

Reflectus alternatyvė

Reflektavimo programėlės koncepcija, funkcijos, analitika, saugumas ir MVP planas

Dokumento data: 2025-12-30

Versija: v1.0 (sudaryta iš šio pokalbio reikalavimų ir idėjų)

-
- Tikslas: sukurti reflektavimo įrankį, kuris ne tik renka atsakymus, bet ir paverčia juos statistiškai analizuojamais duomenimis bei pateikia tendencijas laike.
 - Išskirtinumas: atviri atsakymai apdorojami AI, paverčiami struktūromis (temos, emocijos, pasiūlymai), todėl galima daryti grafikus, grupines žžvalgas ir koreliacijas.
 - Naudojimo scenarijai: mokykla (pamokos/kontroliniai/projektai), susirinkimai, komandos darbas, asmeninis reflektavimas.

Turinys

- 1. Vizija ir sprendžiama problema
- 2. Naudotojai, rolės ir teisių modelis
- 3. Klausimynų modelis ir klausimų tipai
- 4. Universalūs pasirinkimai: „Nenoriu atsakyti“ ir „Nežinau“
- 5. Refleksijų planavimas, galiojimas, priminimai ir grupės prisijungimas
- 6. Duomenų apdorojimas ir AI (tekstų sužymėjimas į struktūrą)
- 7. Statistika: individas, grupė, laikas, palyginimai
- 8. Vizualizacijos: rekomenduojami grafikai pagal klausimo tipą
- 9. Koreliacijos ir ryšių analizė tarp klausimų
- 10. Eksportas: CSV/Excel, Google Sheets, PDF ataskaitos
- 11. Valdymo skydelis: kalendorius, aktyvumas, kontrolė ir redagavimas
- 12. Saugumas, privatumas, atitiktis (GDPR)
- 13. MVP apimtis, techninė architektūra, rizikos

1. Vizija ir sprendžiama problema

Tradicinės refleksijos sistemos dažniausiai renka tekstinius atsakymus, tačiau jų nepaverčia aiškiai išmatuojamais duomenimis. Dėl to sunku pastebėti dėsningumus: kas kartojasi, kas blogėja/gerėja, kas susiję tarpusavyje. „Reflectus alternatyvė“ siekia sujungti refleksiją ir analitiką: klausimai gali būti įvairių tipų, o atviri tekstai apdorojami taip, kad įmanoma statistika, grafikai ir įžvalgos laike.

Pagrindiniai principai

- Duomenys turi būti palyginami laike (tas pats klausimas, tas pats kontekstas, aiškus galiojimo langas).
- Sistema turi padėti surinkti refleksijas laiku (planavimas, galiojimas, priminimai, paprastas prisijungimas).
- Atviri atsakymai turi virsti struktūra (tema, emocija, pasiūlymo tipas, santrauka).
- Tiek individas, tiek grupė turi matyti aiškius grafikus ir tendencijas.
- Saugumas ir privatumas yra „default“, ypač mokyklos kontekste.

2. Naudotojai, rolės ir teisių modelis

Klausimynus gali kurti ne tik pats naudotojas. Juos gali kurti ir priskirti kiti asmenys: mokytojas mokiniams, tėvai vaikui, vadovas komandai, treneris sportininkams ir pan.

Rolės (pavyzdinis rinkinys)

- Savininkas (Owner): sukuria klausimyną, valdo grupes, mato statistiką, eksportuoja.
- Atsakantis (Respondent): pildo refleksijas, mato savo statistiką (pagal taisykles).
- Stebėtojas (Viewer): mato rezultatus (pvz., tik grupės agregatus).
- Komentatorius (Commenter): gali palikti komentarus prie refleksijos/ižvalgų.
- Redaktorius (Editor): gali siūlyti klausimų pakeitimus (su patvirtinimu).
- Administratorius (Admin): valdo organizacijos/mokyklos politiką, teises, log'us.

