

**VILNIAUS KOLEGIJA**

**ELEKTRONIKOS IR INFORMATIKOS FAKULTETAS**

**Programinės įrangos katedra**

**PROGRAMŲ SISTEMŲ INŽINERIJA**

DOKUMENTACIJA

SD - 6531BX028 PI17B

|  |  |  |
| --- | --- | --- |
| **STUDENTAI** |  | Augustinas Jonušas |
|  |  | Augustas Matorka  Karolis Žilevičius  Simonas Pajarskas |
| **DĖSTYTOJA** |  | J. VAIČIULYTĖ |

Vilnius

2020

**TURINYS**

[ĮVADAS 3](#_Toc29145629)

[TRUMPAS PROJEKTO PLANO APRAŠYMAS 4](#_Toc29145630)

[FUNKCINIAI, NEFUNKCINIAI REIKALAVIMAI 4](#_Toc29145631)

[ARCHITEKTŪRA / DIZAINAS 5](#_Toc29145632)

[TESTAVIMO PLANAS 11](#_Toc29145633)

[VARTOTOJO INSTRUKCIJA 12](#_Toc29145634)

[IŠVADOS 17](#_Toc29145635)

# ĮVADAS

**Kam skirta kuriama sistema:** ši sukurta programų sistema, bus skirta asmenis, kuriems reikia pagalbos ieškant mašinos internetu. Įvedus automobilio skelbimo nuorodą į šią programą, vartotojas matys visas automobilio specifikacijas ir palyginimus su kitais tos pačios markės automobiliais.

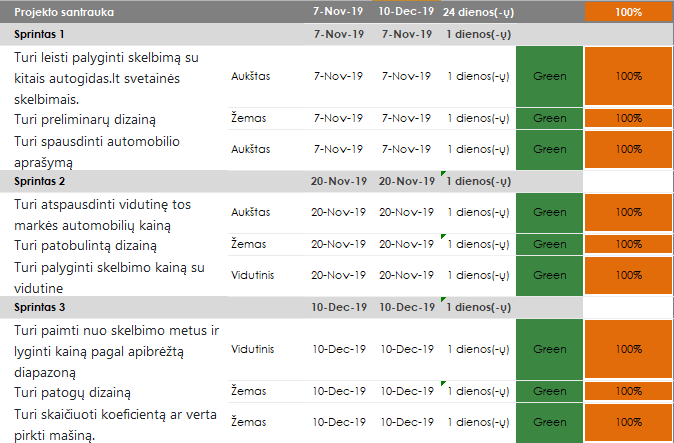
**Būsimi vartotojai:** Vartotojas šios programų sistemos gali būti bet kas, kas internetu ieško pirkti automobilį.

**Paskirtis:** Šios programų sistemos vienintelis tikslas yra suteikti informacijos vartotojui apie automobilio skelbimą.

**Patalpinto projekto nuorodą:** <https://github.com/Augiiz/XPlus>

# TRUMPAS PROJEKTO PLANO APRAŠYMAS

Projektas buvo pradėtas lapkričio 7 d. Suplanavus sistemos funkcinius ir nefunkcinius reikalavimus tą pačią dieną buvo pradėtas programavimo procesas, kuris buvo baigtas sausio 10 d.



1 pav. Sprinto pavyzdys

# FUNKCINIAI, NEFUNKCINIAI REIKALAVIMAI

**Funkciniai reikalavimai**

1. Turi leisti palyginti skelbimą su kitais autogidas.lt svetainės skelbimais.
2. Turi spausdinti automobilio aprašymą
3. Turi atspausdinti vidutinę tos markės automobilių kainą
4. Turi palyginti skelbimo kainą su vidutine
5. Turi paimti nuo skelbimo metus ir lyginti kainą pagal apibrėžtą diapazoną

**Nefunkciniai reikalavimai**

1. Sistema turi veikti korektiškai.
2. Sistemos turinys neturi įžeisti naudotojo.
3. Sistema turi atitikti iš politinių apribojimų išplaukiančius reikalavimus.
4. Sistema turi atitikti iš ekonominių apribojimų išplaukiančius reikalavimus.
5. Sistema turi būti pritaikyta Windows 10 arba naujesniai operaciniai sistemai.

# ARCHITEKTŪRA / DIZAINAS

**Architektūros aprašymas**

1. **Našumas (Performance)**

Sistemos funkcionalumas yra parašytas python kalba, todėl programa veikia labai sparčiai.

