SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
Fakulta Informatiky a Informačných technológií

Matej Delinčák, Lucia Ondovčíková

Dokumentácia k aplikácii VoidMessenger

Vývoj progresívnych webových aplikácií

Krúžok: Streda 14:00
Cvičiaci: Ing. Eduard Kuric, PhD.
Akademický rok: 2022

Obsah

Zadanie

Vytvorte progresívnu webovú aplikáciu na textovú komunikáciu v štýle IRC (Slack), ktorá komplexne rieši nižšie definované prípady použitia.

1. registrácia, prihlásenie a odhlásenie používateľa
    - používateľ má meno a priezvisko, nickName a email
2. používateľ vidí zoznam kanálov, v ktorých je členom
    - pri opustení kanála, alebo trvalom vyhodení z kanála je daný kanál odobratý zo zoznamu
    - pri pozvánke do kanála je daný kanál zvýraznený a topovaný
    - v zozname môže cez používateľské rozhranie kanál vytvoriť, opustiť, a ak je správcom aj zrušiť
    - dva typy kanálov - súkromný (private channel) a verejný kanál (public channel)
    - správcom kanála je používateľ, ktorý kanál vytvoril
    - ak nie je kanál aktívny (nie je pridaná nová správa) viac ako 30 dní, kanál prestáva existovať (následne je možné použiť channelName kanála pre "nový" kanál)
3. používateľ odosiela správy a príkazy cez "príkazový riadok", ktorý je "fixným" prvkom aplikácie. používateľ môže odoslať správu v kanáli, ktorého je členom
4. vytvorenie komunikačného kanála (channel) cez príkazový riadok
    - kanál môže vytvoriť ľubovolný používateľ cez príkaz /join channelName [private]
    - do súkromného kanála môže pridávať/odoberať používateľov iba správca kanála cez príkazy /invite nickName a /revoke nickName
    - do verejného kanála sa môže pridať ľubovolný používateľ cez príkaz /join channelName (ak kanál neexistuje, automaticky sa vytvorí)
    - do verejného kanála môže člen kanála pozvať iného používateľa príkazom /invite nickName
    - vo verejnom kanáli môže člen "vyhodiť" iného člena príkazom /kick nickName. ak tak spravia aspoň 3 členovia, používateľ má "trvalý" ban pre daný kanál. správca môže používateľa vyhodiť "natrvalo" kedykoľvek príkazom /kick nickName, alebo naopak "obnovit" používateľovi prístup do kanála cez príkaz /invite
    - nickName ako aj channelName sú unikátne
    - správca môže kanál zatvoriť/zrušiť príkazom /quit
5. používateľ môže zrušiť svoje členstvo v kanáli príkazom /cancel, ak tak spraví správca kanála, kanál zaniká
6. správu v kanáli je možné adresovať konkrétnemu používateľovi cez príkaz @nickname
    - správa je zvýraznená danému používateľovi v zozname správ
7. používateľ si môže pozrieť kompletnú históriu správ
    - efektívny inifinite scroll
8. používateľ je informovaný o každej novej správe prostredníctvom notifikácie
    - notifikácia sa vystavuje iba ak aplikácia nie je v stave "visible" (pozrite quasar docu App Visibility)

    - notifikácia obsahuje časť zo správy a odosielateľa
    - používateľ si môže nastaviť, aby mu chodili notifikácie iba pre správy, ktoré sú mu adresované
9. používateľ si môže nastaviť stav (online, DND, offline)
    - stav sa zobrazuje používateľom
    - ak je nastavený DND stav, neprichádzajú notifikácie
    - ak je nastavený offline stav, neprichádzajú používateľovi správy, po prepnutí do online sú kanály automaticky aktualizované
10. používateľ si môže pozrieť zoznam členov kanála (ak je tiež členom kanála) príkazom /list
11. ak má používateľ aktívny niektorý z kanálov (nachádza sa v okne správ pre daný kanál) vidí v stavovej lište informáciu o tom, kto aktuálne píše správu (napr. Ed is typing)
    - po kliknutí na nickName si môže pozrieť rozpísaný text v reálnom čase, predtým, ako ju odosielateľ odošle (každá zmena je viditeľná) :-)

