

RiotSupportTracker

Készítette: Szomor Levente SZFT5

Dátum: 2026-02-22

Készítsen egy egyszerű hibakövető (support ticket) rendszert, ahol hibákat lehet felvenni, support agenthez rendelni, adataikat módosítani és a státuszukat követni. A feladat League of Legends / Riot Support kontextusban készüljön.

Backend programozás

Adatbázis és modellek

- Hozzon létre megfelelő migrációs fájlokat, amelyek létrehozzák az agents és a tickets táblákat az adatbázisban az alábbiaknak megfelelően. Ne legyenek időbényegek a táblákban.

agents tábla		
Mező neve	Típus	Leírás
id	Egész [AI, PK]	A support agent azonosítója
name	Szöveg (max. 50 karakter)	A support agent neve

tickets tábla		
Mező neve	Típus	Leírás
id	Egész [AI, PK]	A ticket azonosítója
title	Szöveg (max. 100 karakter)	A ticket rövid összefoglaló címe
status	Szöveg (max. 20 karakter), default érték: open (open, investigating, resolved, closed)	defiktétérkép (open, investigating, resolved, closed)
agent_id	Egész [FK] vagy NULL	A ticket felelősének azonosítója
description	Szöveg (max. 2000 karakter)	A ticket részletes leírása

- Állítsa be, hogy a seederek megfelelő sorrendben fussanak le. Futtassa a migrációt és a seedelést is, majd ellenőrizze az adatokat az adatbázisban.
- Hozza létre az Agent és Ticket modelleket. Gondoskodjon róla, hogy a modellek ne használjanak időbényegeket. A Ticket modell status mezőjének alapértelmezett értéke open legyen.
- Az Agent modellben definiáljon egy tickets metódust, ami visszaadja az agenthez rendelt ticketeket.
- A Ticket modellben definiáljon egy agent metódust, ami visszaadja a tickethez rendelt agentet.

Routing

- A feladat megoldása során az alábbi végpontokat kell megvalósítani (a {ticket} és {agent} helyett {id} is használható). Ügyeljen rá, hogy az azonosító csak szám lehet!

Metódus	Végpont	Controller	Action
GET	/api/tickets	TicketController	index
GET	/api/tickets/{ticket}	TicketController	show
POST	/api/tickets	TicketController	store
PUT	/api/tickets/{ticket}	TicketController	update
DELETE	/api/tickets/{ticket}	TicketController	destroy
GET	/api/agents	AgentController	index
GET	/api/agents/{agent}	AgentController	show

Validálás

Jogosultságot nem kell ellenőrizni, minden felhasználónak engedélyezze a műveleteket.

7. A Ticket validálási szabályai:

- A title mező kötelező, maximum 100 karakter hosszú lehet.
- A status mező nem kötelező (adatbázisbeli alapértelmezett értéke open), csak az open, investigating, resolved, closed értékeket veheti fel.
- Az agent_id mező nem kötelező, ha meg van adva, akkor csak létező Agent azonosító lehet.
- A description mező kötelező, minimum 10, maximum 2000 karakter hosszú lehet.

8. Az Agent validálási szabályai:

- A name mező kötelező, maximum 50 karakter hosszú lehet.

Controllerek

TicketController

9. A TicketController-ben valósítsa meg az alábbi metódusokat:

- index: Visszaadja az összes ticketet.
- show: Visszaadja a megadott azonosítójú ticketet.
- store: Validálás után felveszi az új ticketet, és visszaadja az adatait. Sikertelen validálás esetén tájékoztatja a felhasználót a hibákról.
- update: Validálás után módosítja a megadott azonosítójú ticket adatait, és visszaadja a módosított ticketet.
- destroy: Törli a megadott azonosítójú ticketet. Sikeres törlés esetén 204 No Content választ adjon.

AgentController

10. Az AgentController-ben valósítsa meg az alábbi metódusokat:

- index: Visszaadja az összes agentet, mindegyik agenthez a hozzá rendelt ticketeket is.

- show: Visszaadja a megadott azonosítójú agentet a hozzá rendelt ticketekkel együtt.

API

11. A kommunikáció minden irányban JSON formátumban történjen.

12. Az API leírását a Swagger tartalmazza. A mellékelt openapi_riot_support_tracker.yaml fájl alapján legyen elérhető a dokumentáció.

Tesztelés

13. Tesztelje, hogy az elkészített API a leírásnak megfelelően működik-e. A tesztelést végezheti böngészőben, Thunder Client vagy Bruno segítségével is.

Frontend programozás

Útvonalak

- Készítse el az alábbi útvonalakat a tanult módszerek egyikével. (Megjegyzés: a Komponens oszlop csak azok számára információ, akik nem az unplugin-vue-router-t használják.)

