

# Reflectus alternatyve

Frontend UX ir refleksiju formu tobulinimas – pokalbio irasu PDF

Data: 2026-01-04

Kalba: LT/EN (kaip pokalbyje)

Turinys:

1. Santrauka
2. Pokalbio transkriptas
3. Pateikti ekrano vaizdai (priedas)

## Santrauka

Pokalbyje suformuota kryptis, kaip sutvarkyti Reflectus frontend: 1) sutvarkyti navigacija (visi mygtukai turi vesti i realius route), 2) sutvarkyti globalu layout (turinys centre, nera prilipimo prie krasto), 3) pagerinti refleksiju formu UX (kortelles, spacing, 1-5 ivertinimai kaip pasirinkimai), 4) patobulinti refleksiju klausimus (pamokos, savaites, kontrolinio, projekto, savijautos). Taip pat aptarta, kaip pushinti projekta i GitHub per VS Code terminal.

## Pokalbio transkriptas

### User

sukuriam dbar frontend pilnai. daug geriau. reikia user friendly padaryt. i6kark sukurk pagrindines kokias refleksijas.

### Assistant

Žemiau yra pilnas, veikiantis, user-friendly frontend (React + Vite + Tailwind) su:

- Login (role: student/teacher)
- Student UI: "Nauja refleksija", formos su progresu, autosave, istorija
- Teacher UI: klasės peržiūra, filtrai, refleksijos peržiūra + komentaras
- Pagrindinės refleksijų šablonoai (pamokos, savaitės, kontrolinio, projekto, savijautos)
- Veikia ir be backend: Mock režimas (localStorage). Kai backend bus – perjungiamas per VITE\_API\_URL.

---

## 1) Sukuriam frontend (Ubuntu Terminal)

Atidaryk Ubuntu Terminal ir vykdyk (projekto šaknyje):

```
cd ~/projektai/reflect # pakeisk į savo repo kelią
npm create vite@latest frontend -- --template react
cd frontend
npm i

npm i react-router-dom zustand lucide-react react-hook-form zod @hookform/resolvers
npm i -D tailwindcss postcss autoprefixer
npx tailwindcss init -p
```

---

## 2) Tailwind konfigūracija

```
### tailwind.config.js (pakeisk turinį)
/** @type {import('tailwindcss').Config} */
export default {
  content: ["./index.html", "./src/**/*.{js,jsx}"],
  theme: { extend: {} },
  plugins: [],
};

### src/index.css (pakeisk turinį)
```

```
@tailwind base;
@tailwind components;
@tailwind utilities;

:root { color-scheme: light; }
html, body, #root { height: 100%; }
```

---

## 3) Failų struktūra (sukurk VS Code)

Sukurk šituos failus:

```
src/
App.jsx
main.jsx
router.jsx
data/templates.js
lib/storage.js
lib/mockApi.js
lib/api.js
stores/authStore.js
components/
ui.jsx
Layout.jsx
ProtectedRoute.jsx
ReflectionForm.jsx
pages/
Login.jsx
student/
StudentHome.jsx
NewReflection.jsx
FillReflection.jsx
History.jsx
teacher/
Teacher
```

#### User

darbu plana codexui

#### Assistant

Žemiau yra darbo planas Codex/Copilot'ui (kad galėtum kopijuoti promptus į VS Code Copilot Chat / Codex ir jis generuotų kodą tvarkingai). Planas daromas taip, kad veiktu be backend (mock + localStorage), o vėliau tik pakeisi api.js į realius endpoint'us.

---

## 0) Tikslas ir taisyklės (viena tiesa visur)

User flow:

- Login su mokyklos duotais prisijungimais.
- Rolė (student/teacher) nustatoma pagal prisijungimą (vėliau backend atsakymas).
- Student: kuria refleksiją pagal šabloną, pildo, pateikia, mato istoriją.
- Teacher: mato mokinii refleksijas, filtruoj, atsidaro, palieka komentarą.