Bendradarbiavimo logika (kaip Google dokumentai)

- Klausimyno pakeitimai gali būti „pasiūlymai“ (suggestions), kuriuos savininkas patvirtina arba atmeta.
- Kiekvienas pakeitimas turi audit logą: kas, kada, ką pakeitė.
- Jei klausimynas jau turi duomenų, keičiant klausimą sistema turi aiškiai įspėti apie poveikį statistikai.

3. Klausimynų modelis ir klausimų tipai

Klausimynas yra kelių klausimų rinkinys. Klausimas turi tipą, nustatymus, validacijas ir vizualizacijos rekomendacijas. Visi tipai turi būti konfigūruojami (pavadinimai, spalvos, ikonėlės).

Klausimų tipai (aptarti)

- Taip/Ne (arba sutinku/nesutinku, pritariu/nepritariu) — su galimybe keisti etiketes.
- Šviesoforas (3 būsenos: žalia, geltona, raudona) — su prasmės pervadinimu (pvz., supratau/iš dalies/nesupratau).
- Emocijos (viena arba kelios) — veidukai/ikonos, grupavimas į teigiamas/neutralias/neigiamas.
- Skalė / termometras (pvz., 1-10 arba „blogai“–„puikiai“) — horizontali/vertikali, su žymomis.
- Daugybiniis pasirinkimas (multi-select) — keli variantai vienu metu.
- Vieno pasirinkimo sąrašas (single-select) — vienas variantas iš sąrašo.
- Atviras tekstas (trumpas / ilgas) — sakinio ar pastraipos atsakymas.
- Neužbaigtas sakiny (fill-in) — vartotojas užpildo dalį sakinio.
- Procentinis paskirstymas („pyragas“) — vartotojas paskirsto 100% tarp kelių dalių.

Minimalūs reikalavimai kiekvienam klausimui

- Aiški paskirtis: ką matuosime (nuotaika, aiškumas, įsitraukimas, pasiūlymai, pasitenkinimas).
- Validacijos: privalomas pasirinkimas (kai taikoma), min/max ribos, teksto min simboliai (kai taikoma).
- Metaduomenys analitikai: tema (pvz., „pamoka“, „susirinkimas“), kontekstas, auditorija (grupė).
- Vizualizacijos tipas: numatytas grafikas ir alternatyva.

4. Universalūs pasirinkimai: „Nenoriu atsakyti“ ir „Nežinau“

Kiekvienas klausimas privalo turėti du papildomus pasirinkimus, nepriklausomai nuo klausimo tipo:

4.1 „Nenoriu atsakyti“

- Tai pilnavertis atsakymo statusas.
- Sistema neforsuoja papildomų žingsnių.
- Statistikoje rodoma atskirai (komforto/įsitraukimo indikatorius).

4.2 „Nežinau“ (nukreipiamųjų klausimų srautas)

- Paspaudus „Nežinau“, sistema paleidžia trumpą nukreipiamųjų (pagalbinių) klausimų seką (pvz., 1–3 žingsniai).
- Tikslas: padėti žmogui apsibrėžti atsakymą (pateikti pavyzdžius, suskaidyti į paprastesnius pasirinkimus).
- Po nukreipiamųjų klausimų vartotojas privalo grįžti ir pateikti galutinį atsakymą į pradinį klausimą.
- Išimtis: vartotojas visada gali sąmoningai pasirinkti „Nenoriu atsakyti“.

Rekomenduojama UX taisyklė

Nukreipiamųjų klausimų srautas turi turėti aiškų progresą (pvz., 1/3), o užbaigus — vieną aiškų veiksmą: „Grįžti ir atsakyti“.

5. Refleksijų planavimas, galiojimas, priminimai ir grupės prisijungimas

Pagrindinė problema praktikoje — surinkti refleksijas laiku. Todėl refleksijos turi būti planuojamos kaip „įvykiai“ su galiojimo langu. Jei refleksija užpildoma per vėlai, jos atsakymai gali būti nebevalidūs.

