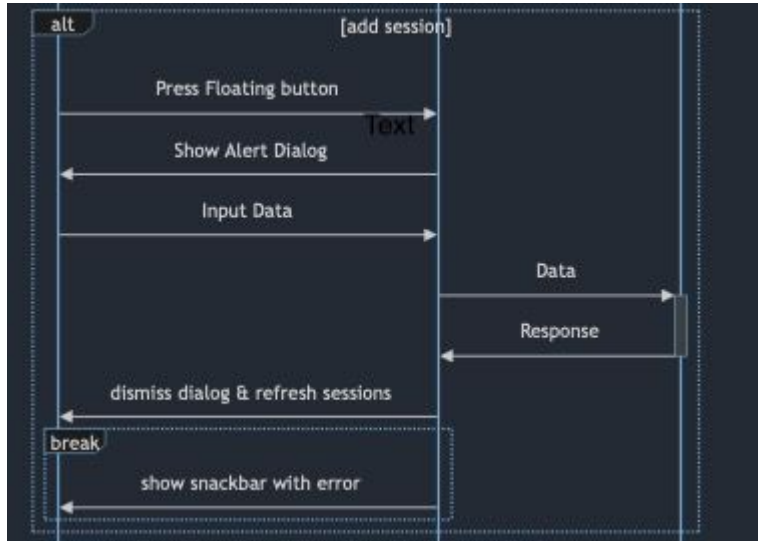
Interacțiuni

- home screen



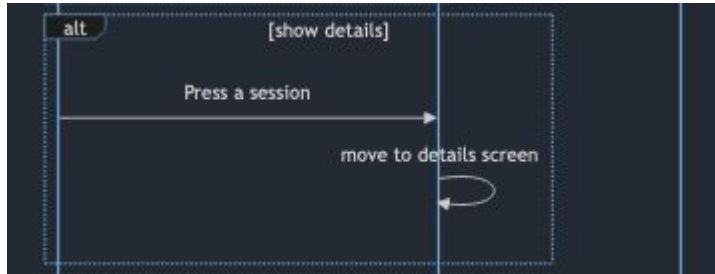
Interacțiuni

- home screen



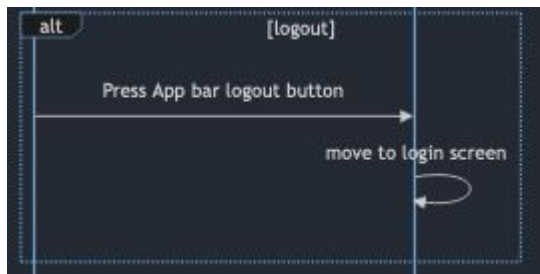
