Turinio autokūrėjo maketo reikalavimų analizė

2025-04-28



# Turinys

[Turinys 2](#_Toc198217488)

[Santrauka 3](#_Toc198217489)

[1. Įvadas 4](#_Toc198217490)

[2. Pagrindiniai naudotojų scenarijai (workflows) 6](#_Toc198217491)

[3. Funkciniai reikalavimai 8](#_Toc198217492)

[4. Ne-funkciniai reikalavimai 12](#_Toc198217493)

[5. Techniniai komponentai 13](#_Toc198217494)

[6. MVP apimtis 16](#_Toc198217495)

[7. Roadmap: tolesnės versijos 18](#_Toc198217496)

[8. Rizikos 20](#_Toc198217497)

# Santrauka

**Projekto tikslas** – sukurti dirbtiniu intelektu grįstą autokūrėjo funkciją, kuri leistų įvairiems vartotojams greitai, lengvai ir kokybiškai generuoti turinį interneto svetainėms, jį peržiūrėti, patvirtinti ir publikuoti. Funkcija yra viena iš kertinių *AdoreIno* platformos dalių.

**Pagrindinės įžvalgos iš rinkos tyrimo**

* Vartotojai neturi laiko, biudžeto ar kompetencijos kurti svetainės turinio.
* Turinys dažnai paviršutiniškas, be struktūros ar lokalizacijos.
* Trūksta įrankių, kurie padėtų ne tik generuoti, bet ir sistemingai **publikuoti bei prižiūrėti** turinį.
* Vertę kuria ne „teksto masė“, o **pritaikymas, aiškumas ir publikavimo kontrolė**.

**Tiksliniai naudotojai**

* **Verslo savininkai** – nori greitai gauti tekstą be papildomos pagalbos.
* **Rinkodaros specialistai** – siekia kokybiško, SEO pritaikyto, struktūruoto turinio.
* **Agentūros** – kuria didelį kiekį turinio skirtingiems klientams.
* **IT palaikymo komanda** – įgyvendina pateiktą turinį svetainėje.

**Pagrindiniai funkcionalumo blokai**

1. Šablonų pasirinkimas pagal puslapio tipą (pvz., „Apie mus“, „Paslaugos“)
2. Įvesties forma (tema, stilius, raktiniai žodžiai)
3. AI sugeneruotas tekstas + keletas variantų
4. Vizualų integracija (vėlesniuose etapuose)
5. Preview-as-page peržiūra
6. Publikavimo režimai: demo → patvirtinimas → live
7. Eksportas į HTML / Notion / WordPress / Prestashop

**Techniniai komponentai**

* **Modulinė API-first architektūra**
* **GPT arba analogiškas LLM** turinio generavimui
* **Patvirtinimo ir kokybės tikrinimo sistema**
* **CMS (WordPress / Prestashop) integracija** per API ar eksportą
* **Preview peržiūra + publikavimo patikra** (automatinė ir/ar rankinė)

**MVP apimtis**

* Vieno puslapio generavimas (iš 5–7 šablonų)
* Paprasta įvestis, viena GPT integracija
* Rankinis tvirtinimas ir eksportas
* „Demo“ publikavimas prieš „live“
* Peržiūra kaip tinklapio blokas

**Roadmap (tolimesnės versijos)**

* Vaizdų generavimas
* CMS publikavimo API
* Komandinė sąveika (delegavimas, komentarai)
* Agentūrų režimas (keli klientai vienoje paskyroje)
* Versijų lyginimas ir turinio monitoringas

**Rizikos**

* Nestabilus LLM generavimas
* Vartotojų pasitikėjimo stoka be aiškių paaiškinimų
* Integracijos su CMS ir publikavimo kokybės testavimas
* UX sudėtingumas mažai kompetentingiems naudotojams

**Išvada:** Turinio autokūrėjo funkcija turi būti ne tik teksto generavimo įrankis, bet **pilna sprendimų kūrimo sistema**, leidžianti skirtingiems naudotojams *sukurti, peržiūrėti, patvirtinti ir publikuoti turinį be papildomų grandžių*. MVP versija turi atskleisti šios vertės pagrindą, o plėtra – paversti funkciją svarbiausiu AI turinio operacijų centru skirtingiems verslo segmentams.

# 1. Įvadas

**1.1 Reikalavimų analizės paskirtis**

Šios reikalavimų analizės užduoties tikslas – suformuluoti aiškius reikalavimus „AdoreIno“ produkto **turinio autokūrėjo** funkcijai (1.4 modulis). Ši funkcija yra esminė bendros platformos dalis ir skirta automatizuotam, dirbtiniu intelektu grįstam **svetainės turinio kūrimui**, orientuotam į realius verslo naudotojų poreikius: greitį, kokybę, struktūrą, SEO ir vizualinį vientisumą.

Funkcija turi būti suprojektuota taip, kad programuotojai, dizaineriai ir AI integracijų inžinieriai galėtų įgyvendinti ją etapais, vadovaudamiesi apibrėžtais vartotojo scenarijais, funkcinių blokų struktūra ir sąveika su kitomis platformos dalimis.