1. **Saugumas (Security)**

Naudojama sluoksninė architektūra, kur svarbiausi dalykai yra prieinami tik žemiausiame lygyje.

1. **Prieinamumas(Availability)**

Sistema suprojektuota taip, kad tam tikras jos dalis keičiant reikėtų sustabdyti programą.

1. **Palaikomumas (Maintainability)**

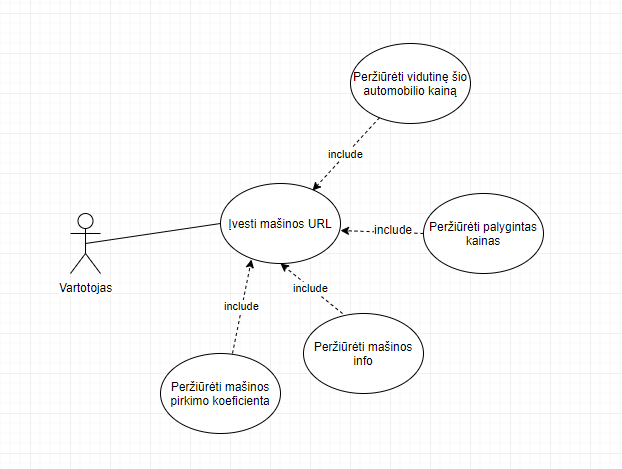
Naudojamus komponentus galima nesunkiai pakeisti kitais, kaip ir programos grafinę sąsają.

1. **Architekturos sluoksniai**

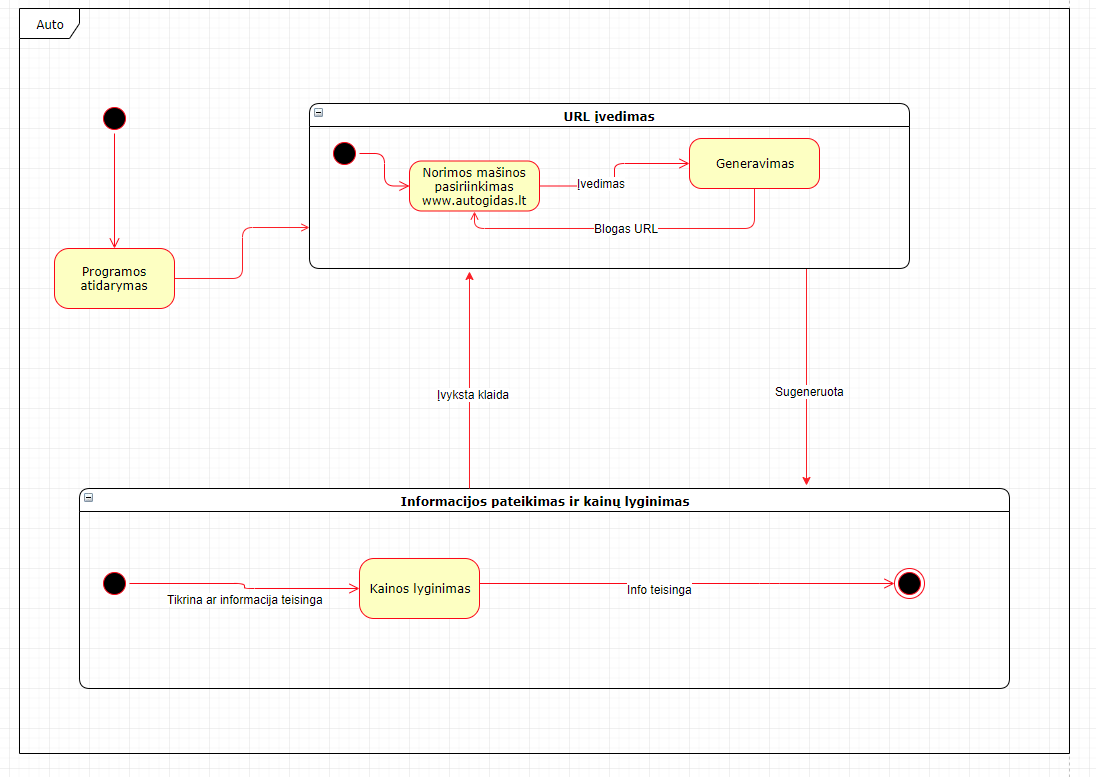
|  |
| --- |
| Grafinė sąsaja (visual studio) |
| Funkcionalumas (python) |