Interacțiuni

- home screen



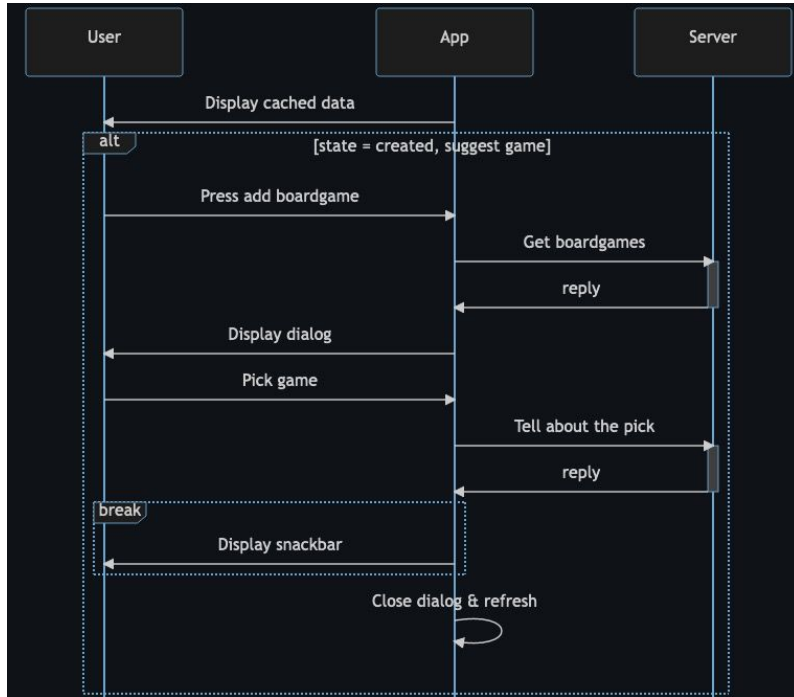
Interacțiuni

- home screen



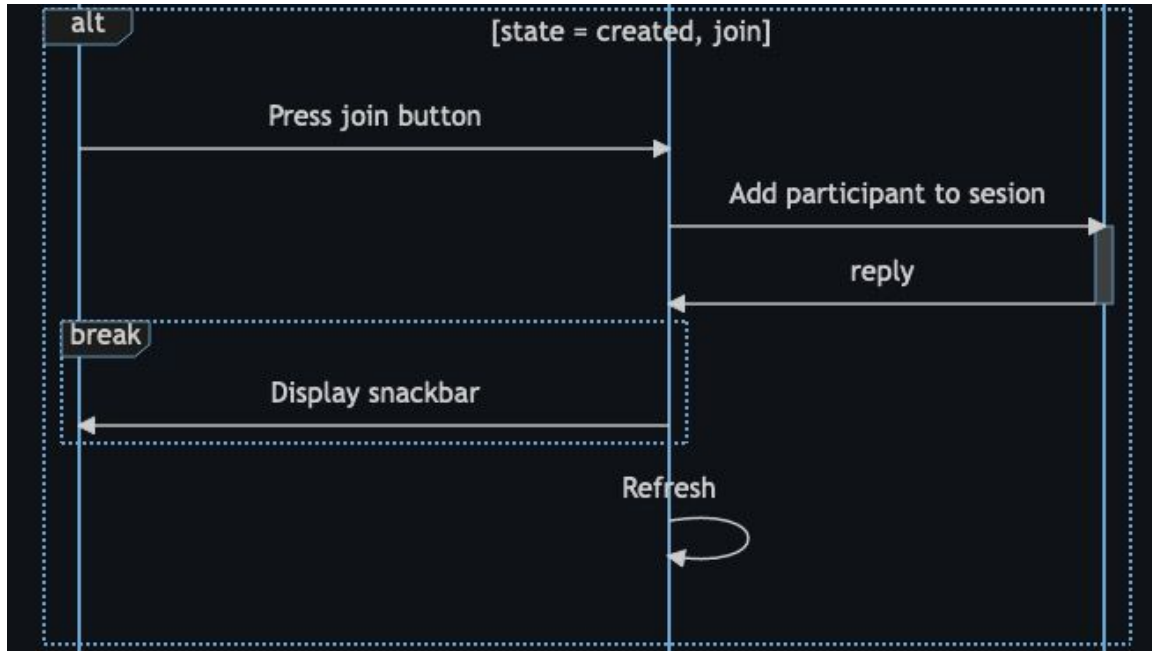
