Zadanie č. 4

Vypracovali: Gurbaľová T., Kridlová R., Revaj J., Ševčíková E., Šurab L.

1 Rozšírenie databázového modelu podľa zadania

V tejto časti zadania sme rozšírili našu databázu o tabuľky **PoliticalParty**, **EncryptedVote**, **EncryptedVoteServer** a **Vote**.

1.1 PoliticalParty

Tabuľka **PoliticalParty** obashuje polia **id, name, description** a **attributes**. Pole id je typu **AutoField** a predstavuje primárny kľúč tejto tabuľky. Pole name predstavuje meno strany a musí byť unikátne pre každú stranu. Pole description obsahuje popis samotnej strany a attributes je list, v ktorom sa nachádzajú tagy pre jednotlivé strany (či je strana liberálna, konzervatívna, sociálna, pravicová či ľavicová) podľa ktorých sa dá vyhľadávať.

```
class PoliticalParty(models.Model):
    id = models.AutoField(primary_key=True, editable=False)
    name = models.CharField(max_length=255, unique=True)
    description = models.TextField()
    attributes = ArrayField(models.CharField(max_length=50), blank=True, null=True)

    Lukas Surab
    def __str__(self):
        return self.name
```

Figure 1: PoliticalParty model

1.2 EncryptedVote a EncryptedVoteServer

Tieto 2 tabuľky sú čo sa polí týka identické, no do jednej sa ukladajú dáta zašifrované používateľským verejným kľúčom a do druhej serverovským. Tabuľky obashujú polia **id, data, key** a **timestamp**. Tieto polia predstavujú id konkrétneho zašifrovaného hlasu, ktorý sa vytvorí pri samotnom hlasovaní, zašifrané data aes kľúčom a zašifrovaný aes kľúč v jednom prípade používateľovým verejným kľúčom a v druhom prípade serverovským. Ako posledné sa pridá timestamp, ktoré je non-editable a predstavuje čas vytvorenia hlasu.

```python
class EncryptedVote(models.Model):
    id = models.AutoField(primary_key=True, editable=False)
    data = models.BinaryField(editable=False)
    key = models.BinaryField(editable=False)
    timestamp = models.DateTimeField(default=timezone.now, editable=False)

    Lukas Surab
    def __str__(self):
        return str(self.id)

new *
class EncryptedVoteServer(models.Model):
    id = models.AutoField(primary_key=True, editable=False)
    data = models.BinaryField(editable=False)
    key = models.BinaryField(editable=False)
    timestamp = models.DateTimeField(default=timezone.now, editable=False)

    new *
    def __str__(self):
        return str(self.id)
```

FIGURE 2: Encrypted votes models

1.3 Vote

Posledná tabuľka predstavuje samotný dešifrovaný hlas pre sčítanie a obsahuje id, meno samotnej volenej strany a timestamp, ktorý sa kopíruje pri dešifrovaní hlasov z tabuľky EncryptedVoteServer. Dešifrovanie hlasov prebieha na backende na endpointe **/api/users/dec**.

```python
class Vote(models.Model):
    id = models.AutoField(primary_key=True, editable=False)
    party_name = models.TextField(editable=False, null=True)
    timestamp = models.DateTimeField(default=timezone.now, editable=False)

    Lukas Surab
    def __str__(self):
        return str(self.id)
```

FIGURE 3: Vote model

2 Doimplementovanie funkcionalít

Do našej webaplikácie sme doimplementovali samotné hlasovanie, ktorého funkcionalita sa dá nájsť v **backend/volby_backend/users**, po ktorého prebehnutí sa používateľovi stiahne pdf s potvrdením o zahlasovaní aj so zvolenou stranou. Toto pdf je zašifrované používateľovým verejným kľúčom a dá sa dešifrovať na stránke pomocou používateľovho súkromného kľúča. Pre túto operáciu nepotrebuje byť používateľ prihlásený. Používateľ si dokáže takisto zmeniť svoj email (ostatné osobné údaje nám neprišli ako dôležité pri

voľbe online a preto sa nedajú meniť nakoľko jediná ďalšia možnosť by bola heslo a tá nám pri voľbách, kde sa treba prihlásiť práve jedenkrátF neprišla ako relevantná). Pri hlasovaní sa vytvoria dve inštancie hlasu (kde kľúč je zašifrovaný serverovským verejnýn kľúčom a kde kľúč je zašifrovaný verejným používateľovým kľúčom). Taktiež po zahlasovaní používateľ stratí právo hlasovania a toto pole sa stane non-editable. Hlasy sa dajú dešifrovať a nahrať do tabuľky hlasov na endpointe /dec, pre ktorý sa používa funkcia decrypt_and_save_votes v **backend/volby_backend/users**. Pri dešifrovaní hlasov sa dešifrujú na serveri zašifrované hlasy a vytvoria sa do tabuľky Vote. Správy sú šifrované pomocou funkcií symmetric_encrypt a symmetric_encrypt_user z predchádazjúcich zadaní. Na endpointe **https://localhost/decrypt** si vie používateľ následne svoje potvrdenie dešifrovať potom, čo nahrá svoj súkromný kľúč a celý obsah PDF súboru.