1lentelė. Architektūros sluoksnių pavyzdys

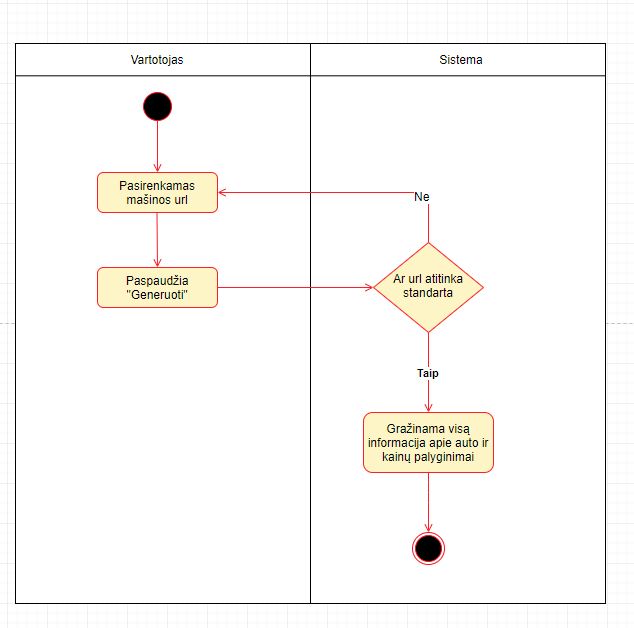
Paveikslėlyje (žr. 2pav) galima matyti sistemos UML diagramą. Šioje diagramoje, vaizduojama, kokie veikėjai bus kuriamoje sistemoje bei ką kiekvienas veikėjas galės atlikti naudojant šią sistemą.



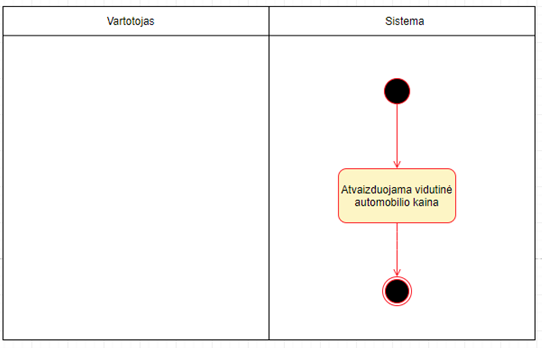
2 pav. UML diagrama



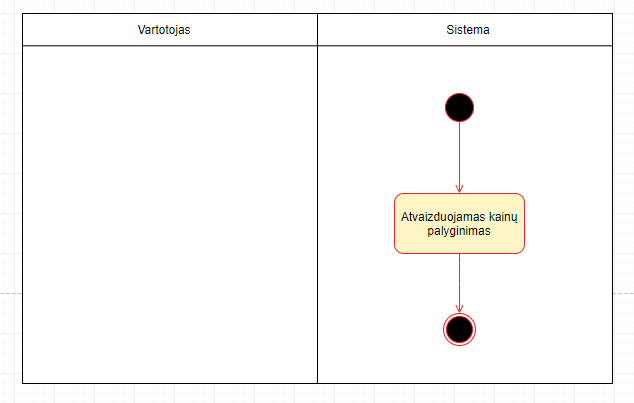
3 pav. Vartotojo būsenos diagrama



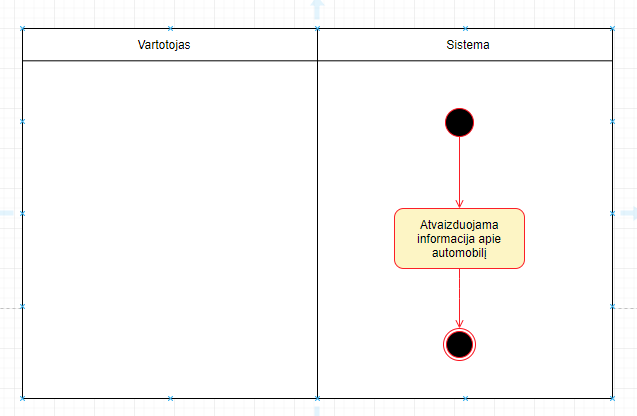
4 pav. Vartotojas įveda automobilio adresą



