

Reflectus: Apple design inspiration (pokalbio išrašas)

Dokumentas sugeneruotas: 2026-01-04

Turinys: šio pokalbio (User <-> Assistant) išrašas, skirtas dizaino įkvėpimui ir techniniams pritaikymui Reflectus programėlei.

Pokalbio išrašas

User

dabar man papsakok ir apibūdink apple design elementus ir taktika. mana svarbu kad apaibudint apple healt design. mental health skilty . noriu pritaikyti apple design reflectus programele nes ten labai gera design. nenukopiųoti nes taio nelegalu bet GET INSPIRED

Assistant

1) „Apple design DNA“ (kā kopijuoti kaip principą, o ne kaip vaizdą)

Apple savo platformose remiasi 3 kertiniais UI „temomis“:

- Deference (turinys pirmoje vietoje): UI neturi konkuruoti su turiniu – jis turi padėti suprasti atlikti veiksmą.
- Clarity (aiškumas): tekstas įskaitomas visur, ikonografija tiksliai, nereikalingų dekoracijų mažai, funkcija akivaizdi.
- Depth (gylis): sluoksniuose + realistiškas judesys padeda suvokti hierarchiją ir kur esi.

Praktikoje tai reiškia:

- Mažai triukšmo: daug „oro“, mažai rėmelių, 1–2 akcentai ekrane.
- Progresyvus atskleidimas: pradžioje tik svarbiausia, detalės atsiranda tik tada, kai jų reikiama.
- Vienas sprendimas per žingsnį: vietoj „ilgos formos“ – trumpas „flow“.

2) Kuo išsiskiria Apple Health „Mental Wellbeing“ dizainas (taktika)

A. „Quick logging“: minimalus įvedimas, maksimalus signalas

Health logika: greitai užregistruoja būseną, tada (jei nori) pridedi detales ir matai istoriją grafikuose.

Reflectus vertimas: pamokos refleksija neturi prasidėti nuo teksto. Pirma – 1–2 paspaudimai (būsena/įvertinimas), tada tik papildymai.

B. Struktūra: „Moment“ vs „Day“ + „Contributing factors“

Gali loginti momentines emocijas, dienos nuotaikas ir prisidedančius veiksnius, o tada gauti ižvalgas per grafikus.

Reflectus vertimas: atskirk:

- „Pamoka dabar“ (momentas)
- „Savaitė“ (tendencija)
- „Kas labiausiai įtakojo?“ (faktoriai)

C. Interaktyvūs grafikai ir „insights“

Istorija ir pokytis laike motyvuojant pildyti, nes matai prasmę.

Reflectus vertimas: rodyk mini tendenciją mokiniui + mokytojui agregatą.

D. Standartizuotos anketos, bet aiškiai atskirtos nuo „log“ „Gilesnes“ apklausas laikyti atskirai nuo kasdienės refleksijos.

E. Privatumas kaip UX dalis

Mokinys pildys tik jei supras kas mato atsakymus, turės kontrolę (pvz., „nenoriu atsakyti“) i jausis saugiai.

3) Kaip tai pritaikyti Reflectus (konkretus receptas)

A. Ekranų taktika (idealus 30–60 s „flow“)

- 1) Check-in (2 paspaudimai): „Kaip sekėsi pamoka?“
- 2) Priežastys (chip/tag'ai): „Kas labiausiai įtakojo?“ (pasirink 1–3)
- 3) Vienas sakinys (optional): „Jei nori – parašyk vieną mintį“
- 4) Užbaigimas: „Išsaugota“ + mini grafikas / mini ižvalga

B. Vietoj Apple „mood“ – tavo unikali ašis (kad nebūtų kopijos)

Pvz. ašys:

- Aišumas: „Supratau / Nesupratau“
- Energija: „Energijos daug / mažai“
- Saugumas klausti: „Drąsu / nedrąsu“

C. Privalomi „Nenoriu atsakyti“ ir „Nežinau“

- „Nenoriu atsakyti“ → priimk ir judék toliau.
- „Nežinau“ → 1–2 nukrypimo klausimai, tada grįžti prie pagrindinio.

D. Mokytojos vaizdas (Health „insights“ analogas)

- Klasės būsenos aggregatas,
- 3 top faktoriai,
- anomalijos (pvz., daug „nežinau“ ties tema).

4) Dizaino detalės, kurios sukuria „Apple Health“ jausmą (bet lieka tavo)

- Tipografiros hierarchija
- Kortelės su turiniu
- Chip/tag'ai faktoriams
- Švelni animacija tik ten, kur padeda
- Neutralus tonas („User control“ copy)

5) Saugumas ir legalumas

- Nekopijuok Apple ikony/iliustracijų/komponentų 1:1.
- Naudok legalius fontus (pvz., system-ui / Inter).
- Venk medicininių „diagnozių“ – laikyti tai mokymosi savijauta.