**1.2 Produkto kontekstas**

AdoreIno – tai AI pagrindu veikianti IT palaikymo ir turinio automatizavimo platforma, kuri skirta padėti smulkiam, vidutiniam ir agentūriniam verslui sistemingai kurti, analizuoti, atnaujinti ir valdyti interneto svetainės turinį. Autokūrėjo funkcija (1.4) yra viena iš pagrindinių produkto dalių, tiesiogiai reaguojanti į vartotojų rinkos tyrime išreikštus skausmo taškus:

* **Nėra kam kurti turinio**
* **Nėra laiko ar biudžeto**
* **Trūksta kompetencijos rašyti kokybiškai**
* **Svetainės tekstai yra neįtaigūs, lekšti, neaktualūs**
* **Trūksta SEO, stiliaus, vizualų ir tęstinumo**

Autokūrėjo funkcija turi leisti naudotojui sukurti pilnavertį, publikuojamą svetainės turinį iš minimalaus įvesties kiekio – pasitelkiant iš anksto apgalvotus šablonus, dirbtinį intelektą (GPT), lokalizacijos modelius ir vizualų valdymą.

**1.3 Tikslas**

1.4 funkcijos tikslas – suteikti naudotojui galimybę **automatizuotai sukurti arba atnaujinti svetainės turinį** be profesionalaus tekstų kūrėjo ar dizainerio įsikišimo, išlaikant kokybę, struktūrą, stilistinį vientisumą ir pritaikomumą skirtingoms svetainėms bei verslo sritims.

Funkcija turi:

* Palaikyti **turinio šablonus** pagal puslapių tipus (pvz., „Apie mus“, „Paslaugos“, „Kontaktai“, „Produktų aprašymai“)
* Leisti įvesti minimalų kiekį informacijos (pvz., raktiniai žodžiai, paslaugos pavadinimas, niša, auditorija)
* Sugeneruoti turinį su pritaikytu **stiliumi, tonu ir SEO elementais**
* Integruoti vizualinius pasiūlymus arba placeholder‘ius (vaizdo rekomendacijas)
* Užtikrinti, kad turinys būtų **lokalizuotas**, struktūruotas, **be gramatinių klaidų** ir tinkamas publikavimui

**1.4 Santykis su kitomis funkcijomis**

Autokūrėjo funkcija veikia glaudžiai susijusi su šiomis kitomis produkto funkcijomis:

* **Turinio analitikas (1.2)** – pateikia, kuriuos puslapius reikia atnaujinti, sukurti ar perrašyti. 1.4 funkcija tai įgyvendina.
* **Vizualų generatorius (1.5)** – siūlo vaizdus ar stiliaus gaires, atitinkančias sukurtą tekstą.
* **Publikavimo modulis (1.6)** – leidžia paruoštą turinį paskelbti svetainėje arba eksportuoti.

Ši specifikacija apibrėžia, ką **konkrečiai turi sugebėti ši funkcija**, kaip ji turėtų būti kuriama (MVP ir tolesnės versijos), ir kaip ją turės naudoti tikslinės grupės (naudotojai aprašyti rinkos tyrimo 2 skyriuje).

# 2. Pagrindiniai naudotojų scenarijai (workflows)

Turinio autokūrėjo funkcija yra orientuota į keturias pagrindines naudotojų grupes, kurios turi skirtingą lygį kompetencijos, atsakomybių bei turinio kūrimo apimties. Visi šie naudotojai turi vieną bendrą tikslą – **greitai, patogiai ir kokybiškai sukurti arba atnaujinti svetainės turinį**, tačiau jų darbo pobūdis ir sąveika su įrankiu skiriasi.

**2.1 Verslo savininkas (paprasta sistema)**

* **Profilis:** Mažo verslo savininkas, dažnai pats administruoja svetainę, neturi vidinės rinkodaros komandos.
* **Poreikiai:** Nori sukurti ar atnaujinti pagrindinius svetainės puslapius („Apie mus“, „Paslaugos“, „Kontaktai“) be rašytojo ar dizainerio pagalbos.
* **Elgsena:** Tikisi minimalaus įvedimo (raktiniai žodžiai, paslaugos pavadinimas) ir greito rezultato. Vertina aiškumą, paprastumą, „copy–paste“ paruoštą turinį.
* **Vertės centras:** Greitas ir praktiškas rezultatas. Nereikia gilintis į procesus.

**2.2 Rinkodaros specialistas (vidinis darbuotojas)**

* **Profilis:** Įmonės viduje dirbantis žmogus, atsakingas už komunikaciją, turinio kūrimą, SEO.
* **Poreikiai:** Nori turėti priemonę, kuri padėtų sukurti struktūruotą ir strategiškai pritaikytą turinį, derantį su rinkodaros tikslais.
* **Elgsena:** Dirba pagal planą – turi kurti kelis puslapius per savaitę, ruošia kampanijas, atnaujina blog'ą. Nori naudotis AI kaip „asistentu“, o ne pilnai automatizuotu juodraščių tiekėju.
* **Vertės centras:** Kontroliuojama kokybė, aiški struktūra, SEO integracija.

**2.3 Agentūra (išorinis tiekėjas)**

* **Profilis:** Agentūra ar freelancerių komanda, kurianti svetainių turinį savo klientams.
* **Poreikiai:** Greitai sukurti profesionalų svetainės turinį naujam klientui, dažnai neturint visos informacijos ar konteksto.
* **Elgsena:** Naudoja autokūrėją kaip pradinį kūrybinį impulsą, taupant laiką, dirbant su 5+ klientų projektais vienu metu.
* **Vertės centras:** Greitis, aiškus šablonas, tonas pagal kliento nišą, galimybė greitai gauti kelių variantų tekstą.