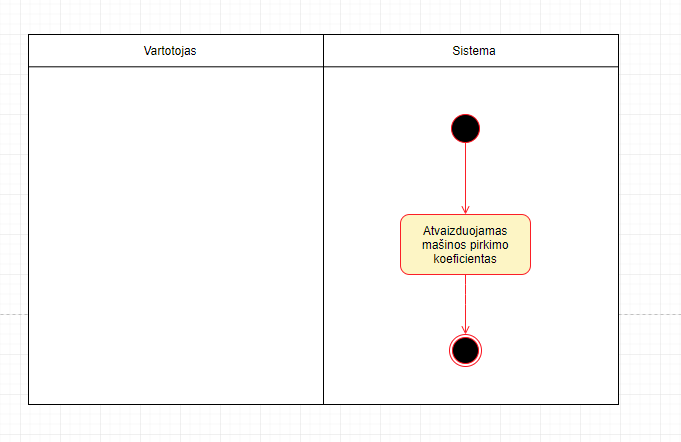
5 pav. Sistemos vidutinės automobilio kainos veiksmų diagrama



6 pav. Sistemos kainų palyginimo veiksmų diagrama



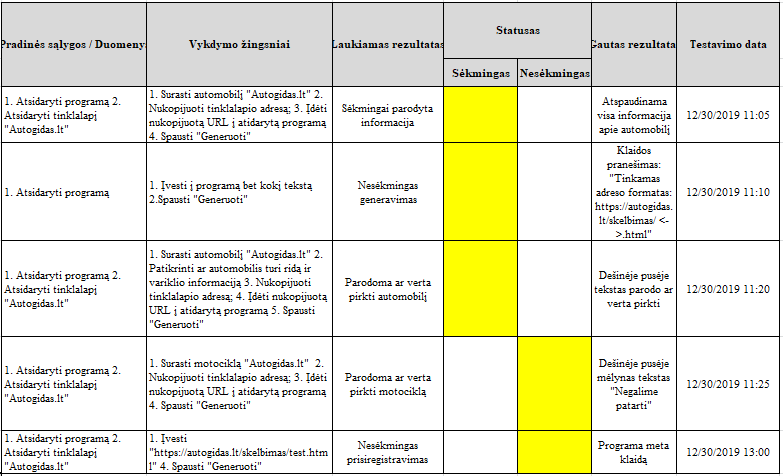
7 pav. Sistemos informacijos atvaizdavimo veiksmų diagrama



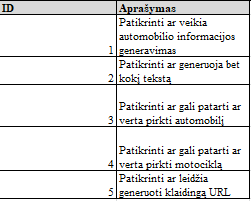
8 pav. Sistemos pirkimo koeficiento atvaizdavimo veiksmų diagrama

# TESTAVIMO PLANAS

Kuriamai programų sistemai buvo kuriami testai. Šių testų paskirtis buvo patikrinti svarbiausias programos funkcionalumo dalis. Testams atlikti naudojome unit testus, nes Visual Studio aplinkoje jais yra labai patogu naudotis.