Technologijos:

- React + Vite + Tailwind
- react-router-dom
- zustand (auth + app state)
- react-hook-form + zod (formoms)
- localStorage mock režimui
- 1 vieta API: src/lib/api.js (vėliau prijungiam backend)

---

## 1) Repo ir paleidimas (Ubuntu Terminal)

Ubuntu terminale (projekto šaknyje):

```
cd ~/projektai/reflect  
cd frontend  
npm i  
npm run dev
```

---

## 2) Milestone'ai (ką turi matyti po kiekvieno)

M1: veikia routing + UI layout + login puslapis  
M2: auth store + protected routes + role-based meniu  
M3: reflection templates + student dashboard + new reflection  
M4: reflection form (autosave, progress, validate) + submit  
M5: student history + detail view  
M6: teacher dashboard (list+filters) + detail + comment  
M7: api wrapper + mockApi + perjungimas į real API per .env  
M8: UI polishing (empty states, error states, mobile) + build

---

## 3) Codex prompt'ai (kopijuok po vieną, eilės tvarka)

... (turinys tėsiamas šiame pokalbyje)

User

how to push to github

Assistant

### Push to GitHub (Ubuntu Terminal)

```
### 1) Go to your project folder  
cd ~/projektai/reflect # change to your real folder  
### 2) Verify it's a git repo + see changes  
git status
```

If it says “not a git repository”, you’re in the wrong folder (go one level up/down and try again).

### 3) Make sure the GitHub remote exists

```
git remote -v
```

If you see origin ...github.com... → OK.

If empty, add it (replace with your repo URL):

```
git remote add origin https://github.com/USERNAME/REPO.git
```

### 4) Add files + commit

```
git add -A  
git commit -m "Initial frontend"
```

### 5) Push

Check your branch name

User

vscode terminal

Assistant

## Push to GitHub using VS Code Terminal (Ubuntu)

### 1) Open the correct folder in VS Code

- VS Code → File → Open Folder...

- Choose your project folder (the one that has frontend/, backend/, etc.)

### 2) Open VS Code Terminal

- Top menu: Terminal → New Terminal

- It opens at the project folder (usually). Verify:

```
pwd  
ls
```

### 3) Check git status

```
git status
```

If you see “not a git repository”, you opened the wrong folder. Open the folder that contains .git.

### 4) Add remote (only if needed)

Check:

```
git remote -v
```

If nothing is shown, add your GitHub repo URL:

```
git remote add origin https://github.com/lmcorpofficial/reflect.git
```

### 5) Add → commit → push

```
git add -A  
git commit -m "Update frontend"
```

Now push (try one of these, depending on branch name):

```
git push -u origin main
```

If that fails because your branch is master:

```
git push -u origin master
```

If you don't know the branch:

```
git branch
```

The one with \* is the current branch.

### 6) If GitHub asks for login/password

GitHub no longer accepts account passwords for git. Use Personal Access Token (PAT) as the password.

Quick setup (recommended):

```
sudo apt update  
sudo apt install -y gh
```

```
gh auth login
```

Then push again:

```
git push -u origin main
```

If you paste the exact error output from the terminal, I'll tell you the exact fix for your case.

#### User

Uploaded screenshots. turim struktūra. bet neatrodo profesionalai. ir tie mygtukai nevisi veikai ypač pas mokytojus ir mokinius. refleksiju

#### Assistant

### Problema pagal screenshot'us

- 1) UI "išsitempės" per visą ekraną, daug tuščios erdvės, tipografija/spacing nesuderinti → neatrodo profesionaliai.
- 2) "Kortelės-mygtukai" dalis jų greičiausiai yra tik <div> be realaus Link/onClick, arba veda į neegzistuojantį route.
- 3) Refleksijų srautas néra pilnai sujungtas: sukurti → išsaugoti → atsiranda istorijoje / mokytojui peržiūroje.

Žemiau – konkretus Codex darbo planas, kad sutvarkytų UI + navigaciją + refleksijų veikimą.

---

## 0) Greitas audit (VS Code terminal)

```
cat package.json
```

---

## 1) Navigacija: sutvarkyti VISUS mygtukus, kad jie realiai veiktu

... (planas tēsiamas)

#### User

log in puslapis. man nepatinka kad jis negražus ir kad jis taip sugrūstas... reikia userfriendly. 1 refleksijos kūrimas is mokinnio puses... toliau negaliu nieko daryti nes mygtukai paspausti nieko nedaro

#### Assistant

Žinau, ką matai: UI yra „susigrūdės“ kairėje, o likusi ekrano dalis tuščia. Ir svarbiausia – kortelės/mygtukai yra „dead click“ (paspaudi ir nieko nevyksta). Reikia sutvarkyti dviem etapais: (A) kad viskas būtų paspaudžiamai ir vestų į puslapius, tada (B) profesionalus layout.