**2.4 IT / svetainės palaikymo darbuotojas**

* **Profilis:** Techninis darbuotojas, kuris nėra atsakingas už turinio kūrimą, bet turi įkelti / atnaujinti tekstus svetainėje.
* **Poreikiai:** Gauti struktūrizuotą, redaguojamą tekstą, kurį galima greitai įkelti į CMS be papildomų paaiškinimų.
* **Elgsena:** Nekuria turinio, bet atlieka veiksmus pagal paruoštą tekstą. Retkarčiais gali naudoti įrankį, jei nėra kam sukurti turinio.
* **Vertės centras:** Aiškus rezultatas, techninis formatas, be „kūrybinio triukšmo“.

**Naudotojų apibendrinimas:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Naudotojas** | **Kompetencija** | **Poreikis** | **Elgsena** | **Vertės centras** |
| Verslo savininkas | Žema | Greitai sukurti svetainės tekstą | Įveda raktinius žodžius, kopijuoja tekstą | Laikas ir paprastumas |
| Rinkodaros specialistas | Vidutinė | Struktūruotas, tikslingas turinys | Formuoja strateginius puslapius, koreguoja | Kontrolė, kokybė, SEO |
| Agentūra | Aukšta | Sukurti turinį daugeliui klientų | Dirba greitai, nori šablonų ir stiliaus | Efektyvumas ir profesionalumas |
| IT / palaikymas | Techninė | Gauti paruoštą tekstą įkėlimui | Įkelia, formatuoja, kartais tikrina turinį | Struktūra ir techninis tikslumas |

**Išvada:**

Turinio autokūrėjas turi būti universalus, bet adaptuojamas: vieniems – maksimaliai paprastas, kitiems – redaguojamas ir profesionaliai valdomas. Sistemos projektavimas turi leisti:

* **Personalizuoti**: pritaikyti toną, ilgį, tikslą.
* **Valdyti**: matyti keletą variantų, atlikti korekcijas.
* **Taupyti laiką**: pateikti turinį struktūroje, pritaikytoje publikavimui.
* **Integruoti**: į kitas darbo priemones ar sistemą (Notion, CMS ir pan.)

# 3. Funkciniai reikalavimai

Turinio autokūrėjo funkcionalumas susideda iš dviejų pagrindinių grupių:

1. **Reikalavimai pagal turinio kūrimo procesą** – šie funkcionalumai būtini tam, kad naudotojas galėtų nuosekliai, intuityviai ir efektyviai sukurti svetainės turinį nuo idėjos iki paruošto rezultato.
2. **Reikalavimai pagal turinio tipus** – specifiniai funkcionalumai, reikalingi skirtingo pobūdžio turiniui: tekstui, vizualams, produktų aprašymams, video.

**3.1 Funkciniai reikalavimai pagal turinio kūrimo procesą**

Autokūrėjas turi remtis nuosekliu vartotojo veiksmų ciklu, kuris leidžia bet kuriam naudotojui – nepriklausomai nuo kompetencijos – sukurti svetainės turinį nuo nulio. Žemiau pateikiami pagrindiniai reikalavimai kiekvienam proceso etapui:

**1. Turinio kūrimo pradžia / poreikio nustatymas**

* Galimybė pasirinkti puslapio tipą iš šablonų sąrašo (pvz., „Apie mus“, „Paslaugos“, „Kontaktai“, „Produktas“).
* Galimybė importuoti poreikį iš analitiko (pvz., „Reikia atnaujinti puslapį X“).
* Galimybė nurodyti tikslą: informuoti, parduoti, supažindinti, padidinti konversiją.

**2. Pradinės informacijos įvedimas**

* Forma, kurioje naudotojas pateikia pagrindinę informaciją:
  + raktažodžius,
  + temą / paslaugą / produkto pavadinimą,
  + tikslinę auditoriją,
  + stilių / toną (pvz., formalus, draugiškas, pardaviminis),
  + turinio apimtį (trumpas, vidutinis, ilgas).

**3. Turinys generavimas**

* AI generuoja turinį pagal šabloną ir naudotojo įvestį.
* Galimybė peržiūrėti bent 2–3 turinio versijas.
* Kiekvienai versijai rodomas „paaiškinimas“: kodėl toks tonas, kokia struktūra, kur integruoti raktažodžiai.
* Kiekvienoje generacijoje galima:
  + redaguoti tekstą rankiniu būdu,
  + pakeisti toną / ilgį ir regeneruoti,
  + išsaugoti ar atmesti.

**4. Vizualo / komponentų integracija**

* Galimybė pridėti vizualinius komponentus (pagal žemiau aprašytus tipo reikalavimus).
* Galimybė naudoti AI pasiūlytus paveikslėlių raktinius žodžius.
* Galimybė integruoti vietą iliustracijoms / ikonoms / video.

**5. Peržiūra ir struktūrinis patikrinimas**

* Turinys rodomas su vizualine struktūra (pvz., antraštės, paryškinimai, sąrašai).
* Galimybė tikrinti:
  + ar yra CTA (kvietimas veikti),
  + ar turinys atitinka struktūros reikalavimus (pvz., H1–H3, paragrafai),
  + ar tekstas atitinka lokalizacijos nustatymus.

**6. Eksportas / publikavimo paruošimas**

* Galimybė eksportuoti:
  + tekstą į HTML, Markdown, Notion, WordPress, Prestashop ar el. laiško formatą,
  + vizualų sąrašą (pvz., kartu su tekstu ZIP arba su nuorodomis).
* Galimybė išsaugoti juodraštį.
* Galimybė žymėti statusą: „pasiruošęs“, „reikia korekcijos“, „publikuota“.