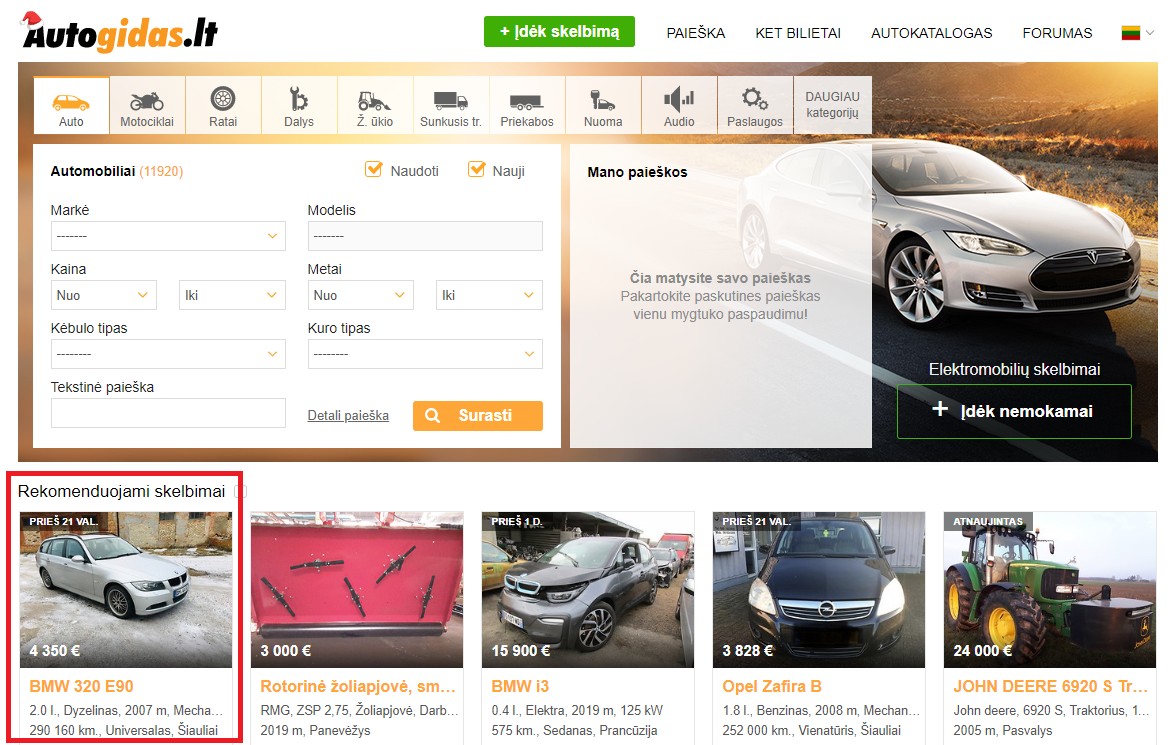
9 pav. Testavimo scenarijai



10 pav. Testavimo scenarijų aprašymai

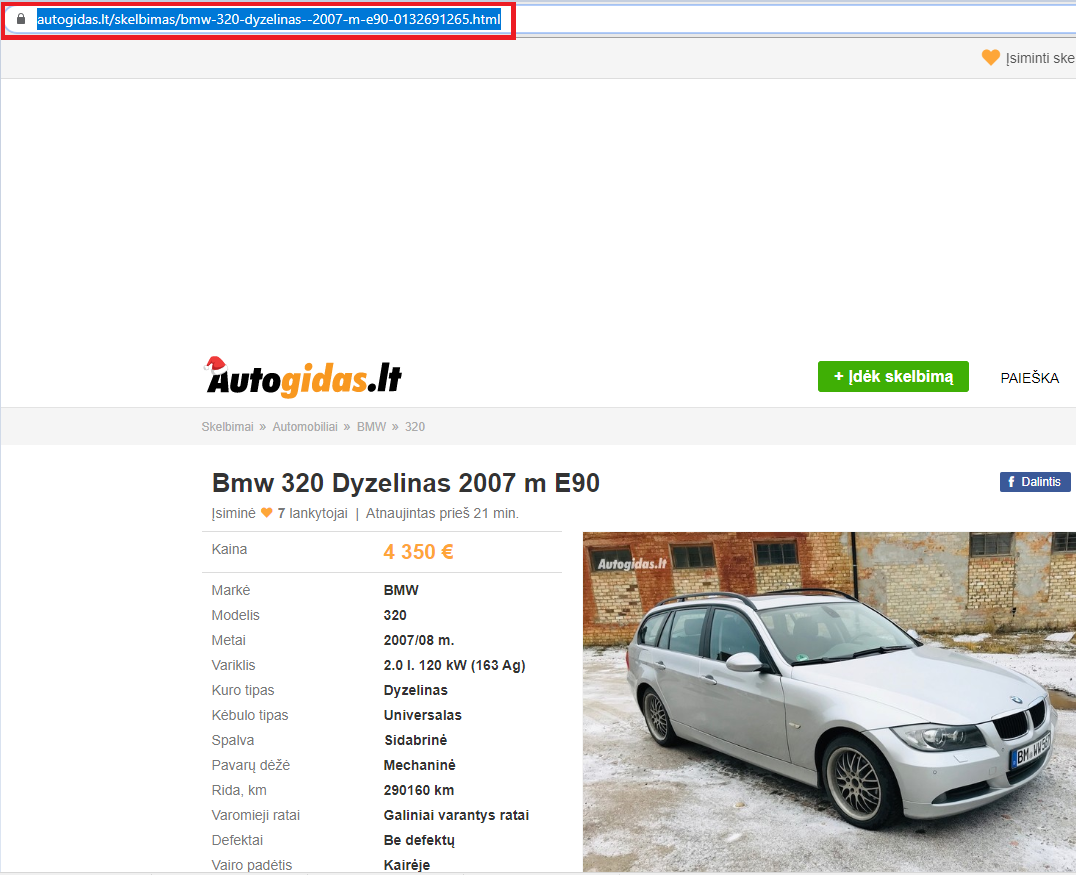
# VARTOTOJO INSTRUKCIJA

Šioje skiltyje bus pademonstruotas programos funkcionalumas ir kaip reikia ja naudotis. Pirmas žingsnis yra nueiti į svetainę [autogidas.lt](http://www.autogidas.lt) ir susirasti automobilio skelbimą, kuris Jus domina.