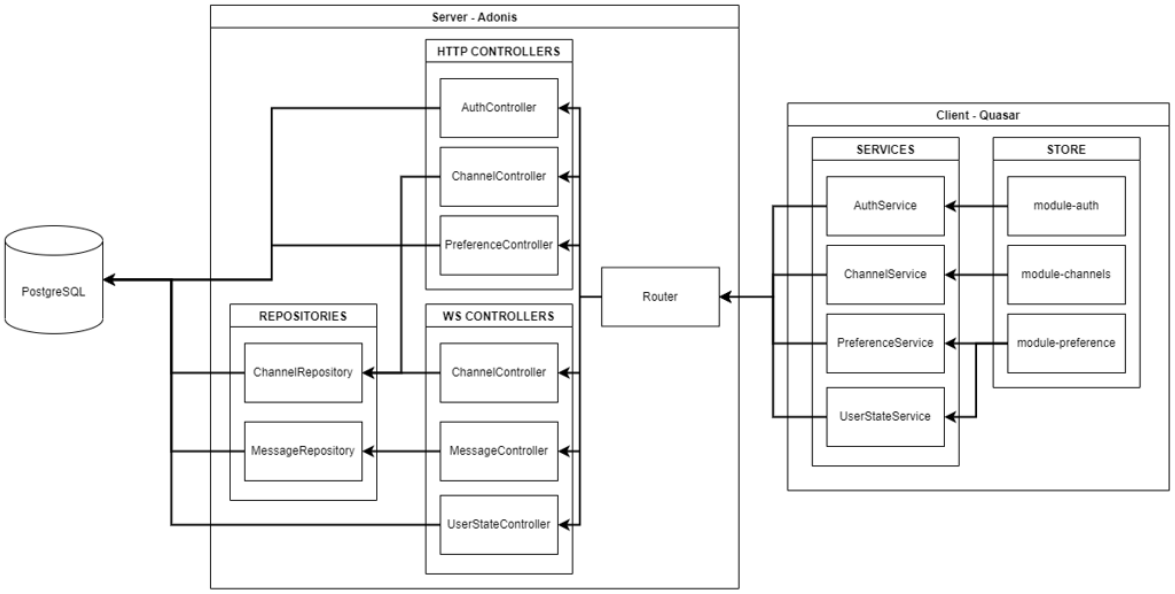
Diagram fyzického dátového modelu

Obrázok 1: Fyzický dátový model

- pridaná tabuľka api_tokens - autentifikačný aparát
- tabuľka relations premenovaná na roles
- tabuľka unread_channels bola vypustená

- tabuľku states (ktorá obsahovala iba aktuálny stav používateľa) sme nahradili tabuľkou preferences, ktorá bola rozšírená o dark_mode a notifications_on
- v tabuľke channel_users sme vypustili kick_counter a nahradili to:
    - kickedBy - tam si uchovávame id používateľov, ktorí ho kickli, aby sme predišli viacnásobnému vyhodeniu tým istým používateľom
    - kickedAt - na začiatku je nastavený na null, ak bude vyhodený niekym z kanálu, nastaví sa mu timestamp. Pri private kanáloch ho vie zrušiť len owner channelu, nastaví ho naspäť na null. Pri public kanáli ho vie zrušiť používateľ sám opätovným joinom.

Diagram architektúry aplikácie



Obrázok 2: Diagram architektúry aplikácie

Pri emitovaní akcie používateľom, sa zavolá akcia zo storu. Tá následne zavolá metódu z príslušného service-u. Service pošle request na server, kde pomocou routeru sa požiadavka posunie daným kontrolerom. Ak kontrolér obsahuje len málo funkcií a do db toľko nezasahuje, nevytvárali sme repozitár. Ak kontrolér má teda viacej funkcionality, túto sme presunuli do repozitárov. Repozitáre a kontroléry komunikujú s databázou a požiadavku vrátia s výsledkom naspäť na klienta.