Žemiau – Codex darbo planas su aiškiais rezultatais.

---

## A. 1 prioritetas: sutvarkyti, kad VISI mygtukai veiktu (navigacija)

... (planas tēsiamas)

User

MATAI PADARKYK KAD JOS NORMALIAI BUTU... PABANDYK PAGERINTI KALSIUMUS.  
SUTVARKOM DIZAINA... IŠ MOKINIO PUSĖS. PAKEITIMAI

Assistant

Žiūriu į ekranus: problema ne funkcijose, o layout'e ir UI komponentuose. Viskas prilipę prie kairio krašto, formos "išsitiesia" neteisingai, kai kur tekstas "išlenda", o 1-5 vertinimai yra kaip paprasti input'ai (ne user-friendly).

Žemiau – konkretus pakeitimų planas "iš mokinio pusės" + pagerinti klausimai.

---

1) Globalus UI standartas: nieko prie krašto, viskas centre

... (toliau – dizainas ir klausimų patobulinimai)

## Priedas – pateikti ekrano vaizdai

Toliau pateikiami pokalbyje ikelti ekrano vaizdai, naudoti UI/UX problemoms iliustruoti.

Screenshot 1: Login puslapis (demo režimas).

REFLECTUS

### Sveiki atvykė į refleksiju sistemą

Mokyklos refleksijų platforma mokiniams ir mokytojams.  
Apmasykite savo mokymąsi ir sekite pažanga.

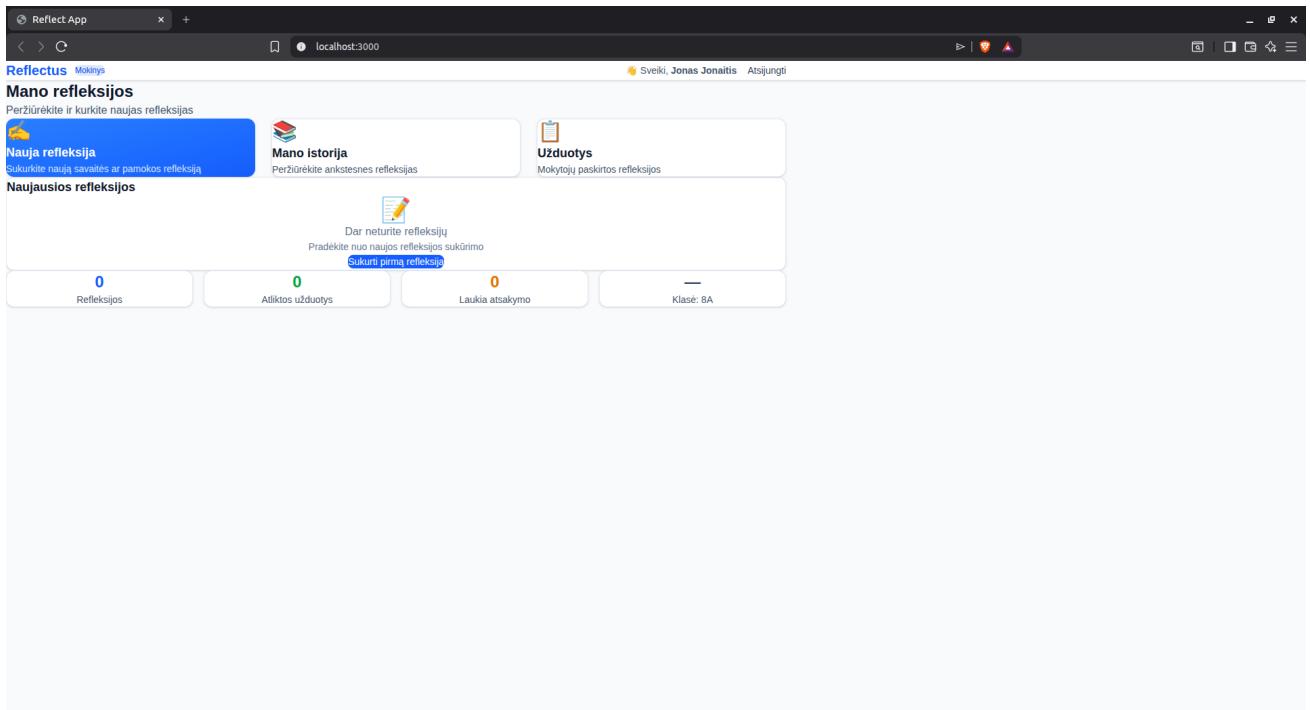
Greitas prisijungimas (demo):  
Mokinys: Mokytojas  
Slaptažodis: test123  
Demo režimas aktyvus

Prisijungimas  
Iveskite el. paštą ir slaptažodį  
El. paštas: mokinys@pastas.lt  
Slaptažodis: \*\*\*\*\*

Prisijungi

Turite problemų? Kreipkitės į mokyklos administratorių.