Interacțiuni

- session screen



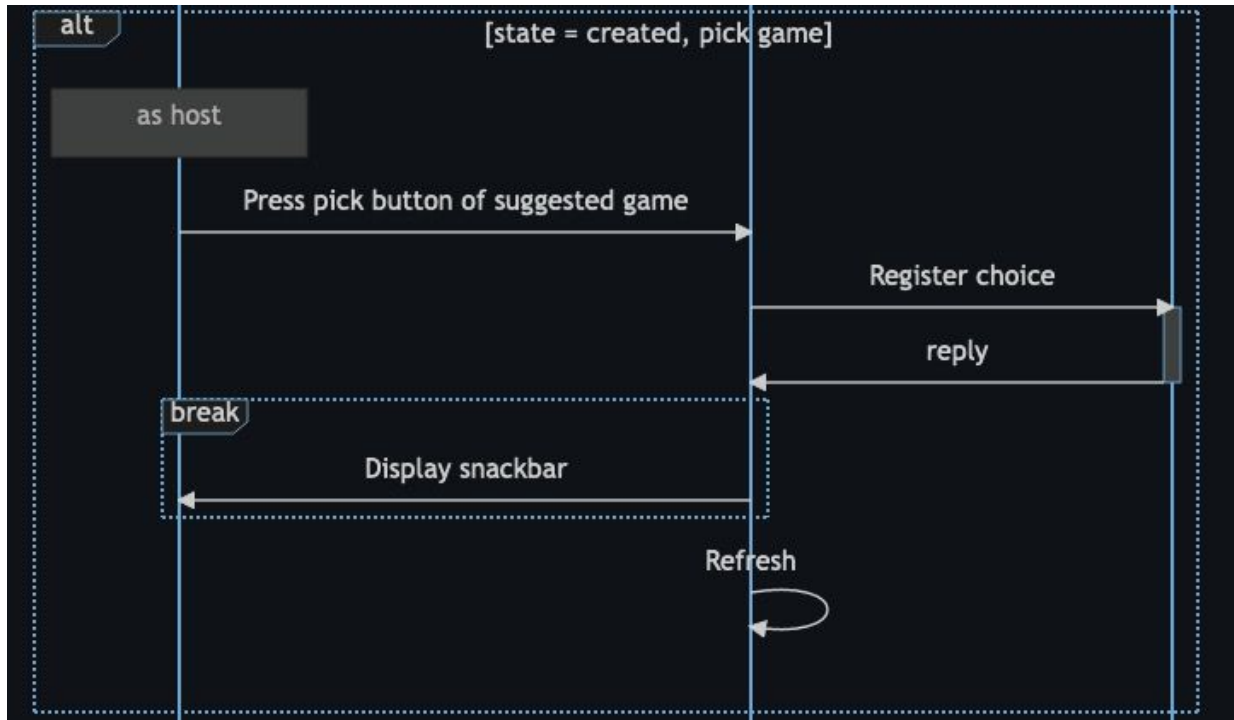
Interacțiuni

- session screen



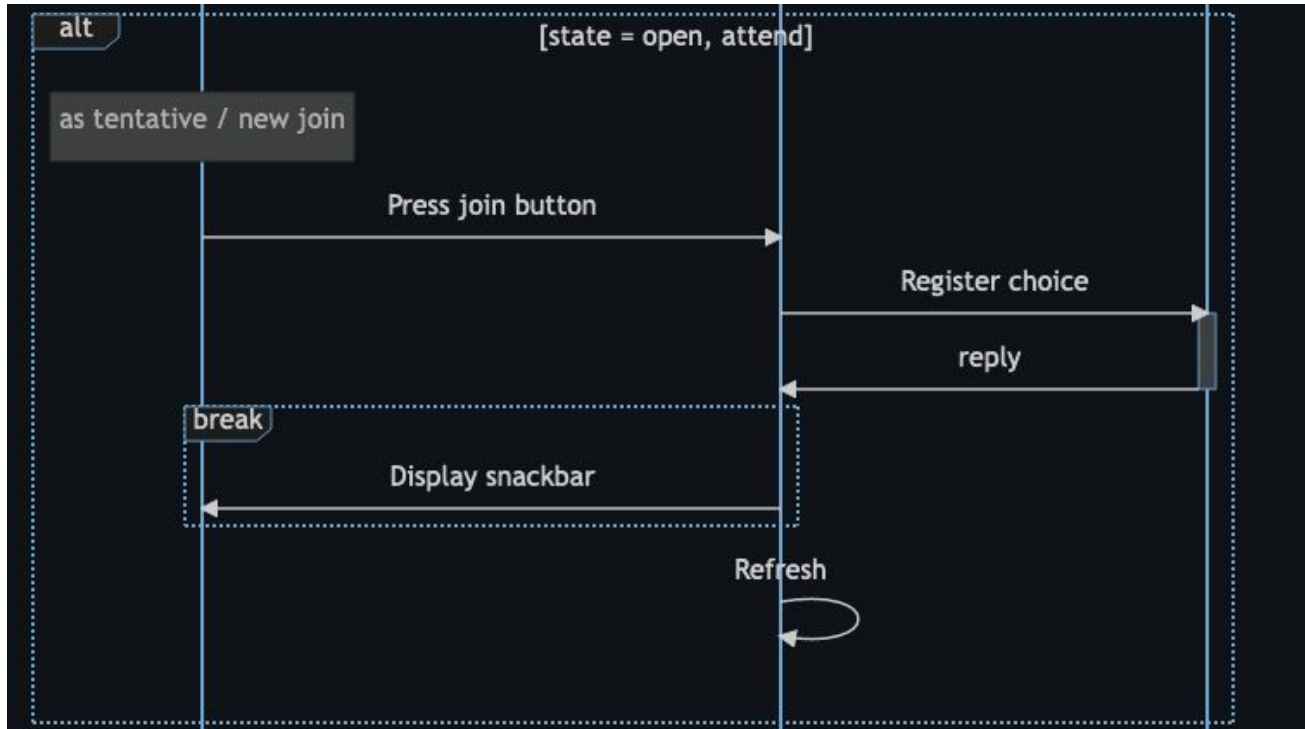
Interacțiuni