Implementačné prostredie

- Rámec Quasar (tučný klient, založený na Vue.js MIT), ktorý umožňuje rýchlo a efektívne vytvárať responzívne webové stránky/aplikácie
- Rámec AdonisJS - Služby biznis logiky
- PostgreSQL - relačný databázový systém

Návrhové rozhodnutia

Pre vývoj projektu sme použili:

- Jazyk TypeScript
- Nodejs verzia 16.14.0
- Vue.js verzia 3

V rámci klientskej časti sme doinštalovali nasledujúce prerekvizity:

- adonis-socket.io - podpora pre websockety
- axios (defaultne pri inite quasaru)
- vuex (defaultne pri inite quasaru)

V rámci server časti sme doinštalovali nasledujúce prerekvizity:

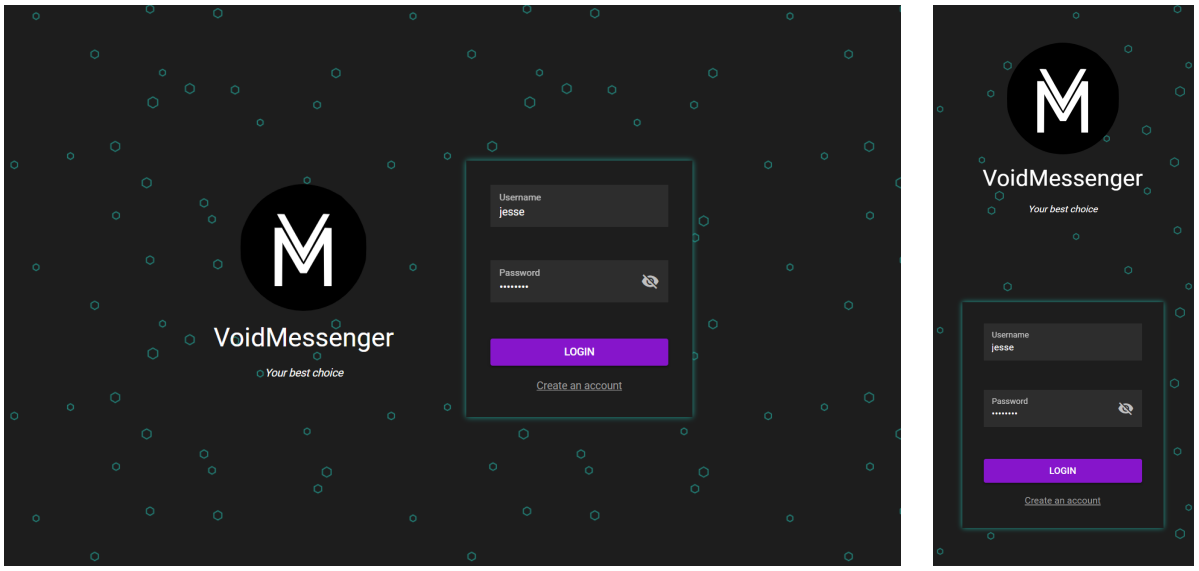
- @adonisjs/lucid - ORM SQL
- @adonisjs/auth - balíček pre autentifikáciu
- phc-argon2 - hashovanie hesiel
- adonis-socket.io - podpora pre websockety

Bonusy

- pre notifikácie sme použili Notification API, namiesto Quasar notifikácií
- pri PWA sme zabezpečili to, aby sa pri strate konektivity na server zobrazil login page, ale v package.json sme zriadili aj SPA verziu, pretože PWA nám z nevysvetliteľných dôvodov neustále refreshuje