**7. Publikavimas ir kokybės patikrinimas**

* **Publikavimo režimai:**
  + **Automatinis publikavimas** – turinys iškart keliamas į svetainę be papildomo tvirtinimo.
  + **Rankinis publikavimas** – turinys keliamas tik po naudotojo patvirtinimo.
  + **Publikavimas į demonstracinę versiją** – turinys pirmiausia įkeltas į *demo/live preview* versiją, kurioje galima peržiūrėti rezultatą.
* **Patvirtinimo sistema:**
  + Prieš galutinį publikavimą naudotojui pateikiamas išankstinis peržiūros vaizdas su galimybe:
    - redaguoti,
    - atmesti,
    - patvirtinti.
* **Patikrinimo logika:**
  + Sistema privalo atlikti automatinį turinio tikrinimą: ar puslapis įsikėlė sėkmingai, ar nėra kritinių klaidų (pvz., tuščias blokas, nutrūkęs HTML, klaidinga kalbos žyma).
  + Galimybė pasirinkti **verifikavimo režimą**:
    - **Automatinis testavimas** – sistema patvirtina, kad turinys rodomas teisingai.
    - **Rankinis peržiūrėjimas** – naudotojas gauna pranešimą ir turi patvirtinti.
    - **Miksuotas režimas** – sistema pirmiausia patikrina, o jei aptinkama neapibrėžtų rizikų – siunčia žmogui.
* **Statuso keitimas ir perėjimas į LIVE:**
  + Po patvirtinimo, turinys automatiškai keliamas į LIVE versiją (pvz., WordPress, Prestashop ar kitą CMS).
  + Sistemoje fiksuojama publikavimo data, atsakingas asmuo ir galimybė grįžti į juodraštį, jei reikia.

**3.2 Funkciniai reikalavimai pagal turinio tipus**

Kiekvienas turinio tipas turi savo specifinius reikalavimus, atsižvelgiant į rinkos tyrime aprašytas problemas ir vartotojų lūkesčius.

**3.2.1 Tekstas**

* Generavimo šablonai pagal puslapio tipą: „Apie mus“, „Paslaugos“, „Produktai“, „Kontaktai“, „Tinklaraštis“.
* Tonų ir stilių pasirinkimas (pvz., verslo, laisvas, SEO-friendly).
* Struktūros patikrinimas: ar yra pavadinimas, įvadas, CTA, išvados.
* Lokalizacijos palaikymas: automatinė adaptacija EN, LT ir kt. kalbomis.
* Gramatinė ir stilistinė analizė.
* SEO patarimai: kur naudoti raktinius žodžius, meta aprašymo sugeneravimas.

**3.2.2 Paveikslėliai**

* Galimybė generuoti AI iliustracijas pagal tekstą ar raktinius žodžius.
* Galimybė naudoti iš anksto įkeltą vaizdų biblioteką.
* Formato parinktys pagal platformą (web, social, banner).
* Rekomendacijos paveikslėlių stiliui (ikonos, fonai, žmonės, produktai).
* Autorių teisių tikrinimas (jei integruojama su išoriniais šaltiniais).

**3.2.3 El. parduotuvės prekių turinys**

* Galimybė masiškai generuoti prekių aprašymus pagal šablonus ir produkto atributus (pvz., pavadinimas, savybės, kaina, gamintojas).
* Paruošti šablonai: techninis aprašymas, privalumų sąrašas, emocinis tekstas.
* Vienodų produktų variantų sujungimas arba diferenciacija.
* SEO integracija: ilgauodegiai raktažodžiai, meta duomenys.
* Eksportas į CSV ar JSON formatą (sinchronizacijai su PIM / e. komercijos platformomis).

**3.2.4 Video (tekstinis paruošimas)**

* Galimybė sugeneruoti:
  + video scenarijų (pvz., 30 sek., 1 min., 3 min.),
  + video titrus / tekstus / voiceover scenarijų.
* Skirtingi scenarijai: paslaugų pristatymas, produkto demo, „how-to“ instrukcija.
* Struktūriniai komponentai: intro, problema, sprendimas, CTA.
* Galimybė eksportuoti tekstą į video kūrimo įrankius (pvz., Pictory, Synthesia).
* Video žanro pasirinkimas (prezentacija, naratyvas, instrukcija).

**Išvada**

Funkciniai reikalavimai turi būti įgyvendinami etapais, pradedant nuo pagrindinio srauto: **turinio kūrimo tekstui**, vėliau papildant **vaizdų integracija**, **produktų masiniu aprašymu** ir **video scenarijų generatoriumi**. Visi funkcionalumai turi veikti modulariai, būti iškeliami kaip atskiri komponentai su galimybe plėstis.

# 4. Ne-funkciniai reikalavimai

Netechnologiniai reikalavimai yra susiję su naudotojo patirtimi, turinio kokybės lūkesčiais, procesų aiškumu ir produkto naudojimo kultūra. Šie reikalavimai būtini siekiant, kad autokūrėjo funkcija būtų **ne tik funkcionali, bet ir naudotina**, **priimtina** bei **pasitikima** tarp skirtingų vartotojų grupių.

**4.1 Turinys turi būti patikimas, profesionalus ir reprezentatyvus**

* **Stilistinis vientisumas:** Sugeneruotas turinys turi atitikti kalbos normų reikalavimus (gramatika, stilius, loginė struktūra), kad jis galėtų būti naudojamas be papildomos korektūros.
* **Tonas ir pozicionavimas:** Teksto tonas turi būti aiškiai apibrėžiamas pagal naudotojo pasirinktą stilių – pvz., formaliai profesionalus, šiltai patikimas ar laisvas ir pardaviminis.
* **Aiški komunikacijos struktūra:** Turinys turi būti pateiktas pagal įprastus internetinės komunikacijos reikalavimus – su antraštėmis, pastraipomis, CTA, skaitomumu mobiliuose įrenginiuose.