Screenshot 2: Mokinio skydelis (dashboard).



## Reflectus – pokalbio irasu PDF

Screenshot 3: Mokytojo skydelis (dashboard).

Nauja užduotis  
Sukurkite refleksijos užduotį klasei

Mano klasės  
Valdykite mokiniai grupės

Statistika  
Mokiniai refleksijų analizė

0 naujos

Laukia peržiūros

Visos refleksijos peržiūrėtos!  
Čia bus rodomas mokiniai pateiktos refleksijos

+ Pridėti klasę  
Peržiūrėti ...  
Peržiūrėti ...

**Mano klasės**

**8A** 1 klasė  
1 mokinys

**7B** 0 mokiniai

**2** Klasės    **1** Mokiniai    **0** Aktyvios užduotys    **0** Refleksijos Ši mėn.

Screenshot 4: Login puslapio išdėstymas (problema: per daug tuščios vietos, elementai susigrūdė).

REFLECTUS

**Sveiki atvykė į refleksijų sistemą**

Mokyklos refleksijų platforma mokiniams ir mokytojams.  
Apmasykite savo mokymąsi ir sekite pažanga.

Geriausias prisijungimas (demo):  
Mokinys: Mokytojas  
Slaptažodis: test123  
Demo režimas aktyvus

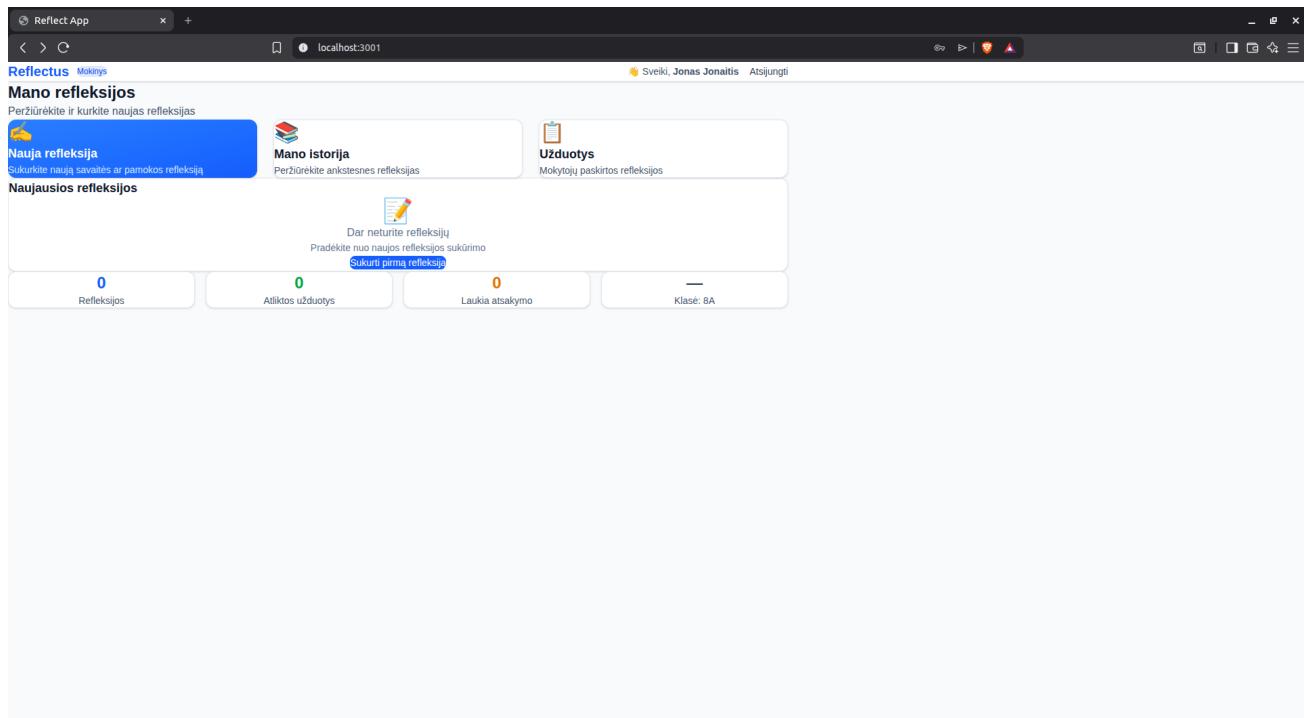
**Prisijungimas**  
Įveskite el. paštą ir slaptažodį  
El. paštas: mokinys@pastas.lt  
Slaptažodis: \*\*\*\*\*