3 Vyhľadávanie

Vyhľadávanie je na frontende implementované pre všetky polia, tj. dá sa vyhľadávať podľa textu, názvu strany aj tagov. Pri načítaní stránky je získaný z backendu zoznam strán, ktorý je zobrazený pod vyhľadávaním. Keďže zoznam strán je v našom prípade malý, vyhľadávanie sa vykonáva len na strane klienta.

4 Používanie aplikácie

Naša aplikácia pre voľby sa používa v ľahkých a intuitívnych krokoch. Ako prvé nás stránka zavedie na prihlasovací endpoint, z ktorého vieme prejsť priamo na registráciu. Pri registrácii musíme do kolóniek číslo OP a rodné číslo použiť jednu z už predpripravených kombinácií aby sme zabezpečili, že používateľ má naozaj právo pre online voľby (tieto kombinácie nahrávame zo súboru **document_fixture.json** v backend priečinku), a teda sa do nich predom zahlásil. Okrem toho si zvolí ešte heslo a email, ktoré slúžia aj na prihlásenie. Nasledujú endpointy usporiadané do tabov, pre jednoduchšie prechody medzi nimi. Po prihlásení je používateľ presmerovaný na endpoint s katalógom strán, v ktorom môže vyhľadávať a prečítať si popisy o jednotlivých stranách. Okrem toho sa tu nachádzajú taby pre samotné hlasovanie a manažmenť účtu. V manažmente účtu si vie používateľ zmeniť používateľský email, ako aj nahrať vlastný kľúčový pár či vygenerovať nový. Po vygenerovaní nového kľúčového páru sa verejný kľúč priradí k používateľovi do databázy a súkromný kľúč si môže stiahnúť (čo bude potrebovať pre možnosť dešifrovania volebného potvrdenia). Ak chce používateľ vložiť kľúč v nesprávnom formáte, je na to upozornený error message-om a kľúč nie je vložený. Samotné zahlasovanie nie je možné bez toho aby mal používateľ vygenerované kľúče. Po zahlasovaní sa používateľovi stiahne PDF súbor so zašifrovaným potvrdením

o zahlasovaní. Na endpointe **https://localhost/decrypt** používateľ nahrá svoj či už vygenerovaný alebo vložený súkromný kľúč, do poľa pre text vloží samotný obsah PDF a súbor dešifruje. Ak je kombinácia kľúča a textu, ktorý vie dešifrovať dostane chybovú hlášku o nesprávnom formáte. Ak je kombinácia správna, na spodu sa zjaví výsledok dešifrovania s menom politickej strany, ktorú zvolil. Nakoľko sa jedná o volebný systém, je tento systém založený na jedinom možnom prístupe. Po zahlasovaní sa už nie je možné do systému opäť prihlásiť (Refresh token bude blacklisted a access tokenu sa nastaví dĺžka na 1s).

