



Kauno technologijos universitetas

Informatikos fakultetas

T120B165 Saityno taikomųjų programų projektavimas

Laboratorinis darbas Nr. 3

Martynas Kuliešius IFF-1/9

Studentas

dėst. Tomas Blažauskas

dėst. Eligijus Kiudys

dėst. Simonas Baltulionis

Dėstytojas

KAUNAS, 2024

Turinys

1. Uždavinio aprašymas	3
2. Architektūros aprašymas.....	3
Backend API:	3
Frontend:	3
Talpinimas:	3
3. Atliktos užduoties UML diagramos	4
UML diegimo diagrama:	4
4. Ranka piešti projekto wireframes	5
5. Atliktos užduoties demonstracija ir dokumentacija	19
Git saugyklos nuorodos:.....	19
Svetainės nuoroda:	19
6. Išvados	20

1. Uždavinio aprašymas

Projekto tikslas – sukurti traktorių forumą, kuriame naudotojai galėtų naršyti traktorius, kurti su jais susijusius skelbimus, komentuoti, tvarkyti savo profilius bei atlikti kitus veiksmus pagal jų vaidmenį sistemoje. Sistema turi palaikyti tris pagrindinius vartotojų tipus: svečias, moderatorius, administratorius, kiekvienam iš jų suteikiant specifines funkcijas.

Svečiai gali peržiūrėti traktorių kategorijas, bei peržiūrėti skelbimus. Prisiregistravę ir prisijungę naudotojai gali kurti ir redaguoti savo skelbimus pasirinktiems traktoriams bei kurti ir redaguoti savo komentarus.

Moderatoriai gali viską ką ir svečiai, tačiau gali taip pat ir trinti visus komentarus, bei skelbimus.

Administratoriai gali viską, ką ir moderatoriai, ir turi teisę pridėti naujus traktorius, valdyti ir redaguoti sistemos naudotojus.

Sistemos backend dalis sukurta naudojant Node.js ir Express, o frontend – su Vue.js. Tarpusavio komunikacija užtikrinama naudojant autentifikaciją per slapukus (cookies).

2. Architektūros aprašymas

Backend API:

Projekto backend sukurtas pasitelkiant Node.js ir Express. Duomenų bazės sprendimui naudoju MongoDB, kuri užtikrina sklandų duomenų valdymą. Sistema atitinka REST API standartus ir užtikrina duomenų integralumą per susijusių įrašų trynimą (cascade delete). Autentifikacija vykdoma per slapukus, o prieigos valdymas pagal naudotojo rolę užtikrina, kad naudotojai galėtų pasiekti tik jiems skirtus išteklius.

Frontend:

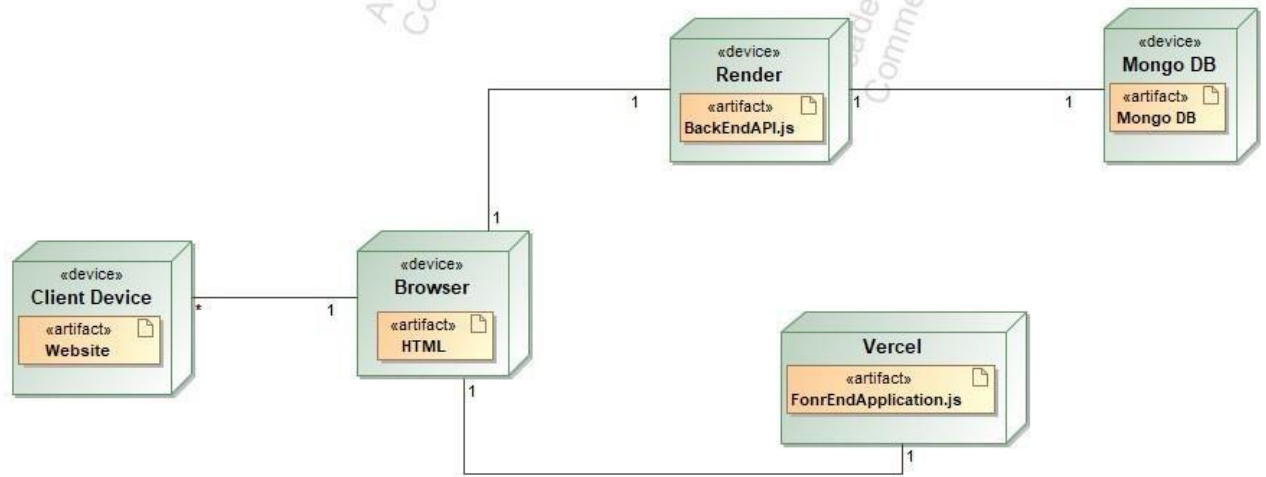
Frontend dalyje projektas sukurtas naudojant Vue.js. Pagrindiniai įrankiai, naudojami šiame projekte, apima Vue Router, kuris atsakingas už puslapių navigaciją, Pinia, naudojama globalios būsenos valdymui, Vee-Validate, kuris padeda validuoti formas ir užtikrina, kad vartotojai teisingai užpildytų reikalingus duomenis, ir Axios, skirtas HTTP užklausoms atlikti, užtikrinant greitą ir efektyvų duomenų mainą su serveriu.