- session screen



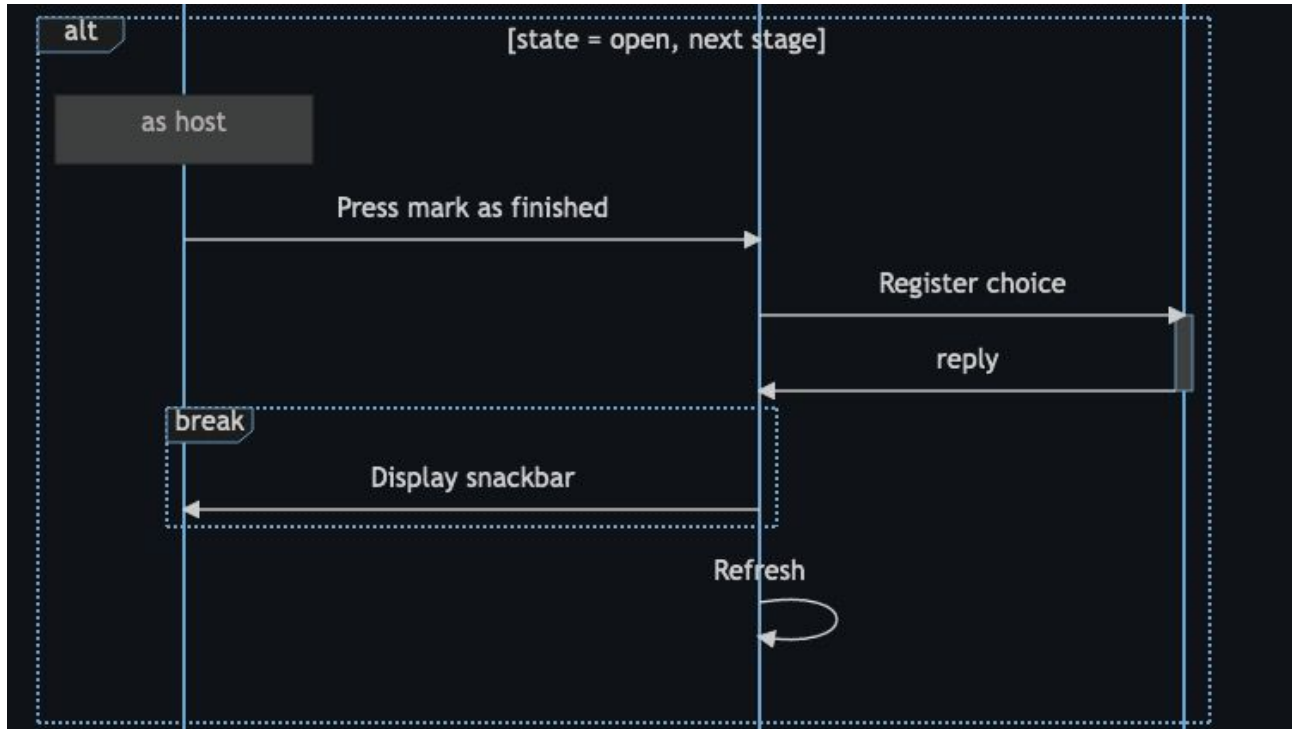
Interacțiuni

- session screen



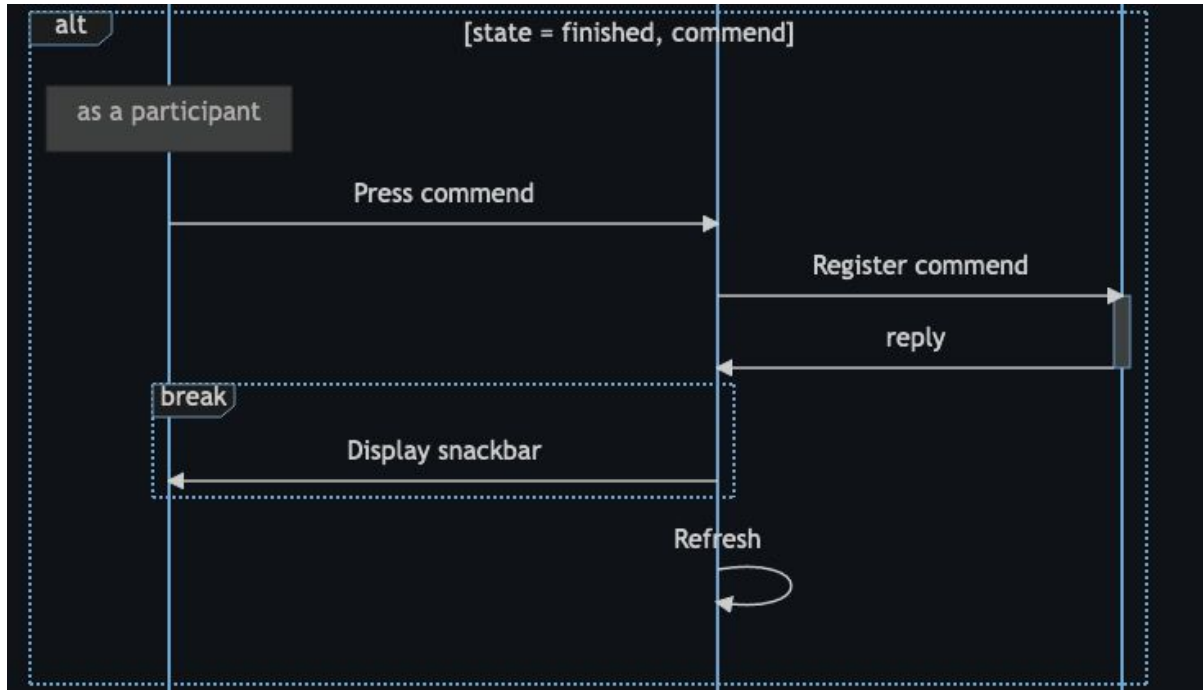
Interacțiuni

- session screen