**4.2 Turinys turi būti publikuotinas „toks, koks yra“ arba beveik be korekcijų**

* **„Ready to use“ filosofija:** Tikslas – turinys, kurį naudotojas galėtų iš karto kopijuoti į svetainę be korektūros, dizainerio ar tekstų kūrėjo įsikišimo.
* **Mažai priklausomas nuo konteksto:** Sistemoje sukurtas turinys turi būti pakankamai bendras, kad tiktų daugumai tipinių verslų, bet kartu pakankamai konkretus, kad nesijaustų šabloninis.
* **Minimalus „post-processing“ poreikis:** Turinys turi būti iškart tinkamas ne tik HTML formai, bet ir SEO prasme – su antraštėmis, meta tekstais, aprašymais.

**4.3 Vartotojui turi būti aišku, ką įvesti ir ko tikėtis**

* **Intuityvus klausimynas:** Vartotojas neturi jausti, kad pildo „formą“ – turi jaustis, kad atsako į pokalbį ar vedlį, kuris padeda kurti.
* **Įvesties pavyzdžiai:** Kiekvienas įvesties laukas turi būti paaiškintas su pavyzdžiais (pvz., „Pateikite paslaugos aprašymą – pvz., ‘veido valymas natūraliomis priemonėmis’“).
* **Rezultatų peržiūros kontekstas:** Turinys turi būti rodomas ne kaip „tekstų laukas“, o kaip realaus puslapio prototipas – taip padidinamas supratimas ir pasitikėjimas.

**4.4 Sistemoje turi būti lengva dirbti be išankstinių techninių žinių**

* **Ne IT žmonėms:** Sprendimas turi būti pritaikytas naudotojams, kurie neturi jokių žinių apie HTML, SEO ar tekstų struktūrizavimą.
* **„Smart defaults“ ir automatiniai pasiūlymai:** Sistemoje turėtų veikti „protingi numatytieji“ (pvz., pasiūlymas įterpti CTA, sukurti SEO meta aprašymą, įkelti paveikslėlį).
* **Minimali klaidų tikimybė:** Naudotojas neturi galėti sugadinti struktūros ar sukelti netvarkos net ir padaręs netikslią įvestį.

**4.5 Turinys turi atitikti rinkodaros praktikos standartus**

* **SEO orientacija:** Tekstai turi būti optimizuoti naudoti svetainėse – raktažodžių tankis, antraščių struktūra, meta aprašymai.
* **Konversijos principai:** Generuojami tekstai turi remtis konversijai svarbiais komponentais – įtikinėjimo struktūra, išskirtinumo įvardijimas, pasitikėjimo elementai (pvz., „klientai vertina...“, „+1000 naudotojų“).
* **Tonų ir auditorijų adaptacija:** Tekstai turi būti tinkami verslui B2B, paslaugoms, kūrybinėms veikloms ar nišiniams sektoriams – be „vienos kalbos visiems“.

**Išvada**

Autokūrėjo funkcija turi būti sukurta taip, kad naudotojas **pasitikėtų tuo, ką mato**, ir galėtų naudoti turinį **iš karto**, nesvarbu, ar jis būtų verslo savininkas, ar agentūra. Tai reiškia, kad sistema turi veikti ne tik kaip tekstų generatorius, bet kaip **turinio kūrimo padėjėjas** – suprantantis, aiškinantis, siūlantis ir pateikiantis ne „žalią tekstą“, bet **paruoštą, logišką, komunikacijos tikslus atitinkantį turinį**.

# 5. Techniniai komponentai

Autokūrėjo funkcija reikalauja kelių glaudžiai susijusių techninių modulių, kurie bendrai leidžia realizuoti visą turinio kūrimo eigą – nuo duomenų įvedimo iki publikavimo su patikra. Ši funkcija remiasi **API-first** logika, **moduline** architektūra ir **asociatyviu veiksmų eigos valdymu**, leidžiančiu ją integruoti į skirtingus naudotojo scenarijus ir darbo aplinkas.

**5.1 Architektūrinės nuostatos**

* **SaaS modelis / Web-based sąsaja**
  + Visi veiksmai atliekami naršyklėje.
  + Įrankis veikia kaip savarankiška aplikacija su galimybe vėliau integruotis į kitą platformą (pvz., klientų CMS ar turinio platformą).
* **API-first struktūra**
  + Visi moduliai – nuo šablonų valdymo iki publikavimo – turi būti pasiekiami per REST ar GraphQL API.
  + Tai užtikrina integraciją su: CMS, el. parduotuvėmis, projektų valdymo sistemomis, agentūrų dashboard’ais.
* **Modulinis veikimo principas**
  + Pagrindiniai komponentai vystomi kaip atskiros loginės posistemos su galimybe būti naudojamais atskirai ar kartu:
    - Turinio šablonų tvarkyklė
    - LLM (GPT) sąsajos sluoksnis
    - Teksto analizės ir pataisų logika
    - Vizualų integracijos modulis
    - Eksporto / publikavimo modulis
    - Patikros / verifikacijos sluoksnis

**5.2 AI integracijos komponentas (LLM)**

* **Modelių tipas:** OpenAI GPT arba analogiškas LLM.
* **Promptų valdymas:** Centralizuota promptų tvarkyklė leidžia:
  + kurti skirtingus režimus (pvz., „emociškai įtraukus“, „SEO draugiškas“, „konversijoms skirtas“),
  + naudoti skirtingus turinio tipus (apie mus, produktas, paslauga, naujiena),
  + koreguoti generavimo rezultatų lygmenį be UI keitimų.
* **Konfigūracijos parametrai:** temperatūra, max tokens, modelio versija, kalbos nustatymai.
* **Post-processing sluoksnis:**
  + Gramatinis ir stilistinis tikrinimas.
  + Raktažodžių patikra ir integracija.
  + Struktūros verifikavimas (antraštės, CTA, sąrašai).