Planavimas

- Nustatyti dienas ir laiką (pvz., po kiekvienos pamokos).
- Nustatyti galiojimo langą (pvz., 20 min nuo pamokos pabaigos).
- Automatinis pasibaigimas: pasibaigus langui refleksija dingsta iš aktyvių.
- Greitas perplanavimas: jei pamoka/susitikimas atšauktas ar perkeltas, pakeitimas atliekamas per valdymo skydą.

Priminimai ir pranešimai

- Push / el. paštas / (vėliau) SMS — pagal konfigūraciją.
- Paskutinis priminimas prieš pabaigą (pvz., liko 10 min).
- Po pabaigos: pažymėti kaip „neužpildyta“, be galimybės vėluoti (arba su aiškia „vėluota“ žyme, jei leidžiama).
- Neperkrauti: ribotas priminimų skaičius, kad neerzintų.

Grupės prisijungimas

- Prisijungimas per kodą arba QR kodą.
- Greitas įtraukimas į grupę su minimaliais žingsniais.
- Po prisijungimo — automatinis aktyvių refleksijų rodymas ir pranešimų prenumerata.

6. Duomenų apdorojimas ir AI (tekstų sužymėjimas į struktūrą)

Atviri atsakymai be apdorojimo netinka statistikai. Sprendimas: kiekvieną tekstą (ar jo dalį) paversti struktūruotomis žymėmis. Tam galima naudoti ChatGPT API ar kitą NLP modelį. Svarbiausia — modelio išvestis turi būti griežtas JSON, kad būtų patikimai skaičiuojama.

Siūlomas AI rezultato rinkinys (minimalus)

- Tema (topic): apie ką kalbama (pvz., „tempo problema“, „grupinis darbas“, „paaiškinimo aiškumas“).
- Emocija (sentiment/emotion) + intensyvumas (1–5): teigiama / neutrali / neigiama.
- Pasiūlymo tipas (suggestion_type): metodika / komunikacija / aplinka / vertinimas / asmeninis / kita.
- Santrauka (summary): 1 sakiny, be interpretacijų.
- Veiksmo pasiūlymas (action_hint): ką galima daryti kitą kartą (jei yra).
- Raktiniai žodžiai (keywords): 3–8 žodžiai paieškai ir grupavimui.

Pavyzdinis JSON (AI išvestis)

```
{
  "topic": "pamokos tempo problema",
  "emotion": {"label": "neigiama", "intensity": 4},
  "suggestion_type": "metodika",
  "summary": "Pamoka vyko per greitai, todėl dalies temų nespėjau suprasti.",
  "action_hint": "Daryti trumpas pauzes po kiekvieno bloko ir patikrinti supratimą.",
  "keywords": ["tempas", "aiškumas", "paaiškinimas"]
}
```

Svarbios pastabos

- AI turi gauti ir kontekstą: klausimo tekstą, tipą, temą, auditoriją (mokykla/susirinkimas).
- Reikia užtikrinti stabilumą: tas pats tekstas turi duoti panašias žymes (versijavimas, testai).
- Privatumas: jautrius duomenis apdoroti atsargiai; mokyklos režime numatyti anonimizavimą ir mažiausio būtino duomenų perdavimo principą.

7. Statistika: individas, grupė, laikas, palyginimai

Sistema turi palaikyti 4 pagrindinius statistikos pjūvius:

- Individas vienu momentu: konkretaus užpildymo rezultatas.
- Individas laike: kaip keičiasi atsakymai per dienas/savaite/mėnesius.
- Grupė vienu momentu: klausimyno (ar klausimo) atsakymų pasiskirstymas grupei.
- Grupė laike: kaip kinta grupės rodikliai ir temos per laiką.

Papildomi pjūviai

- Pagal filtrus: klasė, grupė, tema, pamoka, mokytojas, laikotarpis.
- Pagal statusus: „neužpildė“, „nenoriu atsakyti“, „vėluota“.
- Top temos / dažniausi pasiūlymai iš atvirų atsakymų (AI sugrupavimas).