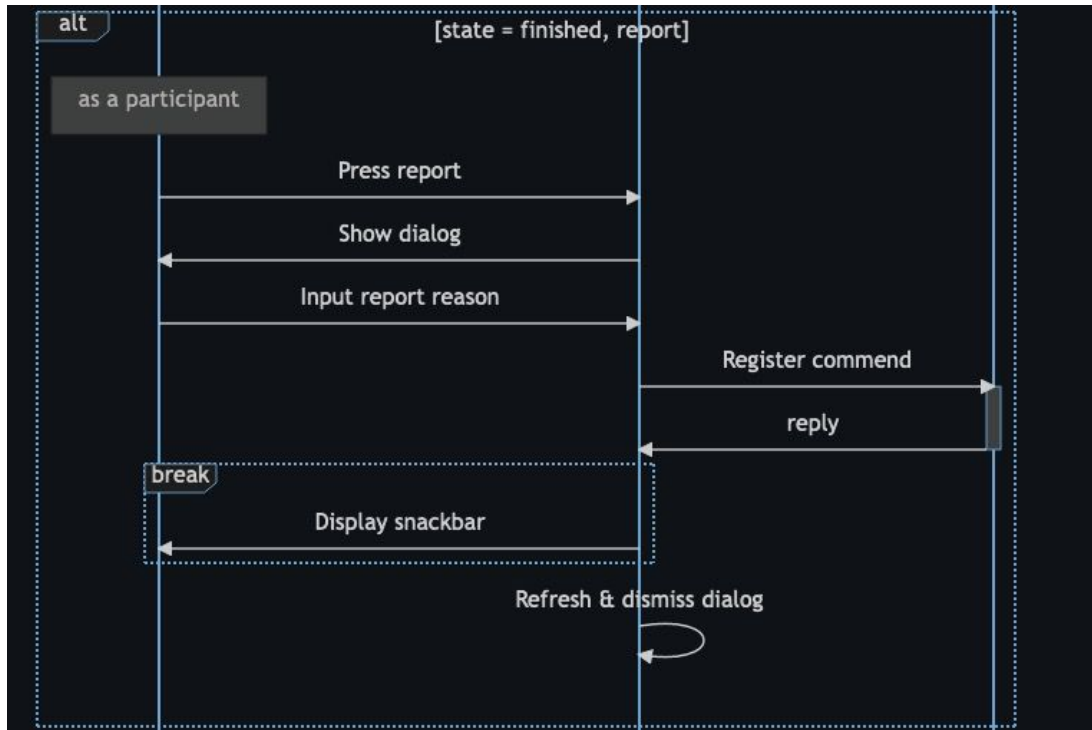
Interacțiuni

- session screen



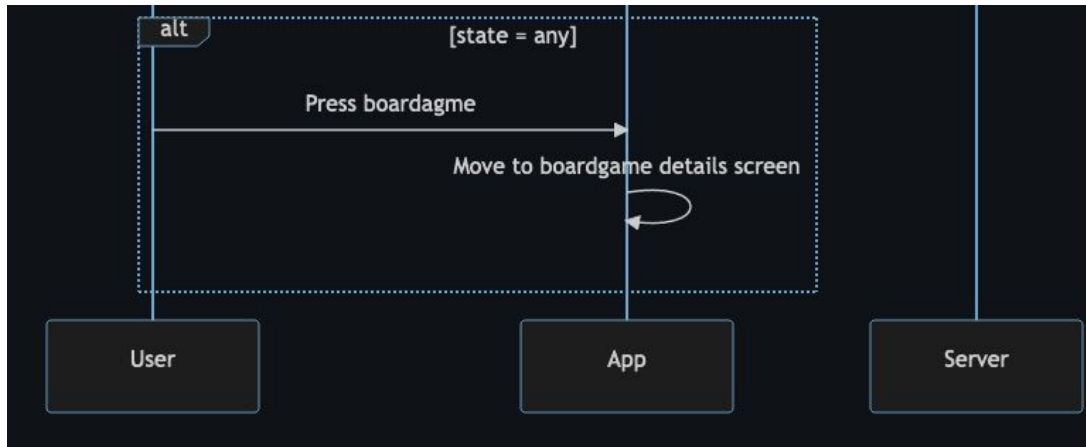
Interacțiuni

- session screen