User

daugiau. suras kdaugiau apple design detaliu. daugiau inspiration. man reikaikad tu sugalvotu
daugiau visko. kaip PRITAIKYTI MUSU REFLECTUS APP. NES APPLE DESING IS VERY GOOD. WE NEED TO
INSPIRED BY THE, AN IMPLEET TO OUR REFLECTION. nes ju dizainas gera. ir jis nukeiptas i user
frienly ir user centered

Assistant

[Pastaba: pokalbio metu buvo parodyta 4 paveikslėlių karuselė. PDF'e paveikslėliai neįtrauki

1) Apple Health (Mental Wellbeing) „formulė“: kodėl tai veikia

- Greitas įrašas (low friction) → tada detalės (high value)

- „Moment“ vs „Period“
- Įžvalgos kortelėmis (Highlights): viena aiški išvada + kontekstas
- Privatumas kaip UX dalis

2) Apple dizaino detalės, kurias gali perkelti į Reflectus (ne kopijuojant vaizdą)

- Vizualinė hierarchija: dideli sprendimai, mažai triukšmo
- Progressive disclosure (progresyvus atskleidimas)
- Microcopy: paprasta, neutrali kalba
- Chips vietoj formų
- „Show in Charts“: grafikai dėl prasmės, ne dėl grožio

3) Kaip tiksliai pritaikyti Reflectus: konkretus UX blueprint

3.1 Studentas: „Pamokos refleksija“ (30–60 sekundžių)

Ekranas 1 – Quick check:

- Pagrindinis įvertinimas (skalė)
- „Nenoriu atsakyti“
- „Nežinau“

Jei „Nežinau“ → Ekranas 1A (pagalbinis, 2 klausimai max):

- Per greitai / Per lėtai / Gerai
- Kiek supratai? (Mažai / Vidutiniškai / Daug)

Ekranas 2 – Factors (chips):

- „Kas labiausiai įtakojo?“ (1–3)

Ekranas 3 – One sentence (optional):

- šablonai: „Šiandien supratau...“, „Sunkiausia buvo...“, „Padėjo...“

Ekranas 4 – Saved + mini insight:

- mini įžvalga iš paskutinių pamokų

3.2 Studentas: „Savaitės refleksija“ (2–4 minutės)

- bendras vaizdas + 3 klausimai (geriausia pamoka, kur strigai, ką keisti)

3.3 Kontrolinio / projekto refleksija

Tas pats UI, skirtinti klausimų šablonai.

4) Mokytojos pusė: Highlights analogas

Dashboard:

- 1) Šiandienos signalai
- 2) Klasės tendencija (7 d.)
- 3) Veiksmo pasiūlymai

5) Dizaino sistema

Komponentai: Page header, Card, Segmented control, Chips, Buttons, Bottom sheet, Toast.

6) Kaip „nekopijuoti“, bet gauti efektą

Kopijuok principus: flow logiką, hierarchiją, toną, kortelių/žymų principą, insights struktūrą.

7) Papildoma informacija, reikalinga taikliam dizainui

- 1) Kiek ekranų flow'ui? (2/3/4)
- 2) Kokie 3 pagrindiniai matavimai?
- 3) Ar reikalingas tekstas, ar pakanka pasirinkimų?

User

daugiau info. daugiau detaliu dabar labiau is technines puses.

Assistant

1) Principai → techniniai reikalavimai

Tikslas: user-centered, mažas friction, aiškus flow, prieinamumas, privatumas.

2) Design tokens (vienas tiesos šaltinis)

- spacing: 4,8,12,16,20,24,32,40,48
- radius: 12,16,20
- typeScale: 12,14,16,20,24,32
- shadow: 2–3 lygai
- motion: 120ms, 180ms, 240ms
- colors: neutralūs + 1 accent; light/dark

3) Komponentų sistema

Rekomendacija (React):

- shadcn/ui (Radix) + tavo tokens

Būtini: Button, Card, Sheet/Modal, Segmented control, Chips, List row, Toast.

Praktika:

- tap targets ~ 44x44
- aiškūs states (hover/focus/disabled)

4) Motion ir „depth“

- laikyk animacijas funkcines (modal/toast/chips)
- palaikyk prefers-reduced-motion

5) Tipografija

- system-ui arba Inter
- rem vienetai, max line length
- aiškūs line-height

6) Schema-driven klausimai (kad flow būtų tvarkingas)

Klausimų schema:

- questionId
- type: scale | singleChoice | multiChoice | text
- options
- special: refuse | dontKnow
- branching: dontKnow → 1–2 clarifier → grįžti

Flow engine:

- reducer arba XState (rekomenduojama jei daug branching)

7) Backend (insights mokytojai)

DB: reflection_entries:

- schoolId, studentId, teacherId, contextType, contextId, timestamp
- primaryScore, secondary, factors[], text?
- answeredState: normal | refuse | dontKnowResolved
- metadata

API:

- POST /reflections
- GET /student/reflections?range=7d
- GET /teacher/insights?classId=...&range=7d
- GET /teacher/insights/:contextId

Insights:

- avg/median/distribution
- top factors
- dontKnow rate
- outliers

8) Privatumas ir saugumas

- RBAC (student/teacher/admin)
- rate limits ant auth/reflection
- multi-tenant per schoolId
- TLS + DB encryption
- agregatai vietoje „vardas + viskas“ (mažina identifikavimo riziką)

9) QA kriterijai

- tap targets $\geq 44 \times 44$
- WCAG AA kontrastas
- reduced-motion
- keyboard/focus
- layout nelaužo tekstu

Tooling:

- Playwright E2E
- Lighthouse + axe-core
- TypeScript + lint

10) Implementavimo seka

- 1) Tokens
- 2) Core komponentai
- 3) Flow engine (schema)
- 4) Student quick flow
- 5) Teacher insights API + dashboard
- 6) Ally/motion/contrast audit
- 7) Observability (Sentry + event'ai)

— Pokalbio pabaiga —