Útvonal neve	Komponens	URL
home	index.vue	/
create-ticket	create.vue	/tickets/create
edit-ticket	[id].vue	/tickets/:id
agents	index.vue	/agents
agent	[id].vue	/agents/:id

- Állítsa be, hogy a router betöltődése előtt az oldal címe legyen az adott oldal meta.title értéke!

Tárolók

TicketStore

- Készítsen egy TicketStore nevű tárolót, amely a ticketek adatait kezeli. Hozzon benne létre egy tickets állapotot.
- A TicketStore-ban hozzon létre egy getTickets() műveletet, amely lekéri a ticketeket az API-ról és beállítja az állapotot.
- A TicketStore-ban hozzon létre egy getTicket(id) műveletet, amely lekéri a megadott azonosítójú ticketet az API-ról és visszaadja.
- A TicketStore-ban hozzon létre egy createTicket(ticket) műveletet, amely létrehozza az új ticketet az API-n, és a válaszként visszakapott objektumot hozzáadja az állapothoz.
- A TicketStore-ban hozzon létre egy updateTicket(id, ticket) műveletet, amely módosítja a megadott azonosítójú ticketet az API-n, és a visszakapott objektummal frissíti az állapotban tárolt ticketet.
- A TicketStore-ban hozzon létre egy deleteTicket(id) műveletet, amely törli a ticketet az API-n, majd eltávolítja az állapotból.
- A TicketStore-ban hozzon létre egy statuses() gettert, amely egy objektumban visszaadja a státuszokat: open, investigating, resolved, closed; az értékek magyar nevek nagy kezdőbetűvel: Nyitott, Vizsgálat alatt, Megoldva, Lezárvva.

AgentStore

- Készítsen egy AgentStore nevű tárolót, amely az agentek adatait kezeli. Hozzon benne létre egy agents állapotot.
- Az AgentStore-ban hozzon létre egy getAgents() műveletet, amely lekéri az agenteket az API-ról és beállítja az állapotot.

12. Az AgentStore-ban hozzon létre egy getAgent(id) műveletet, amely lekéri a megadott azonosítójú agentet az API-ról és visszaadja.

13. Az AgentStore-ban hozzon létre egy agentOptions() gettert, amely olyan objektumot ad vissza, ahol a kulcsok az agentek azonosítói, az értékek pedig az agentek nevei. Legelsőként tartalmazzon egy üres string kulcsot üres string értékkel arra az esetre, amikor nincs hozzárendelt agent. Ennek segítségével töltse fel majd a ticket űrlapon az agent lenyíló listáját.

Betöltés

14. A gyökérkomponens betöltésére fusson le minden két tároló összes adatának lekérése.

Oldalak

Főoldal

15. A főoldalon két komponenst használjon:

- A BaseLayout komponens feleljen az oldalszerkezetért.
- A BaseCard komponens jelenítsen meg egy ticketet.

16. A BaseLayout-on belül helyezze el az oldal főcímét: Riot Support Tracker, alá pedig egy általános tárolóelemet.

17. A főcím minden oldalon legyen vízszintesen középre igazítva és félkövér. Függőleges irányban állítsa be 10 egység külső margót.

18. A főcím betűmérete kis és közepes méretben 2xl legyen, nagy mérettől kezdve 3xl.

19. A főcím alatt jelenítsen meg a TicketStore statuses getteréből kapott státuszokhoz 4 oszlopot egy rácsban. Feltételezheti, hogy a 4 státusz minden adott.

- Kis és közepes nézeten az oszlopok egymás alatt legyenek, és foglalják el a teljes szélességet.
- Nagy nézettől felfelé a 4 oszlop egymás mellett legyen, egyenlő szélességgel.
- Az oszlopok között 4 egység (1rem) távolság legyen.
- Az oszlopoknak legyen 4 egység (1rem) belső margója, neutral-100 háttérszíne és nagy lekerekített sarkai.
- Az oszlop címe legyen középre igazítva, és a magyar státusznév legyen (a getter értéke).

20. Az oszlopokon belül BaseCard használatával jelenítse meg az adott státuszhoz tartozó ticketeket egymás alatt. A kártya címe a ticket title mezője legyen, alatta pedig az agent neve jelenjen meg, ha van hozzárendelve. Az oszlop címe alatt és a kártyák között is legyen 2 egység (0.5rem) távolság.

Új ticket oldal

21. Az oldal tartalmát a BaseLayout komponensben helyezze el. Az oldal főcíme: Új ticket.

22. A főcím alá helyezzen el egy általános tárolóelemet, és azon belül egy űrlapot. Az alapértelmezett műveletet tiltsa le.

Címke	Név	Típus	Validáció
Cím	title	szöveges mező	kötelező, maximum 100 karakter
Státusz	status	legördülő lista	engedélyezett státuszokat ajánlja fel; a magyar nev
Agent	agent_id	legördülő lista	agent nevét mutatja, azonosítót menti; használja az
Leírás	description	többsoros szöveges mező	kötelező, minimum 10, maximum 2000 karakter
Felvétel	createBtn	submit gomb	

23. A lenyíló listák feltöltésénél ne felejtse el használni a tárolókban definiált gettereket.

24. Készítsen egy newTicket(ticket) függvényt, amely a submit gomb megnyomására hívódik. A létrehozás műveletét a TicketStore createTicket függvényével valósítsa meg.