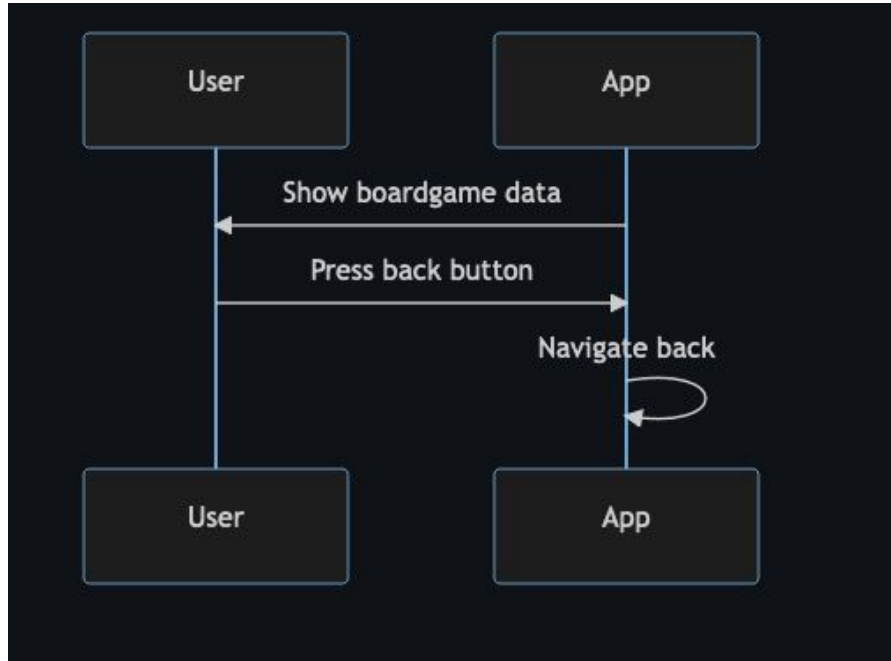
Interacțiuni

- session screen

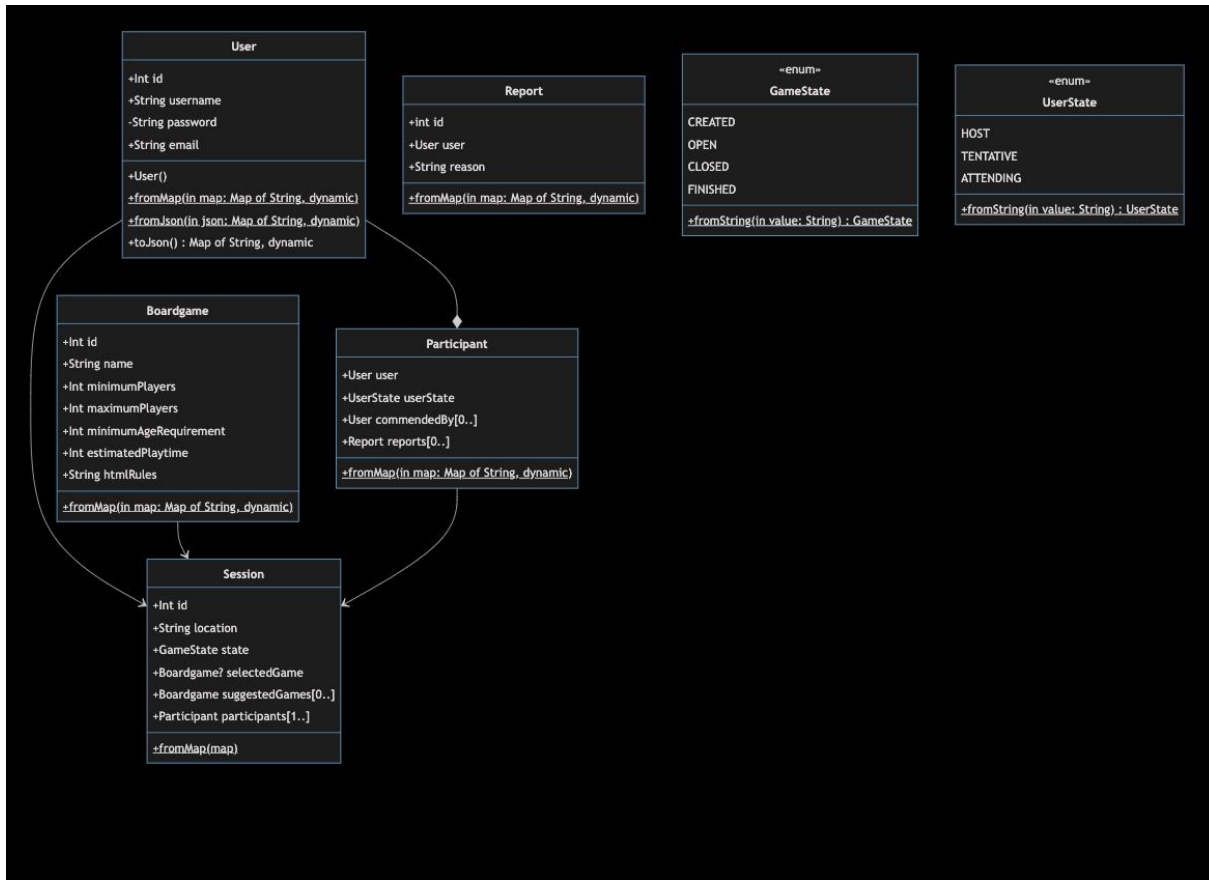


Interacțiuni

- boardgame screen



Entități

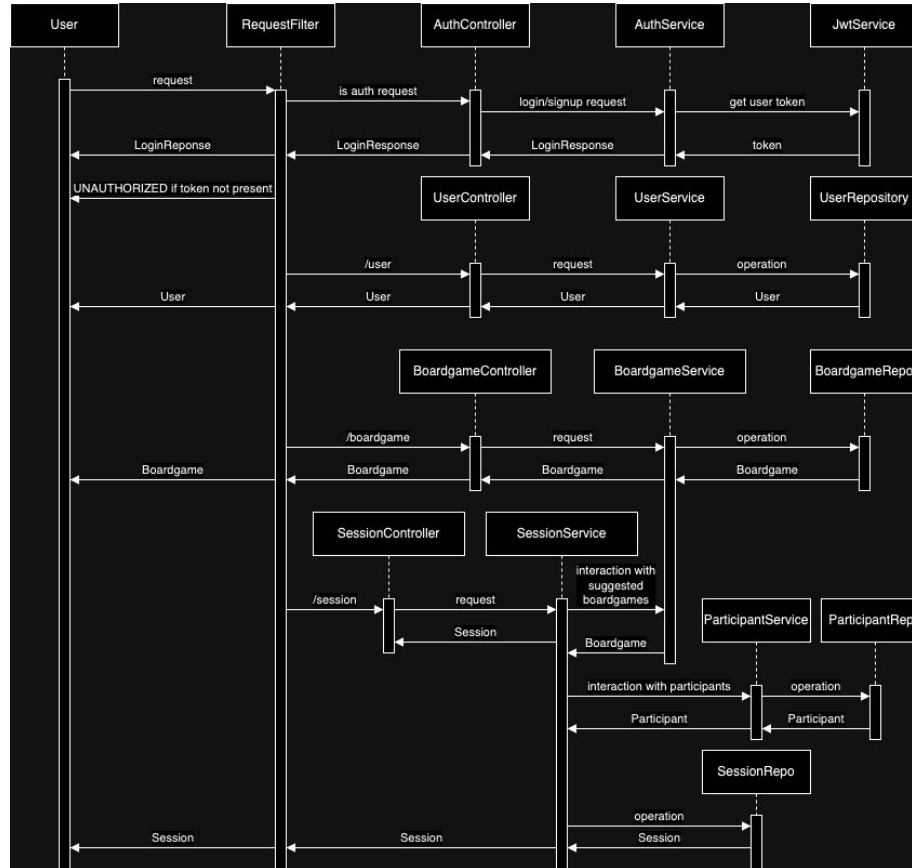