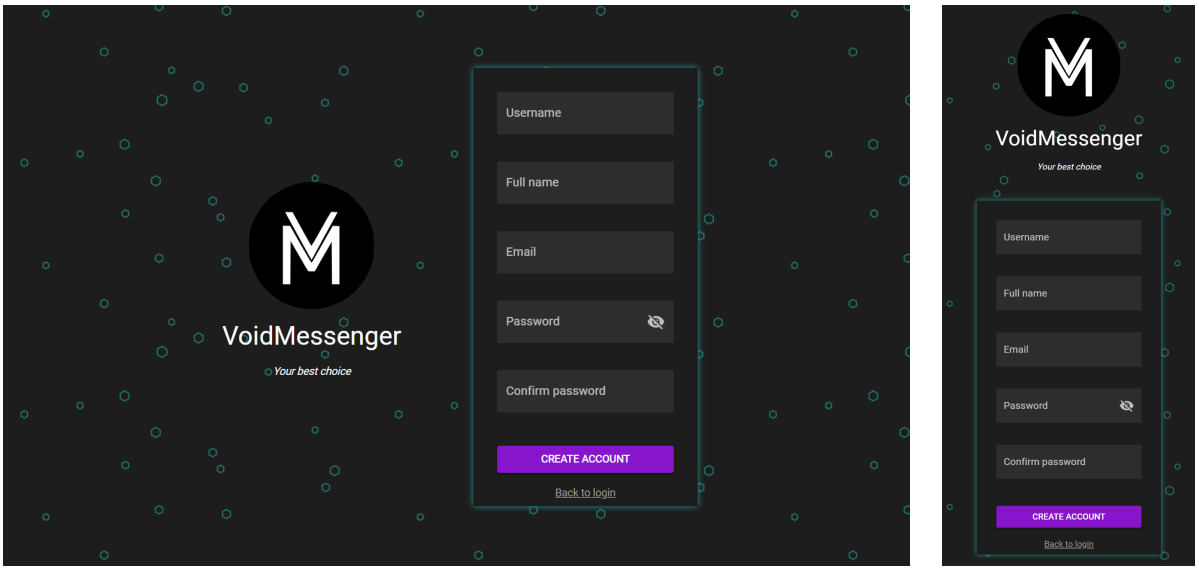
Snímky obrazoviek

Prihlásenie



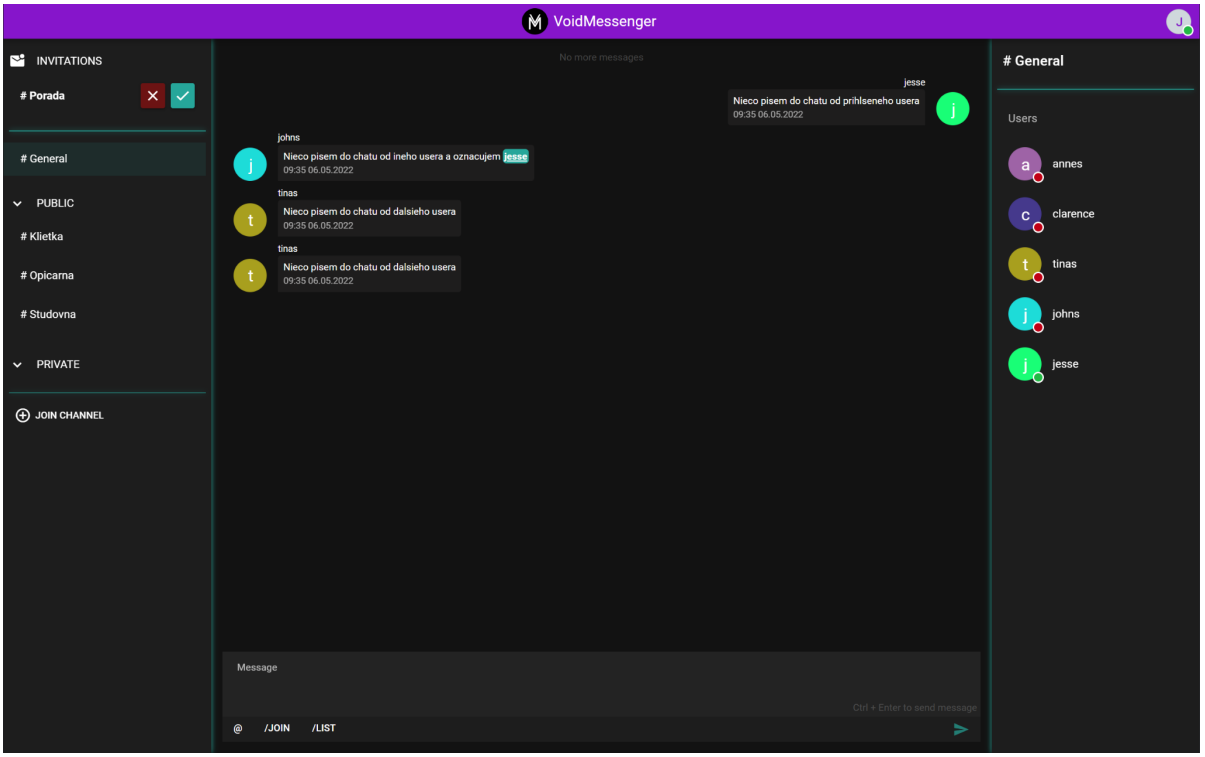
Obrázok 3: Prihlasovanie, desktop a telefón