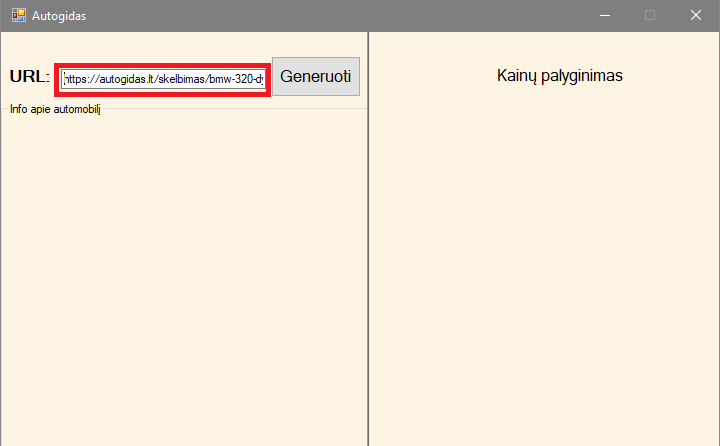
11 pav. Skelbimo pasirinkimas Autogido svetainėje

Antras žingsnis - paspaudus ant automobilio skelbimo programos naudotojas turi nukopijuoti šį adresą.



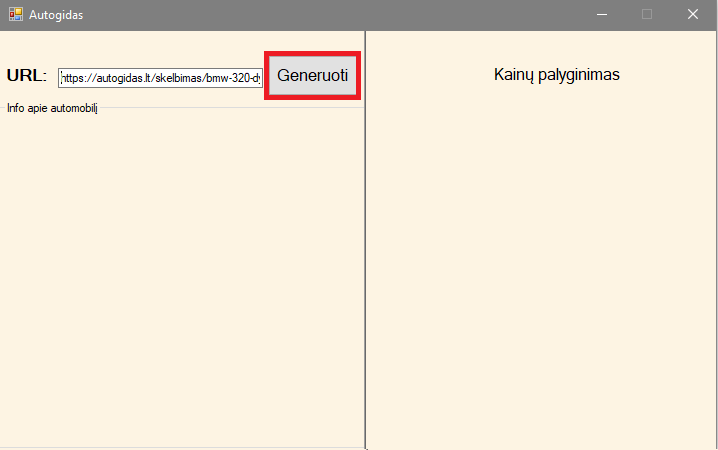
12 pav. Automobilio skelbimo adreso kopijavimas

Trečias žingsnis – paleidžiama programa ir automobilio skelbimo adresas įvedamas į šį laukelį.



13 pav. Automobilio skelbimo adreso įklijavimas į programą

Programos naudotojas spaudžia mygtuką generuoti, kad pamatytų visą informaciją apie automobilį.



14 pav. Programos paleidimas

Jeigu naudotojo įvesta nuoroda yra taisyklinga, tuomet jam bus atspausdinama visa informaciją apie automobilį.



15 pav. Vartotojui atspausdinama informacija apie automobilį

# IŠVADOS

Šio projekto tikslas buvo sukurti programą, kuri naudotojui padės nuspręsti apie automobilio skelbimo vertę. Galime drąsiai teigti, kad projektas buvo sėkmingas, nes įgyvendinome visą funkcionalumą be jokių problemų. Kuriamos programos darbas buvo paskirstytas sąžiningai. Simonas Pajarskas buvo atsakingas už testavimo atvejus, Augustinas Jonušas ir Karolis Žilevičius buvo atsakingi už pačią programą bei jos funkcionalumą, o Augustas Matorka už dokumentaciją. Tačiau nebuvome visiškai įspausti į savo roles, nuolatos komunikavome ir padėjome vienas kitam įvairiais klausimais ar problemomis. Projekto kūrimas prasidėjo lapkričio 7d. ir pasibaigė gruodžio 10.