Request-uri

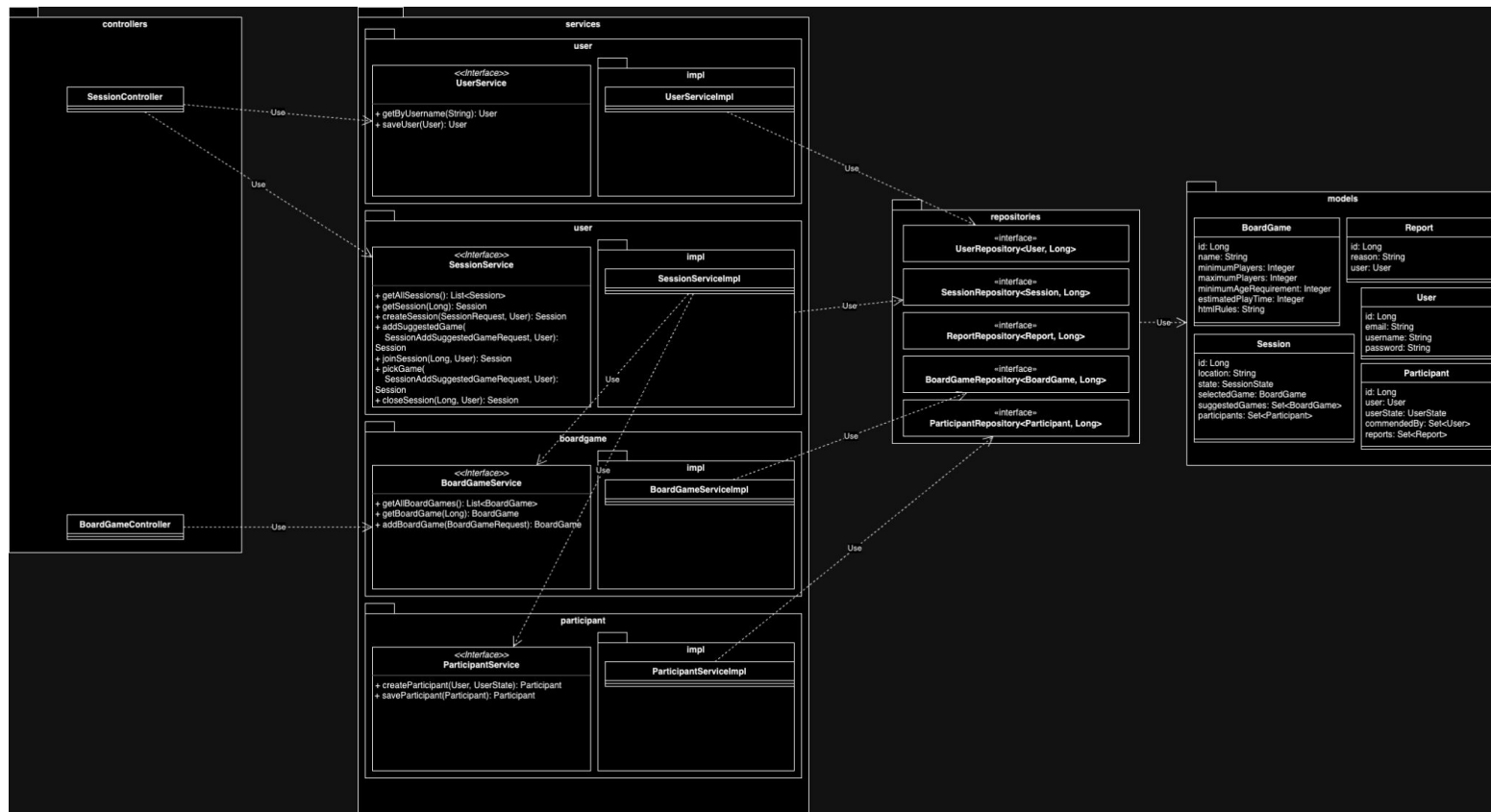
BoardGameRepository
-Dio dio
+BoardGameRepository(in dio: Dio)
+getAll() : List of Boardgame

