Fișa Cerințelor

***“Ride Sharing”***

**Descriere**

Aplicația Ride Sharing este destinată conectării șoferilor cu pasagerii care doresc să călătorească într-o anumită direcție. Utilizatorii pot solicita curse, pot vizualiza detaliile șoferilor, pot efectua plăți și pot lăsa recenzii. Șoferii pot accepta cereri, pot vedea traseul optim și pot primi plata pentru serviciile oferite. Sistemul include mecanisme de siguranță, rating și suport pentru utilizatori.

**Actori**

*Pasager*: solicită curse, face plăți, evaluează șoferii.

*Șofer*: acceptă curse, transportă pasagerii, primește plăți, este evaluat.

*Administrator*: gestionează utilizatorii, monitorizează activitatea, intervine în caz de probleme.

*Sistem de Plată*: procesează plățile efectuate prin aplicație.

*Serviciu de Notificare*: trimite notificări despre curse și actualizări relevante.

*Sistem de Hărți*: oferă rute optime și calculează distanțele.

**Scenarii de Utilizare**

*Înregistrare și autentificare*: Un utilizator își creează un cont și se autentifică în aplicație.

*Solicitarea unei curse*: Pasagerul introduce destinația și primește opțiuni de șoferi disponibili.

*Acceptarea unei curse*: Șoferul primește cererea și decide dacă o acceptă.

*Calcularea rutei*: Sistemul de hărți oferă traseul optim pentru călătorie.

*Realizarea cursei*: Șoferul transportă pasagerul la destinație.

*Efectuarea plății*: Pasagerul plătește prin aplicație sau numerar.

*Oferirea unui rating*: Pasagerul și șoferul își oferă reciproc evaluări.

*Anularea unei curse*: Pasagerul sau șoferul poate anula cursa în condiții specifice.

*Suport pentru utilizatori*: Administratorul intervine în cazul problemelor sau disputelor.

*Notificări*: Utilizatorii primesc alerte despre cursele active și alte actualizări.

**Prompt utilizat**

„Realizează o Fișă a Cerințelor pentru un sistem de Ride Sharing, incluzând descrierea, actorii și scenariile de utilizare (doar descrierea). Apoi, creează diagrama Use Case în format PlantUML, incluzând relații de tip include și extend unde este relevant. Adaugă și o diagramă UML de clasă cu relații de agregare și compoziție.”

*Avantaje utilizării AI:*

**+** Generare rapidă și coerentă a cerințelor și diagramelor.

**+** Identificarea corectă a actorilor și relațiilor dintre use case-uri.

**+** Automatizarea scrierii codului PlantUML, eliminând posibile erori umane.

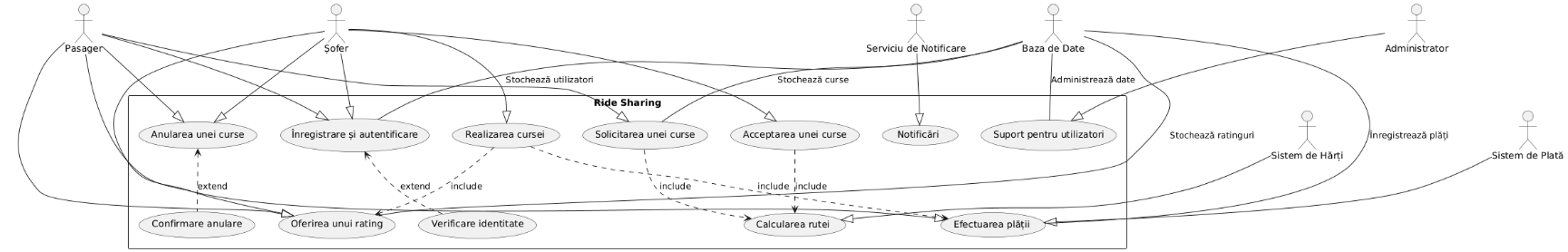
**+** Posibilitatea de a ajusta rapid structura în funcție de cerințele punctajului.