**5.3 Vizualų integravimo komponentas**

* **Vaizdų generavimas (AI / teminiai raktiniai žodžiai):**
  + OpenAI DALL·E ar analogiškų modelių integracija.
  + Galimybė naudoti išorinį šaltinį (pvz., Pexels, Unsplash API).
* **Formatų palaikymas:**
  + Web, kvadratiniai, social (16:9, 4:5, 1:1).
* **Automatinė rekomendacija pagal tekstą:**
  + Pasiūlyti paveikslėlio idėjas pagal antraštes, paslaugų pobūdį, CTA toną.
* **Įkėlimo / keitimo modulis:**
  + Naudotojas gali keisti, trinti, pakeisti poziciją, pritaikyti filtrus ar išvestines.

**5.4 Publikavimo ir patvirtinimo komponentas**

* **CMS integracija (WordPress\Prestashop MVP stadijoje):**
  + Eksportas į HTML arba per WordPress\Prestashop REST API (jei įgalinta).
  + Galimybė nurodyti publikavimo aplinką: *preview* / *live*.
* **Publikavimo scenarijų tvarkymas:**
  + Rankinis / automatinis / miksuotas režimas.
  + Nustatymai, ar reikia papildomo patvirtinimo.
* **Verifikacijos sistema:**
  + Auto-testavimas (turinio rodymas, struktūra, HTML validumas).
  + Žmogaus peržiūros žymėjimas.
  + Sistemos notifikuojamas „viskas OK“ statusas arba grąžinimas korekcijai.

**5.5 Naudotojo sąsajos komponentai**

* **Turinio kūrimo vedlys / klausimynas**
  + Dinaminė forma su AI pagalba, kuri keičiasi priklausomai nuo pasirinktų šablonų.
* **Peržiūros režimas (preview-as-real-page):**
  + Rodo sukurtą tekstą ne „kaip lauko reikšmę“, o kaip svetainės bloką.
  + Įtraukiami stiliai, vizualai, antraštės.
* **Būsenų ir užduočių valdymas:**
  + Žymėjimas: „sukurta“, „redaguojama“, „publikuota“, „nepatvirtinta“.
  + Delegavimo funkcija (pvz., „siųsti peržiūrai“, „priskirti korekcijai“).

**5.6 Duomenų saugojimas ir saugumas**

* **Laikinas turinio saugojimas:** Sesijos metu (jei naudotojas neprisijungęs).
* **Ilgalaikis saugojimas:** Prisijungusiam naudotojui – analizės, tekstai, vaizdai, istorija.
* **Versijų valdymas:** Galimybė išsaugoti kelias versijas, lyginti jas, grįžti prie ankstesnių.

**Išvada**

Autokūrėjo techninė architektūra turi būti suprojektuota kaip lanksti, modulinė, API-first sistema, integruojanti LLM modelių galimybes, vizualų parinkimo logiką, peržiūros UX ir publikavimo mechanizmą su patikros sluoksniu. Tik toks požiūris leidžia pasiekti realiai naudotiną ir profesionalų rezultatą, kurį galima patikėti tiek mažam verslui, tiek agentūrai.

# 6. MVP apimtis

Turinio autokūrėjo funkcijos MVP (Minimalus Veikiantis Produktas) turi užtikrinti, kad naudotojas galėtų:

* **pasirinkti šabloną** (turinio tipą),
* **įvesti esminę informaciją** (tema, tikslas, stilius),
* **gauti kokybiškai sugeneruotą tekstą** su minimaliu redagavimo poreikiu,
* **eksportuoti arba publikuoti turinį**, ir
* **peržiūrėti rezultatą realiame kontekste** (preview režimu).

**6.1 Į funkciją įtraukiami komponentai MVP etape**

|  |  |
| --- | --- |
| **Komponentas** | **Apimtis MVP versijoje** |
| **Šablonų pasirinkimas** | 5–7 baziniai puslapių tipai: „Apie mus“, „Paslaugos“, „Kontaktai“, „Produktas“, „Blogas“ |
| **Įvedimo forma (vedlys)** | Tema / raktiniai žodžiai / stilius / kalba / ilgis |
| **GPT integracija** | Vieno modelio (pvz., GPT-4) versija, centralizuotas promptas kiekvienam tipui |
| **Teksto versijos** | Sugeneruojamos 2 alternatyvos, viena parodoma kaip numatytoji |
| **Post-processing (tekstui)** | Gramatika, CTA patikrinimas, struktūros tikrinimas |
| **SEO komponentai** | Meta pavadinimas, meta aprašymas, raktažodžių sklaida |
| **Peržiūra** | „Preview-as-page“ režimas su H1, paragrafais, CTA |
| **Eksportas** | HTML / Notion (kopijuoti), WordPress\Prestashop copy-paste (rankinis) |
| **Publikavimo paruošimas** | Žymėjimas: „pasiruošęs“ / „reikia patvirtinimo“ / „publikuotas“ |
| **Patvirtinimo logika** | Rankinis patvirtinimas prieš publikavimą |
| **Demo publikavimo režimas** | Turinys publikuojamas į „demo“ režimą prieš live versiją |
| **Papildomos funkcijos** | Naudotojo komentarų laukas, statuso perjungimas, analizės istorijos saugojimas |