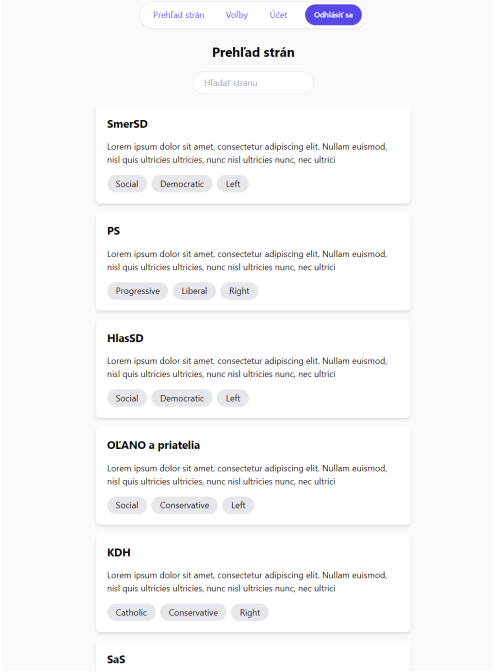

FIGURE 4: Katalóg strán

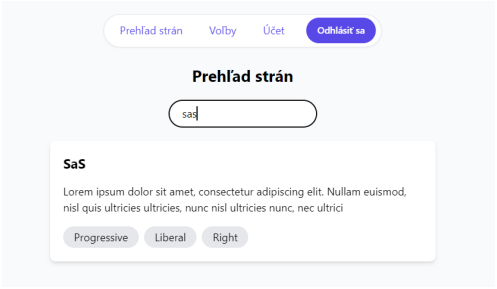

FIGURE 5: Filtrovanie podľa strany

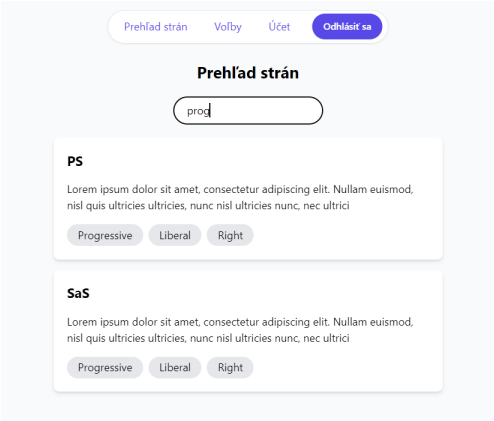

FIGURE 6: Filtrovanie podľa tagu

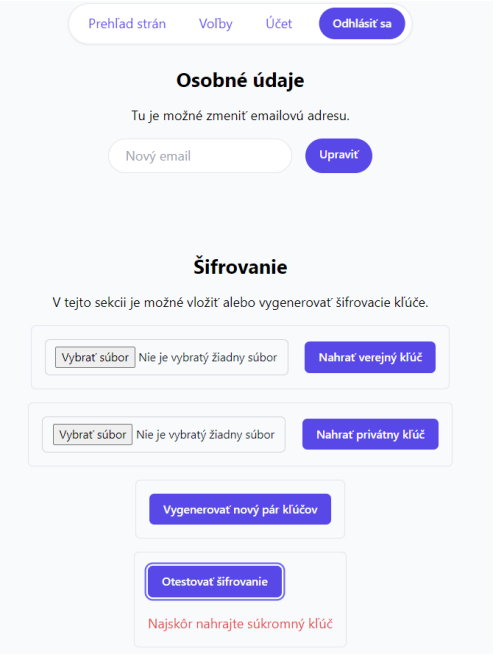

FIGURE 7: Chybová hláška bez nahratia kľúča

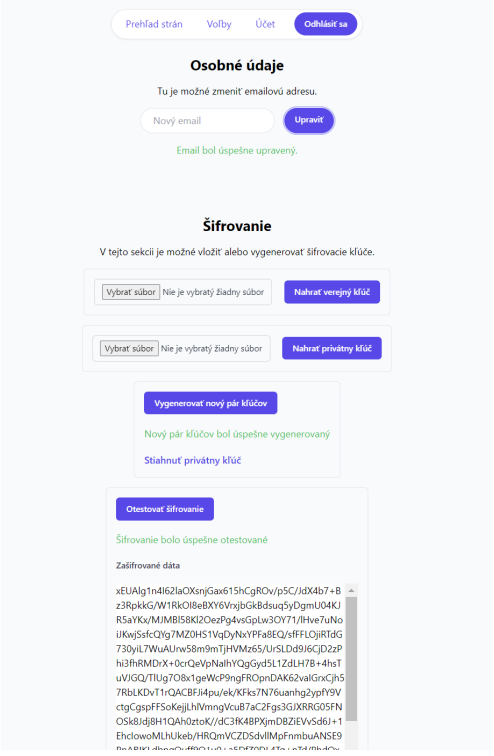

FIGURE 8: Úspešná zmema mailu

FIGURE 9: Endpoint pre hlasovanie

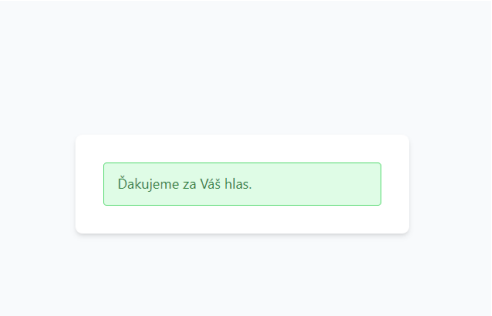

FIGURE 10: Úspešné zahlasovanie

5 Využívané porty

Samotná aplikácia beží na porte **443**. Okrem toho používame pre **backend** port **8000** a pre **frontend** port **3000**. Pre prístup na backendové endpoiny používame cestu **/api/** a pre frontendové endpointy samotnú koreňovú cestu. Testovacie ČOP a rodné číslo sú napríklad **EV42156** a **4312686543**.