8. Vizualizacijos: rekomenduojami grafikai pagal klausimo tipą

Klausimo tipas	Momentinė grupės statistika	Laiko dinamika (individas / grupė)
Taip/Ne (2 reikšmės)	Stulpelinė (2 stulpeliai) arba donut chart	Linija (taip% per laiką) arba slankus vidurkis
Šviesoforas (3)	Stulpelinė (3 stulpeliai) / 100% stacked bar	Linija kiekvienai spalvai arba stacked area
Skalė (1-10)	Histograma / boxplot (jei daug duomenų)	Linija (vidurkis) + dispersija (juosta)
Emocijos	Dažniausių emocijų juostinė / „bubble chart“	Linija pagal emocijų grupes (poz/neu/neg)
Multi-select	Juostinė (kiek kartų pasirinkta kiekviena parinkčia)	Linija kiekvienai parinkčiai (top N)
Atviras tekstas	Temų „word cloud“ (atsargiai) + TOP temos	Linija TOP temai laike + sentimentas laike
Neužbaigtas sakiny	Temų grupavimas + TOP formulės	Linija TOP temų trendai laike
Procentinis paskirstymas (pyragas)	Pyragas/stacked 100% pagal dalis	Linija kiekvienai daliai (vidurkis %)

Pastaba: kai duomenų mažai, rodyti paprastesnius grafikus ir aiškiai žymėti imties dydį (N). Kai N mažas (pvz., <5), detalios statistikos nerodyti (privatumo ir patikimumo sumetimais).

9. Koreliacijos ir ryšių analizė tarp klausimų

Sistema turi leisti pasirinkti 2 klausimus ir analizuoti jų ryšį (pvz., emocijos vs pasitenkinimas). Tai gali būti daroma tiek individui (jei jis pildo reguliariai), tiek grupei (agreguoti duomenys).

Kaip rodyti koreliaciją

- Skaitinė metrika: koreliacijos koeficientas (pvz., Pearson/Spearman) + imties dydis (N).
- Grafikas: scatter plot (jei abu skaitiniai) arba grouped bar/boxplot (jei kategorinis vs skaitinis).
- Filtrai: laikotarpis, grupė, kontekstas, tik aktyvūs atsakymai (be „nenoriu atsakyti“).

Apsaugos ir interpretacija

- Koreliacija nėra priežastis — pateikti aiškų paaiškinimą.
- Rodyti tik nuo tam tikro N (pvz., >30), kad išvados nebūtų klaidingos.
- Anonimiškumo režimuose koreliacijas rodyti tik agreguotai.

10. Eksportas: CSV/Excel, Google Sheets, PDF ataskaitos

Svarbu, kad naudotojas galėtų ne tik matyti statistiką programoje, bet ir išsinešti duomenis.

- CSV/Excel: žali duomenys (atsakymai, laikas, grupė, statusai) ir agregatai (suminės lentelės).
- Google Sheets: tiesioginis eksportas arba sugeneruota nuoroda/importo failas.
- PDF ataskaitos: automatinė ataskaita su pasirinktu laikotarpiu ir pagrindiniais grafikai.
- Spausdinimas: optimizuotas PDF (A4), aiškos antraštės, N rodymas, privatumo žymos.

Rekomenduojami PDF šablonai

- 1 puslapio santrauka: pagrindiniai rodikliai, TOP temos, 2–3 įžvalgos.
- Pilna ataskaita: visi klausimai su grafikai + komentarai + veiksmo planas.
- Individuali ataskaita (jei leidžiama): asmens tendencijos laike.

11. Valdymo skydelis: kalendorius, aktyvumas, kontrolė ir redagavimas

Valdymo skydelis turi veikti kaip kalendorius, kuriame refleksijos/klausimynai yra „įvykiai“. Svarbiausia: greitas vaizdas ir greitas valdymas.

Vaizdai

- Savaitės vaizdas (default), mėnesio vaizdas, dienos vaizdas.
- Aktyvių refleksijų sąrašas su galiojimo laiku ir užpildymo progresu (%).