Registrácia

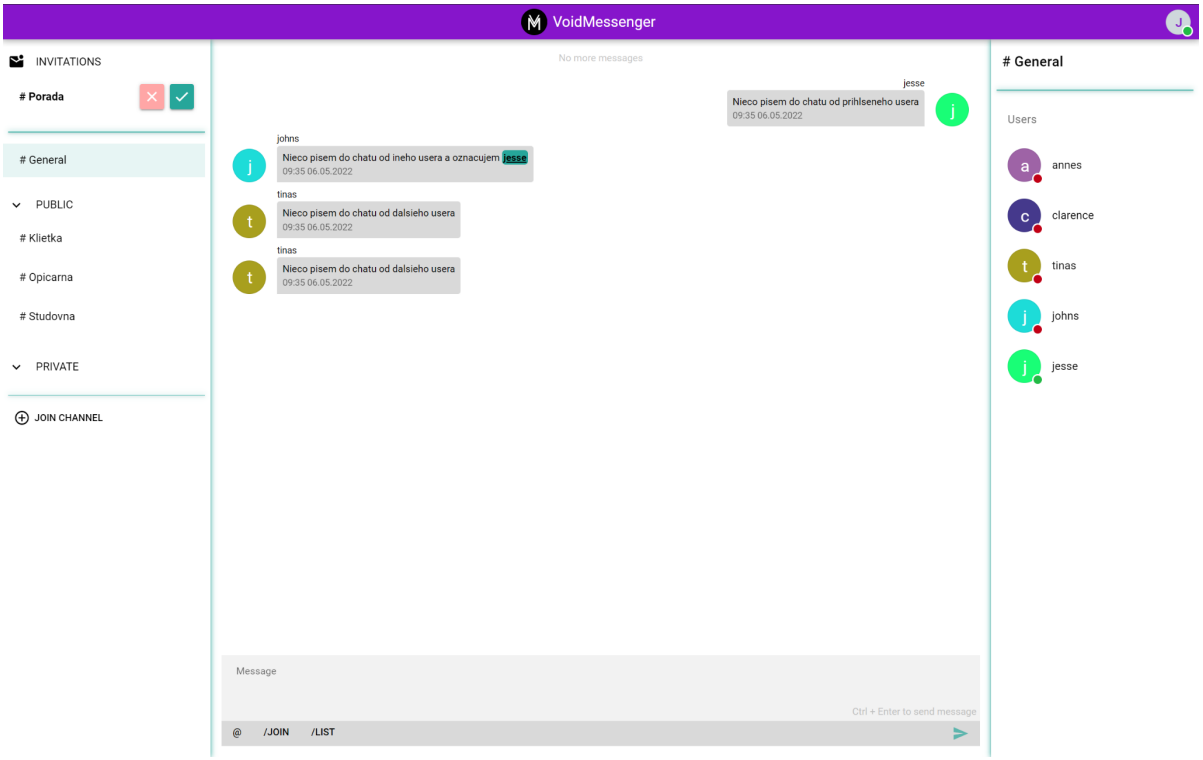


Obrázok 4: Registrácia, desktop a telefón

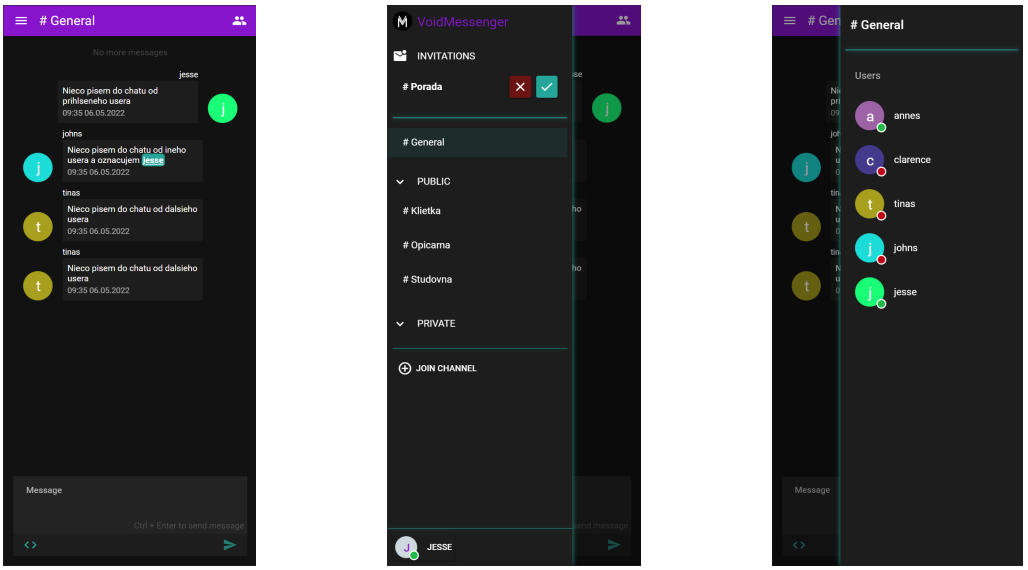
Domovská obrazovka



Obrázok 5: Homepage, čierna téma, desktop