25. Mentés után irányítsa át a felhasználót a főoldalra.

Ticket szerkesztése oldal

26. Az "Új ticket" oldalhoz hasonlóan hozza létre a ticket szerkesztés oldalt is. Az oldalon ne legyen főcím.

27. Az oldal tartalmát BaseLayout komponensben helyezze el.

28. Készítsen egy ticket belső állapotot, amely kezdetben null értékű, továbbá egy loading állapotot true kezdőértékkel.

29. Az oldal megjelenésekor kérje le a tároló getTicket(id) műveletével a ticket adatait az útvonalban kapott azonosító alapján, és töltse fel vele az űrlapot. A loading állapot értéke ennek megfelelően változzon.

30. Amíg a loading értéke igaz, a BaseSpinner komponens jelenjen meg.

31. Betöltés után jelenítsen meg egy általános tárolóelemet, és azon belül az "Új ticket" oldalhoz hasonló szerkezetű űrlapot, előtöltött adatokkal.

32. Az űrlap alján helyezzen el egy "Mentés" feliratú gombot és egy "Törlés" feliratú gombot.

33. Készítsen egy editTicket(ticket) függvényt, amely a Mentés gomb megnyomására hívódik. A módosítást a TicketStore updateTicket(id, ticket) függvényével valósítsa meg.

34. Készítsen egy removeTicket() függvényt, amely a Törlés gomb megnyomására hívódik. Használja a bönghészű beépített confirm ablakát. Megerősítés után hívja meg a TicketStore deleteTicket(id) függvényét.

35. Mind mentés, mind törlés után irányítsa át a felhasználót a főoldalra.

Agente(k) oldal

36. Az oldal tartalmát a BaseLayout komponensben helyezze el. Az oldal főcíme: Agensek.

37. A főcím alá helyezzen el egy általános tárolóelemet, és azon belül egy táblázatot a BaseTable komponenssel, amely az agentek adatait jeleníti meg.

Agenthez rendelt ticketek oldala

38. Az oldal tartalmát a BaseLayout komponensben helyezze el. Az oldal főcíme a kiválasztott agent neve legyen.

39. A főcím alá helyezzen el egy általános tárolóelemet, és a főoldalhoz hasonló elrendezésben, státsz szerint oszlopokban és azon belül kártyákon jelenítse meg az agenthez rendelt ticketeket. A kártyára kattintva irányítson a ticket szerkesztés oldalára.

Komponensek

Gyökérkomponens

40. Állítsa be, hogy az oldal betöltésekor minden két tároló kérje le az adatokat (ticketek és agensek).

BaseLayout

41. Ezt a komponenst nem kell módosítani, csak használni.

BaseHeader

42. A navigációs sávban a cím legyen: Riot Support Tracker.

43. A navigációs sáv jobb oldalán jelenjen meg két menüpont:

- "Agensek" – az /agents oldalra visz.
- "Új ticket" – a /tickets/create oldalra visz.

BaseSpinner

44. Ezt a komponenst nem kell módosítani, csak használni a betöltési folyamat jelzésére.

BaseCard

45. A komponens könyvtárban hozzon létre egy BaseCard.vue fájlt, ami egy kártyát jelenít meg.

46. A komponens a szülő komponenstől kapjon egy url adatot, az egész kártya egy navigációs linkként működjön.

47. A komponens jelenítse meg a kapott title stringet, és opcionálisan egy subtitle stringet.

48. A kártya kerete legyen neutral-800 színű és 1 pixel vastag, a sarka lekerekített. Legyen 2 egység (0.5rem) belső margója. Hover esetén kapjon árnyékot.

BaseTable

49. A komponens könyvtárban hozzon létre egy BaseTable.vue fájlt, ami egy táblázatot jelenít meg.

50. A komponens a szülőtől kapjon egy agents tömböt, ami agent objektumokat tartalmaz, és egy ticketCountsByAgentId objektumot, ami az agent id-khez a ticketek számát tartalmazza.

51. A táblázat fejléce a mintának megfelelő legyen. A következő oszlopok legyenek: "Név", "Hozzárendelt ticketek".

52. A táblázat törzsében generálja le a sorokat az agentek alapján. A sorok váltakozó háttérszínűek legyenek: neutral-100 és neutral-200.

53. Mind az agent neve, mind a ticketek száma legyen router link, ami az agent részletek oldalára irányít.

54. A táblázat teljes szélességen töltse ki a tárolót. A fejlécek háttérszíne neutral-700, a szöveg színe neutral-100 legyen. A táblázat szegélye neutral-800, 1 pixel vastag, a cellák belső margója 2 egység (0.5rem).