Kontrolė ir veiksmai

- Matyti: kas užpildė / kas neužpildė, procentas, likęs laikas.
- Gauti įspėjimus: „šie N žmonių dar neatsakė“ (pvz., liko 10 min).
- Redaguoti/perkelti/ištrinti refleksiją iš karto (be gilių meniu).
- Atbulinis valdymas: galima grįžti į senas refleksijas, koreguoti, trinti su patvirtinimu.

Saugūs patvirtinimai (kai yra duomenų)

- Jei refleksija jau turi atsakymų — prieš trynimą klausti: „ar tikrai norite prarasti duomenis?“
- Keičiant klausimus po atsakymų — įspėti apie statistikos suderinamumą (versijos).

12. Saugumas, privatumas, atitiktis (GDPR)

Kad įrankis būtų realiai naudojamas mokyklose/organizacijose, saugumas ir privatumas turi būti įdiegtas nuo pradžių.

Privatumo režimai (rekomenduojama)

- Vardinis: mokytojas mato visus atsakymus.
- Pseudo-anoniminis: mokytojas mato agregatus, o individualūs tekstai ribojami.
- Anoniminis agregatas: rodyti tik grupės suvestines, taikyti „min N“ taisyklę (pvz., $N \geq 5$).

Saugumo bazė

- Autentifikacija (mokyklos prisijungimai/SSO) + rolėmis grįsta autorizacija (RBAC).
- Audit log: kas kūrė, keitė, peržiūrėjo, eksportavo.
- Duomenų šifravimas: HTTPS visur; jautrių laukų šifravimas DB (kai reikia).
- Apsauga nuo piktnaudžiavimo: rate limiting, bruteforce apsauga, CSRF/Origin apsauga (jei cookie auth).
- Duomenų minimalizavimas: rinkti tik tai, kas būtina; aiški duomenų saugojimo politika.

AI privatumo pastabos

- AI užklausoms perduoti tik reikalingą kontekstą (klausimo tekstas + atsakymas).
- Jei reikia — anonimizuoti vardus/identifikatorius prieš siunčiant į AI.
- Aiškiai dokumentuoti, kur keliauja duomenys ir kaip jie naudojami.

13. MVP apimtis, techninė architektūra, rizikos

Kad projektas judėtų greitai, siūlomas MVP (minimalus veikiantis produktas) su tvirta baze ir plėtros keliais.

MVP (siūlomas)

- Klausimų tipai: Taip/Ne, Šviesoforas, Skalė, Emocijos, Atviras tekstas.
- Universalūs pasirinkimai: „Nenoriu atsakyti“ ir „Nežinau“ su nukreipiamais klausimais.
- Planavimas + galiojimo langas + priminimai.
- Valdymo skydelis: savaitės vaizdas + aktyvumo % + kas neatsakė.
- AI apdorojimas atviram tekstui į JSON žymes.
- Statistika: baziniai grafikai + eksportas CSV/PDF.

Techninė architektūra (aukšto lygio)

- Frontend: web aplikacija (responsyvi), fokusas į greitą pildymą telefone.
- Backend: REST API (vėliau galima pridėti real-time), rolės ir teisių kontrolė.
- DB: struktūruoti atsakymai + atskirai AI žymės (versijuojamos).
- Notifikacijos: suplanuotos užduotys (scheduler/queue) priminimams ir galiojimo pabaigai.
- Ataskaitos: server-side PDF generatorius + CSV eksportas.

Rizikos ir priklausomybės

- Duomenų kokybė: jei klausimai neaiškūs, statistika bus triukšminga — reikia šablonų ir validacijų.
- AI kaina ir stabilumas: būtina riboti užklausas, daryti batch, turėti fallback (be AI).
- Privatumas mokyklose: reikia aiškių režimų ir minimalaus duomenų atskleidimo.
- Suderinamumas keičiant klausimus: versijos, migracijos, aiškos taisyklės.

Šis dokumentas apibendrina reikalavimus ir idėjas iš pokalbio. Kitas praktinis žingsnis: sukurti duomenų modelį (DB schemas) ir MVP ekranų maketus (UI) pagal šią specifikaciją.