**Prisijungi**

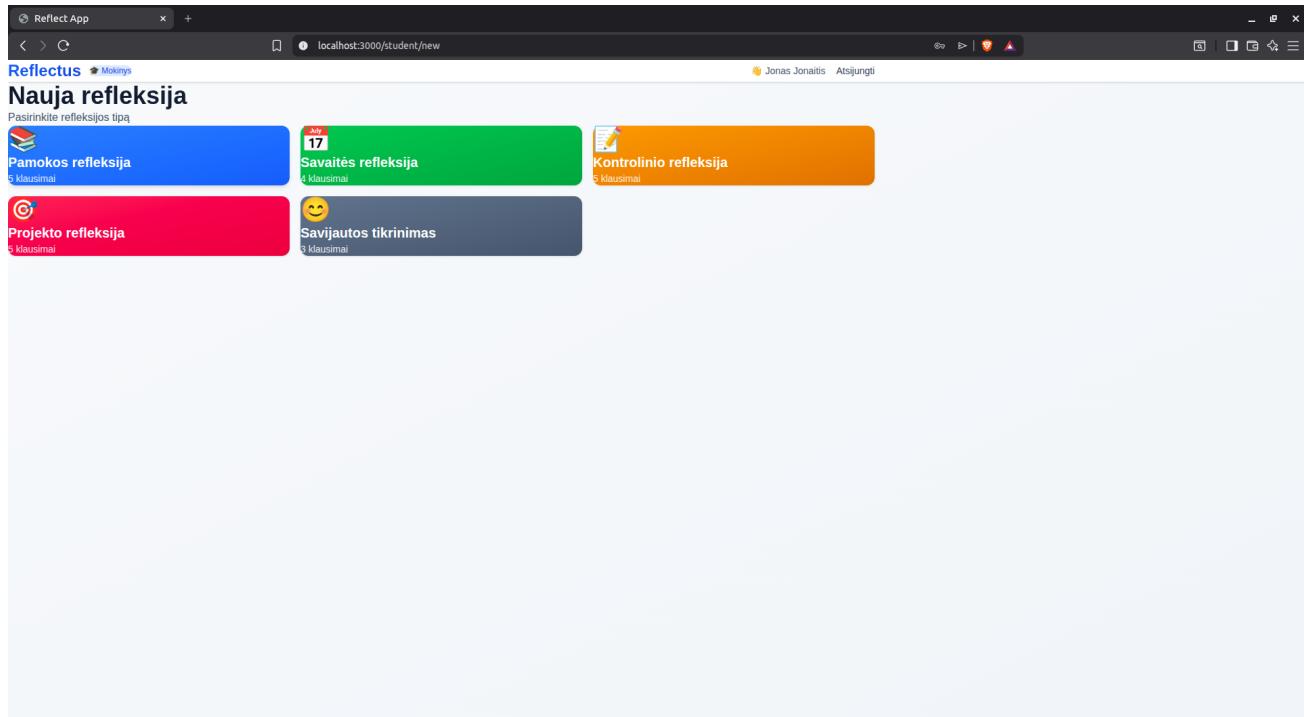
Turite problemų? Kreipkitės į mokyklos administratorių.

## Reflectus – pokalbio irasu PDF

Screenshot 5: Mokinio skydelis – pagrindiniai mygtukai ir naujausios refleksijos (problema: dalis mygtukų neveikia).



Screenshot 6: Refleksijos tipo pasirinkimas (student/new).



## Reflectus – pokalbio irasu PDF

Screenshot 7: Pamokos refleksija forma.