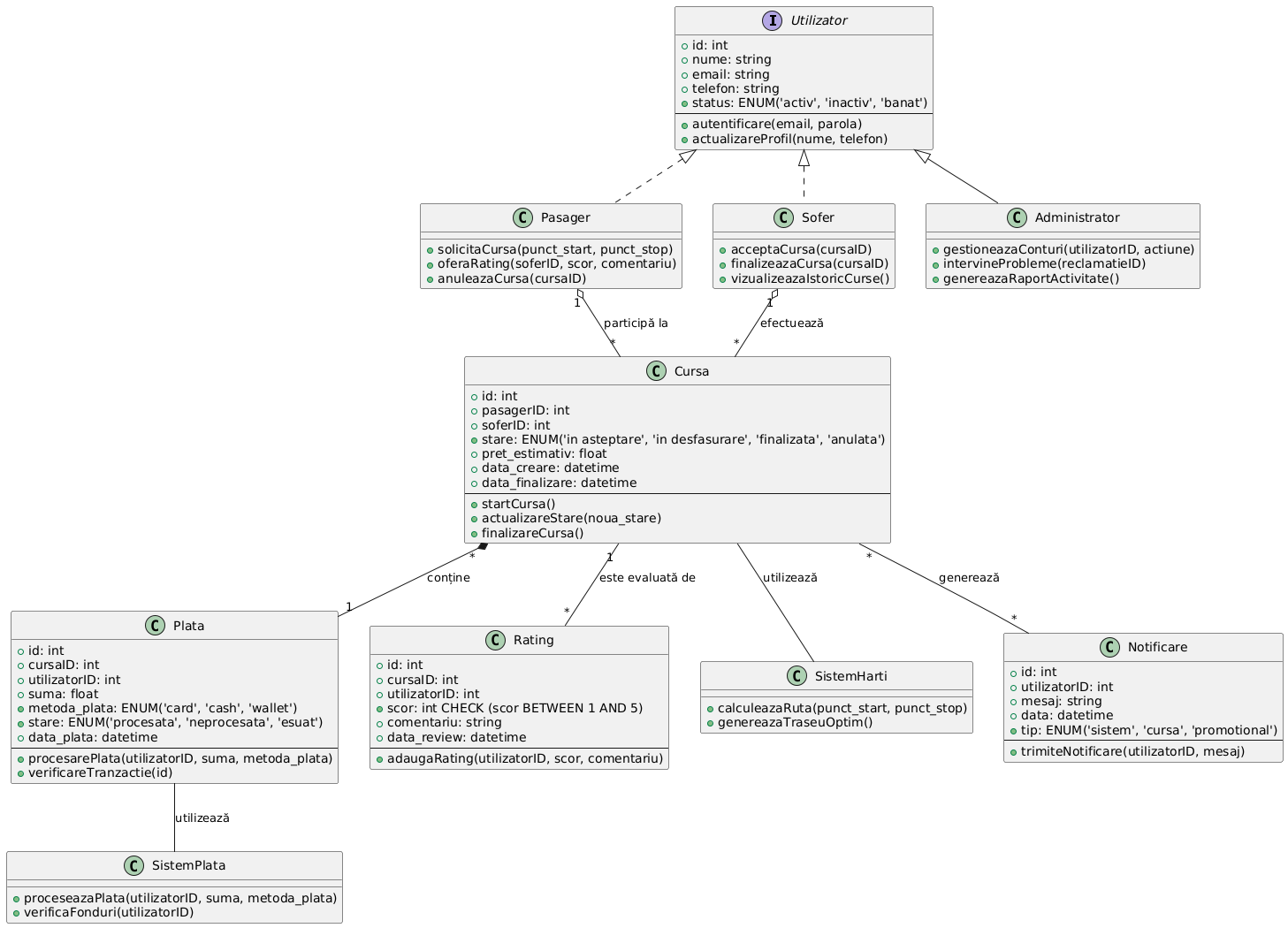
*Dezavantaje utilizării AI:*

**-** Necesită validare manuală pentru a evita suprasimplificarea scenariilor.

**-** Uneori, AI poate omite detalii importante, necesitând ajustări ulterioare.

**-** Nu poate genera direct diagrame vizuale, doar codul PlantUML care trebuie procesat separat.

**Diagrama Use Case:**

**Diagrama de clase:**