Talpinimas:

- Frontend talpinamas [Vercel](#) platformoje, užtikrinant greitą ir patikimą aplikacijos pasiekiamumą visame pasaulyje.
- Backend talpinamas [Render](#) platformoje.

3. Atliktos užduoties UML diagramos

UML diegimo diagrama:



4. Ranka piešti projekto wireframes

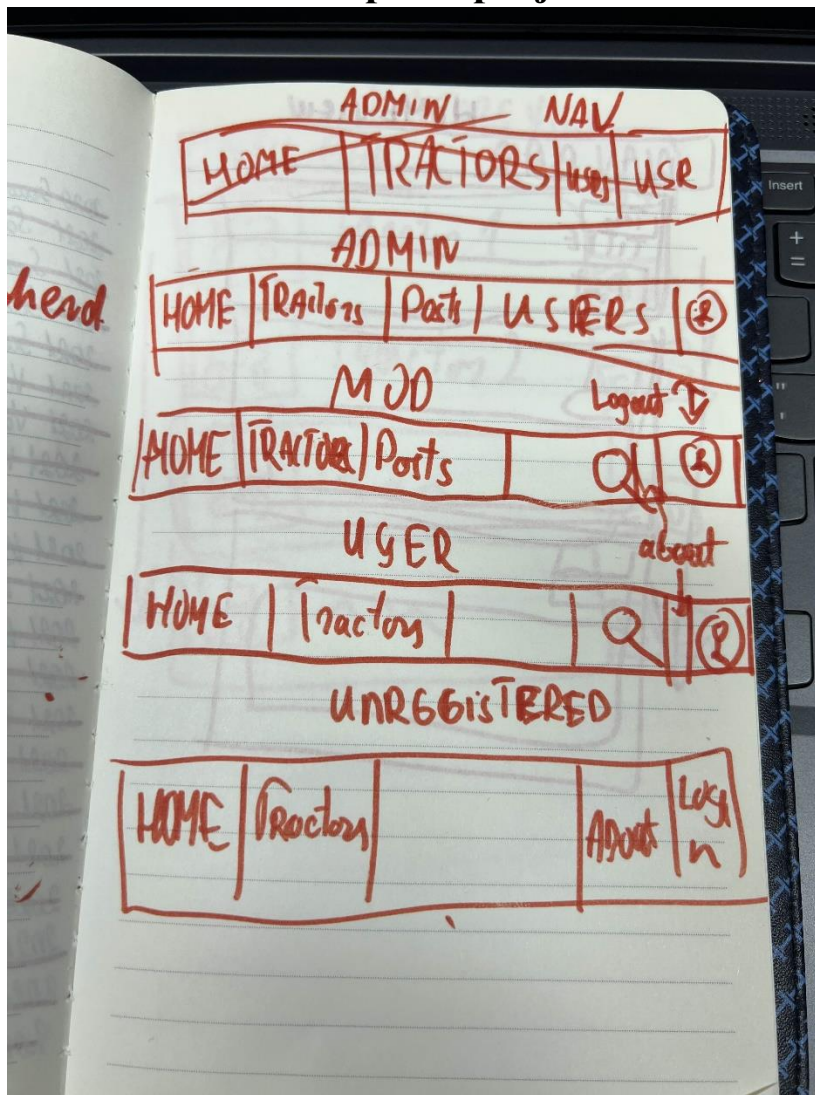


Figure 1Nav Bar wireframes

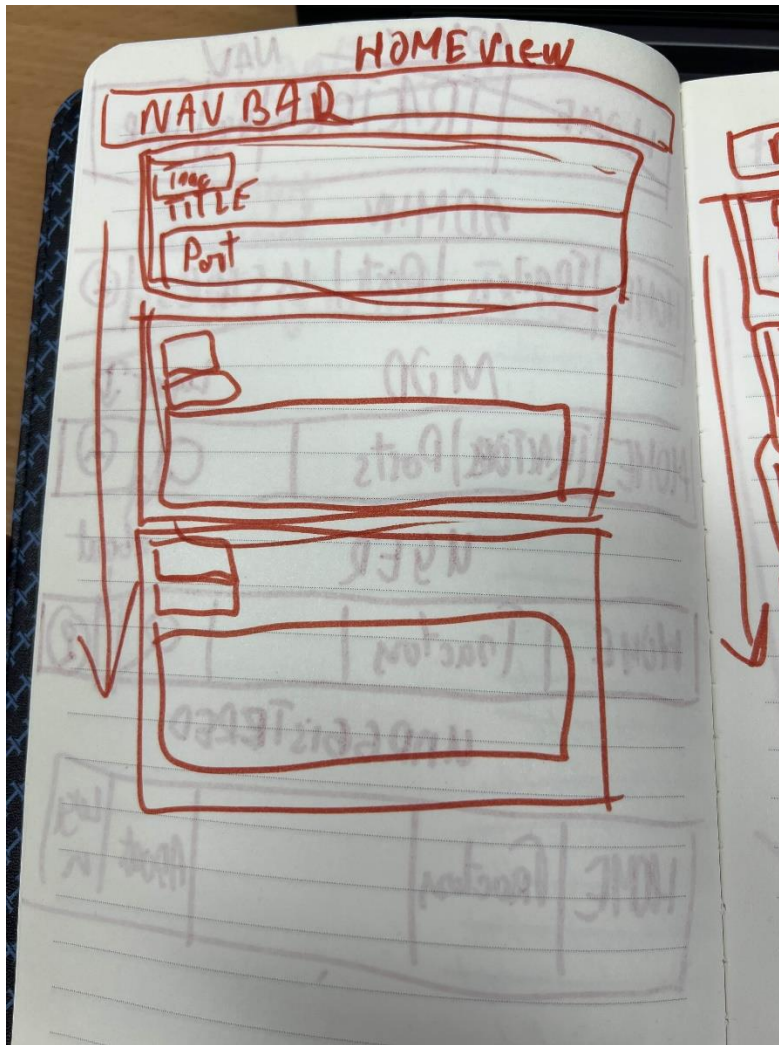