The screenshot shows a web browser window titled 'Reflect App' at the URL 'localhost:3000/student/new/lesson'. The page is titled 'Pamokos refleksija' (Lesson reflection). It contains several text input fields:

- 'Dalykas\*' (Subject): 'Iveskite dalykas...' (Enter subject...)
- 'Tema\*' (Topic): 'Iveskite tema...' (Enter topic...)
- 'Ka išmokau?' (What did I learn?): 'Iveskite ka išmokau?...' (Enter what I learned...)
- 'Kas buvo sudėtinga?' (What was difficult?): 'Iveskite kas buvo sudėtinga?...' (Enter what was difficult...)
- 'Kokie klausimai liko?' (What questions remain?): 'Iveskite kokie klausimai liko?...' (Enter what questions remain...)

At the bottom, there are two buttons: 'Atšaukti' (Cancel) and 'Pateikti refleksiją' (Submit reflection).

Screenshot 8: Savaitės refleksija forma.

The screenshot shows a web browser window titled 'Reflect App' at the URL 'localhost:3000/student/new/week'. The page is titled 'Savaitės refleksija' (Week reflection). It contains several text input fields:

- 'Užpildykitė visus laukus' (Fill in all fields): 'Savaitės pasiekimai' (Week's achievements), 'Iveskite savaitės pasiekimai...' (Enter week's achievements...)
- 'Sunkumai' (Challenges): 'Iveskite sunkumai...' (Enter challenges...)
- 'Tikslai kitai savaitei\*' (Goals for next week): 'Iveskite tikslai kitai savaitei...' (Enter goals for next week...)
- 'Nuotaika (1-5)\*' (Mood (1-5)): '1 - 5', followed by a scale from 1 to 5.

At the bottom, there are two buttons: 'Atšaukti' (Cancel) and 'Pateikti refleksiją' (Submit reflection).

## Reflectus – pokalbio irasu PDF

Screenshot 9: Kontrolinio refleksija forma.

The screenshot shows a web browser window titled 'Reflect App' with the URL 'localhost:3000/student/new/test'. The page is titled 'Kontrolinio refleksija'. It contains several text input fields:

- 'Užpildykitė visus laukus'
- 'Dalykas\*' (with placeholder 'Įveskite dalykas...')
- 'Kaip ruošiaus?'\* (with placeholder 'Įveskite kaip ruošiaus?...')
- 'Kas sekėsi?' (with placeholder 'Įveskite kas sekėsi?...')
- 'Kas nesisekė?' (with placeholder 'Įveskite kas nesisekė?...')
- 'Kaip galėčiau tobulinti?'\* (with placeholder 'Įveskite kaip galėčiau tobulinti?...')

At the bottom, there are two buttons: 'Atšaukti' (Cancel) and a blue 'Pateikti refleksiją' (Submit reflection) button.

Screenshot 10: Projekto refleksija forma.

The screenshot shows a web browser window titled 'Reflect App' with the URL 'localhost:3000/student/new/project'. The page is titled 'Projekto refleksija'. It contains several text input fields:

- 'Užpildykitė visus laukus'
- 'Projekto pavadinimas\*' (with placeholder 'Įveskite projekto pavadinimą...')
- 'Mano vaidmuo\*' (with placeholder 'Įveskite mano vaidmą...')
- 'Ko išmokau?'\* (with placeholder 'Įveskite ko išmokau?...')
- 'Kaip sekėsi komandinis darbas?' (with placeholder 'Įveskite kaip sekėsi komandinis darbas?...')
- 'Ką daryt kitą kartą kitaip?'\* (with placeholder 'Įveskite ką daryt kitą kartą kitaip?...')

At the bottom, there are two buttons: 'Atšaukti' (Cancel) and a blue 'Pateikti refleksiją' (Submit reflection) button.

Screenshot 11: Savijautos tikrinimas forma.

The screenshot shows a web browser window titled 'Reflect App' with the URL 'localhost:3000/student/new/mood'. The page is titled 'Savijautos tikrinimas' (Self-assessment) and has a sub-instruction 'Užpildykitė visus laukus' (Fill in all fields). There are three required fields: 'Nuotaka (1-5)\*' (Time spent (1-5)), 'Energijos lygis (1-5)\*' (Energy level (1-5)), and 'Mintys' (Thoughts), each with a text input field. Below these is a large text area labeled 'Įveskite mintys...' (Enter thoughts...). At the bottom left is a blue button labeled 'Atšaukti' (Cancel) and at the bottom right is a blue button labeled 'Pateikti refleksija' (Submit reflection).