**6.2 Neįtraukiami į MVP, bet numatomi vėliau**

|  |  |
| --- | --- |
| **Komponentas** | **Pastabos** |
| Vaizdų generavimas (AI / biblioteka) | Planuojama kaip 2 etapo funkcionalumas |
| Automatinis publikavimas į CMS (API) | Tik eksportas MVP etape |
| Patikros automatika (autotestavimas) | Pirmame etape naudojamas tik rankinis tvirtinimas |
| Delegavimas ir komentavimas komandoje | Pradžioje tik individualus naudotojas |
| Versijų istorija / lyginimas | Neįgyvendinama MVP etape |
| Agentūrų režimas (multi-project) | Vėlesnė plėtros kryptis |

**6.3 MVP sėkmės kriterijai (vertės validacija)**

* Naudotojas be techninių žinių geba sukurti tekstą ir naudoti jį savo svetainėje
* Naudotojas aiškiai supranta, ką įvesti ir ką gavo
* 70 % sugeneruoto turinio publikuojama be korekcijų
* Produkto komandai grįžtamasis ryšys leidžia identifikuoti:
  + naudotojų tonų pageidavimus,
  + dažniausiai kuriamus puslapius,
  + pasitikėjimo AI ribas.

**Išvada**

MVP versija turi būti tvirta, funkcionali, bet siauro fokuso – padedanti išspręsti pagrindinę naudotojo problemą: **„kaip greitai ir be rašytojo sukurti svetainės tekstą, tinkamą viešam publikavimui.“**

MVP neturi bandyti „išspręsti visko“ – jos vertė turi būti patikrinta per realią naudą naudotojui ir pasirengimą eiti į rinką su esminiu sprendimu.

# 7. Roadmap: tolesnės versijos

Turinio autokūrėjo MVP suformuoja branduolinį sprendimą – galimybę sukurti ir publikuoti tekstą į svetainę be profesionalaus turinio kūrėjo pagalbos. Vėlesnės versijos plečia produkto vertę funkciniu, UX, techniniu ir verslo modelio lygmenimis.

**7.1 Plėtros kryptys pagal funkcinius modulius**

**1. Turinio kūrimo plėtra**

* Daugiau šablonų tipų (pvz., karjera, DUK, atsiliepimai, nišiniai landing’ai)
* Interaktyvūs turinio blokai (pvz., CTA su mygtukais, „susisiekite dabar“ su formos integracija)
* Rinkodaros tikslų režimai (pvz., SEO pozicijos gerinimui, social media srautui, konversijų kėlimui)

**2. Vizualų generavimas ir valdymas**

* AI iliustracijos pagal tekstą (pvz., DALL·E)
* Automatizuoti paveikslėlių raktinių žodžių pasiūlymai
* Vizualų integracija į tekstą su stilistikos nuoseklumu
* Vaizdų publikavimas kartu su tekstu

**3. CMS integracijos**

* WordPress\Prestashop API publikavimas (per REST)
* Shopify / Wix / Webflow eksportas
* Vieno mygtuko publikavimas
* Galimybė redaguoti tekstą tiesiogiai CMS’e per iframe / plugin’ą

**4. Publikavimo ir tvirtinimo logika**

* „Demo → Live“ perėjimas su tiksliu pokyčių log’u
* Patvirtinimo srautas su atsakingo asmens nurodymu
* Automatinis testavimas (HTML, CSS, matomumas)
* Analitinis grįžtamasis ryšys: ar puslapis buvo sėkmingai indeksuotas, peržiūrėtas, veikė CTA

**5. Komandinės funkcijos**

* Naudotojų rolės: redaktorius, tvirtintojas, administratorius
* Užduočių delegavimas (pvz., „pateikti korekcijas“, „paruošti publikavimui“)
* Komentarų sistema tekstų peržiūroje
* Turinio būsenų valdymas komandoje

**6. Versijų istorija ir monitoringas**

* Kiekvieno sukurtos versijos saugojimas
* Galimybė palyginti versijas: „kas pasikeitė“
* Periodinis turinio kokybės monitoringas (pvz., kas 6 mėn.)
* „Ar šis turinys vis dar aktualus?“ tipo notifikacijos

**7. Agentūrų režimas**

* Kelių projektų / svetainių valdymas vienoje paskyroje
* Kliento-peržiūros režimas su white-label ataskaitomis
* Agentūros naudotojo statistika: produktyvumas, korekcijų kiekis, laikas per užduotį

**7.2 Technologinės kryptys**

|  |  |
| --- | --- |
| **Kryptis** | **Tikslas** |
| No-code įskiepių kūrimas | Leisti vartotojui naudotis tiesiogiai per WordPress / Prestashop / Shopify įskiepius |
| AI modelių personalizavimas | Leisti vartotojui „išmokyti“ sistemą savo stiliaus ir preferencijų |
| Promptų redagavimo galimybės | Įrankis pažengusiems naudotojams koreguoti promptų struktūrą (admin) |
| Automatiniai A/B tekstų testai | Sukurti 2 versijas ir matuoti konversiją (susiejus su analitika) |

**7.3 Laiko horizontas (rekomenduojamas)**

|  |  |  |
| --- | --- | --- |
| **Etapas** | **Laikotarpis** | **Apimtis** |
| **MVP (V1.0)** | 2025 Q2 | Tekstas, preview, eksportas, rankinis publikavimas |
| **V1.1** | 2025 Q3 | CMS API, demo → live, AI vaizdai, daugiau šablonų |
| **V1.2** | 2025 Q4 | Komandinė sąveika, komentarai, tvirtinimo grandinė |
| **V2.0 (Agentūrų režimas)** | 2026 Q1 | Projektų valdymas, klientų sąsajos, white-label |

**Išvada**

Autokūrėjo funkcija turi aiškų augimo kelią – nuo MVP, kuris išsprendžia esminę turinio kūrimo problemą, iki daugiafunkcinės turinio valdymo platformos, veikiančios agentūrose ir įmonėse kaip **AI turinio operacijų centras**.