SessionRepository
-Dio dio
+SessionRepository(in dio: Dio)
+getAllSessions() : List of Session
+addSession(in location: String)
+getSession(in id: Int) : Session
+selectGame(in sessionId: Int, in gameId: Int)
+suggestGame(in sessionId: Int, in gameId: Int)
+joinGame(in sessionId: Int)
+commendPlayer(in sessionId: Int, in userId: Int)
+reportPlayer(in sessionId: Int, in userId: Int, in reason: String)

Server handler

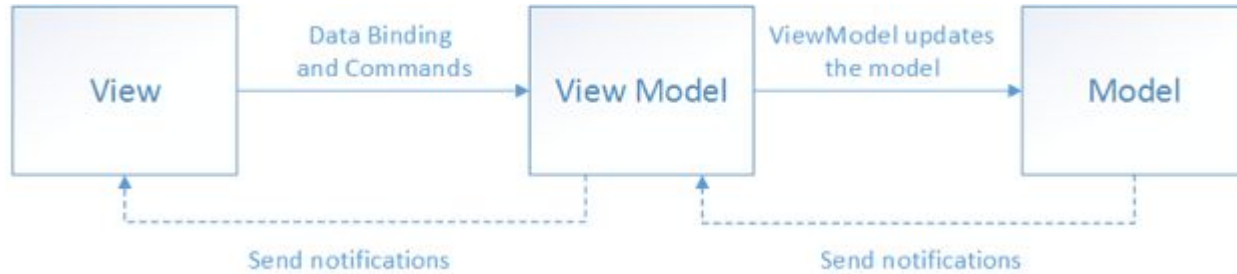


Server structure



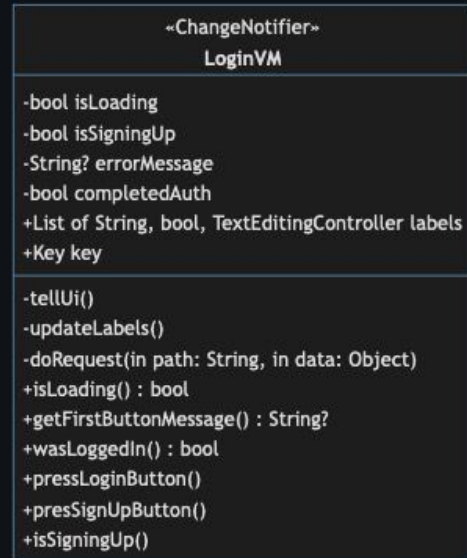
Design Patterns

- MVVM (supportive)



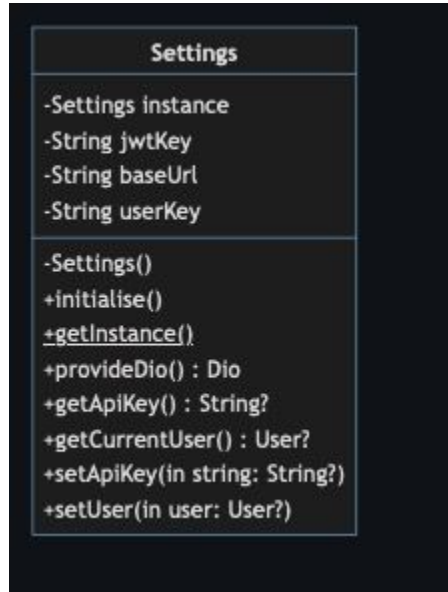
Design Patterns

- Observer & State pattern



Design Patterns

- Singleton Pattern



Design Patterns

BoardGameBuilder	
BoardGame	<ul style="list-style-type: none">- id: Long- name: String- minimumPlayers: Integer- maximumPlayers: Integer- minimumAgeRequirement: Integer- estimatedPlayTime: Integer- htmlRules: String
<ul style="list-style-type: none">+ id: Long+ name: String+ minimumPlayers: Integer+ maximumPlayers: Integer+ minimumAgeRequirement: Integer+ estimatedPlayTime: Integer+ htmlRules: String	<ul style="list-style-type: none">- id: Long- name: String- minimumPlayers: Integer- maximumPlayers: Integer- minimumAgeRequirement: Integer- estimatedPlayTime: Integer- htmlRules: String
<ul style="list-style-type: none">+ builder(): BoardGameBuilder	<ul style="list-style-type: none">+ id(Long): BoardGameBuilder+ name(String): BoardGameBuilder+ minimumPlayers(Integer): BoardGameBuilder+ maximumPlayers(Integer): BoardGameBuilder+ minimumAgeRequirements(Integer): BoardGameBuilder+ estimatedPlayTime(Integer): BoardGameBuilder+ htmlRules(String): BoardGameBuilder+ build(): BoardGame

Muṭumim