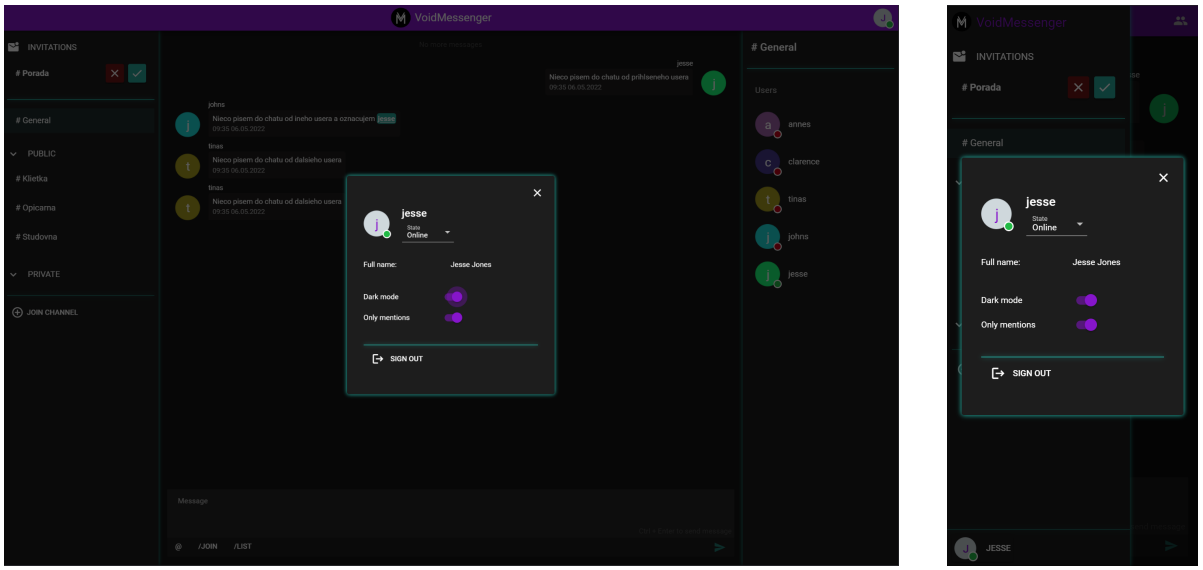


Obrázok 6: Homepage, biela téma, desktop



Obrázok 7: Homepage, čierna téma, telefón

Nastavenia používateľa



Obrázok 8: Nastavenia používateľa, desktop a telefón