Kiekvienas išplėtimas turi būti daromas tik po MVP validacijos, remiantis realių naudotojų patirtimi ir grįžtamuoju ryšiu.

# 8. Rizikos

**8.1 AI generavimo kokybės rizikos**

* **Per daug generiškas turinys:** LLM (pvz., GPT) gali sugeneruoti bendrinius, „neužkabinantį“ tekstus, ypač jei įvestis per trumpa ar abstrakti.
  + *Valdymas:* Įdiegti stipresnę promptų logiką ir papildomas stiliaus personalizavimo parinktis.
* **Konteksto nesuvokimas:** AI gali nesuprasti verslo specifikos, rinkos ar tikslinės auditorijos.
  + *Valdymas:* Vartotojui siūlomi „nišos profiliai“ arba leidžiama naudoti atmintį (per API) apie verslo veiklą.
* **Gramatinės ar stilistinės klaidos:** Nepaisant LLM galimybių, gali būti likusių klaidų, ypač lokalizuotame tekste.
  + *Valdymas:* Integruoti papildomą kalbos tikrinimo modulį (pvz., LanguageTool ar Grammarly API).

**8.2 UX / naudojimo rizikos**

* **Neaišku, ką įvesti:** Vartotojas nežino, kokią informaciją reikia pateikti, kad gautų kokybišką rezultatą.
  + *Valdymas:* Aiškūs pavyzdžiai, vedlys su paaiškinimais, patarimai.
* **Nepasitikėjimas AI rezultatu:** Vartotojas gali abejoti, ar generuotas tekstas tinkamas naudoti profesionaliai.
  + *Valdymas:* Paaiškinimų blokas („kodėl sukurtas toks tekstas“), demo peržiūra, atsiliepimų rinkimas.
* **Per sudėtingas funkcionalumas paprastam naudotojui:** Pvz., tvirtinimo / publikavimo žingsnių perteklius.
  + *Valdymas:* MVP etape siūlomas tik supaprastintas darbo scenarijus su galimybe išjungti nereikalingas funkcijas.

**8.3 Techninės rizikos**

* **LLM atsakymų nestabilumas:** Tas pats promptas gali grąžinti skirtingus rezultatus, priklausomai nuo modelio ar sesijos.
  + *Valdymas:* Fiksuoti promptus, naudoti ribotą temperatūrą, išsaugoti kiekvieną versiją.
* **Eksportavimo / publikavimo klaidos:** Įkeltas tekstas gali neteisingai atvaizduotis CMS sistemoje.
  + *Valdymas:* Publikavimo testavimo sluoksnis su HTML validacija, demo preview mechanizmas.
* **Integracijų sudėtingumas:** Ne visos CMS leidžia lengvą integraciją be papildomų įskiepių ar techninės pagalbos.
  + *Valdymas:* MVP stadijoje naudoti copy-paste / HTML eksportą, o API integracijas vystyti kaip papildinį.

**8.4 Verslo rizikos**

* **Funkcija nesukuria pakankamos vertės MVP etape:** Jei naudotojai nesugeba pritaikyti generuoto turinio realiam naudojimui.
  + *Valdymas:* Ankstyvas testavimas su beta naudotojais, rinkos validacija.
* **Konkurencinis spaudimas:** Dideli žaidėjai (pvz., Wix AI, Jasper, Writesonic) gali pasiūlyti panašią funkciją greičiau.
  + *Valdymas:* Skirtis per UX paprastumą, lokalizacijos kokybę, integracijas su svetaine.
* **Naudotojo duomenų saugumo baimė:** Vartotojai gali bijoti į sistemą įkelti jautrų turinį.
  + *Valdymas:* Skaidri duomenų politika, aiškus paaiškinimas, kad turinys nesiunčiamas ar nesaugomas be sutikimo.

**8.5 Organizacinės rizikos**

* **Komandos resursų nepakankamumas MVP įgyvendinimui pilna apimtimi**
  + *Valdymas:* Aiškiai apibrėžta MVP riba (žr. 6 skyrių), išskaidyta per iteracijas.
* **Promptų / turinio kokybės priežiūros poreikis auga greičiau nei komanda gali reaguoti**
  + *Valdymas:* Įdiegti atsiliepimų rinkimo ir automatinio vertinimo sistemą, kad galėtume greitai identifikuoti blogus rezultatus.

**Išvada**

AdoreIno autokūrėjo funkcija yra technologiškai įmanoma ir rinkos poreikius atitinkanti, tačiau norint, kad ji būtų realiai naudojama, būtina iš anksto suvaldyti pagrindines rizikas: nuo AI rezultatų patikimumo iki UX paprastumo. Todėl **testavimas, lankstus dizainas, aiškūs pasirinkimai ir palaikomas grįžtamasis ryšys** bus kritiniai sėkmei.