Figure 2 Homeview wireframe

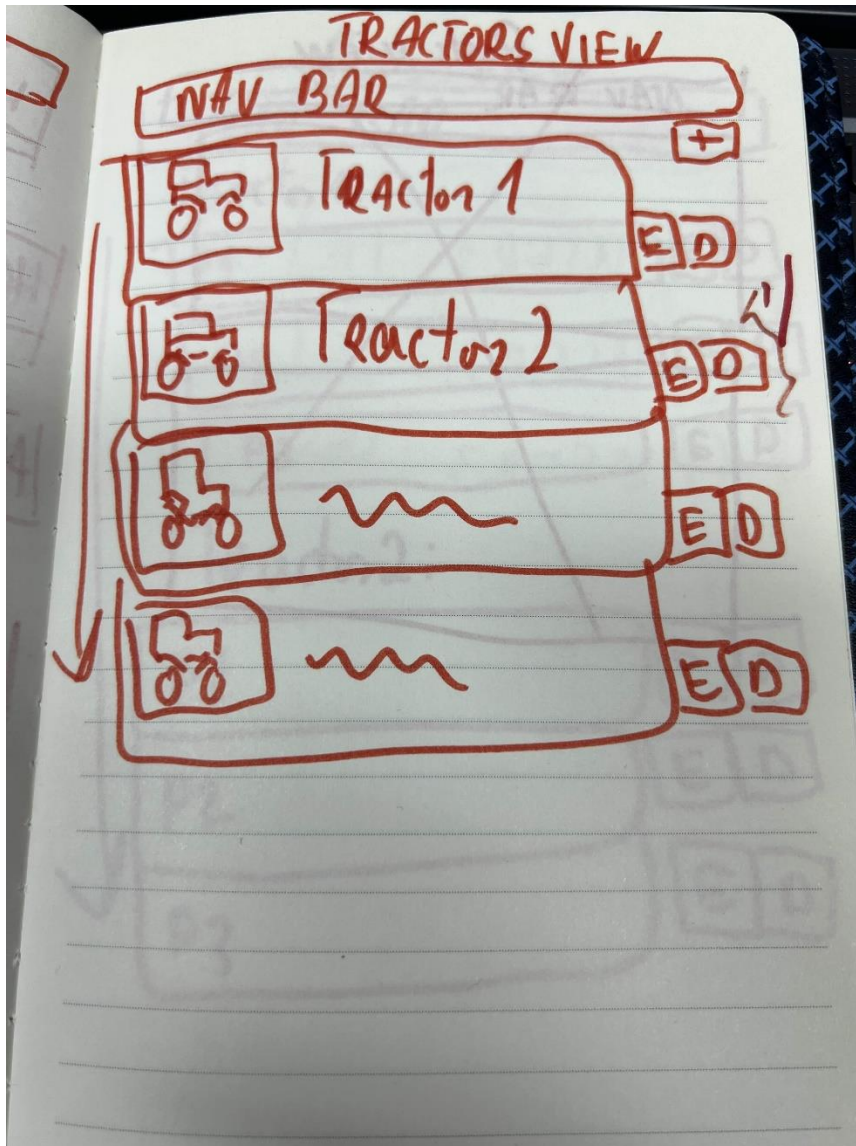


Figure 3 Tractors view wireframe

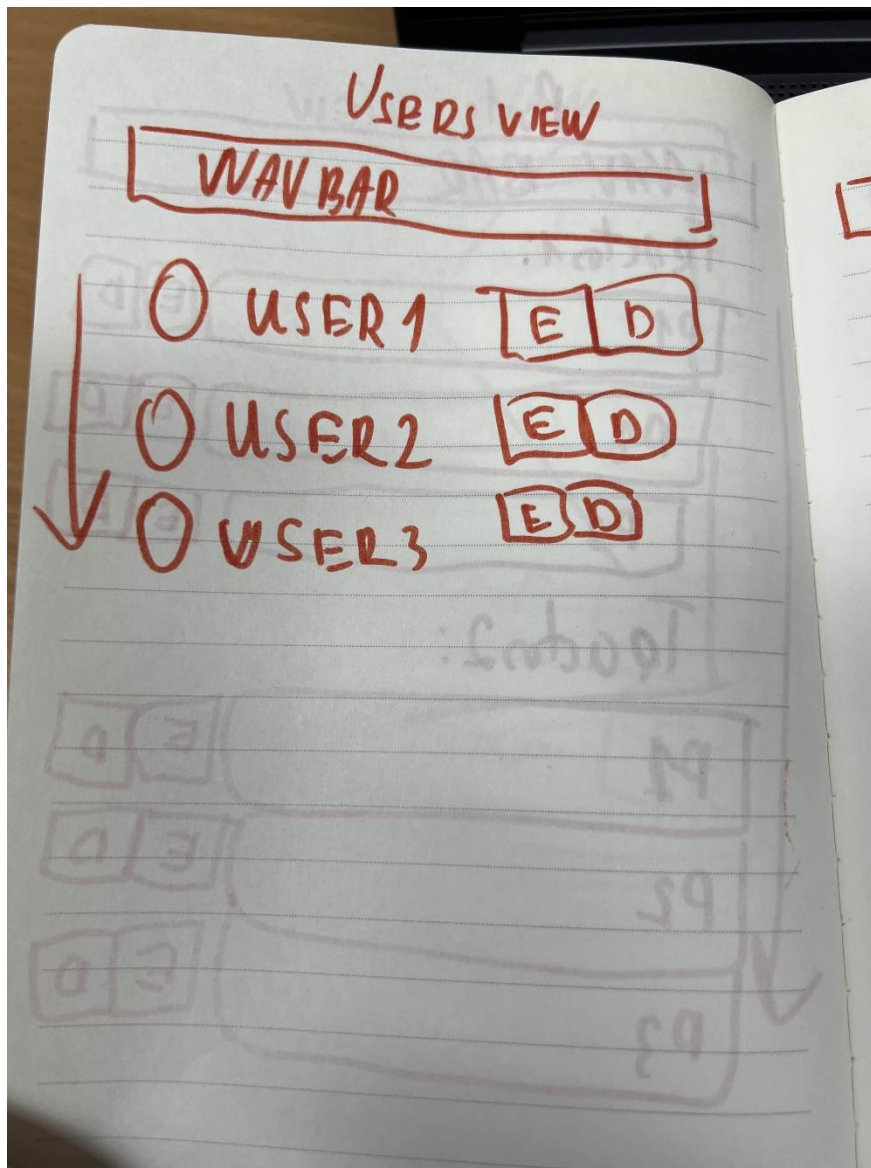


Figure 4 Users page wireframe

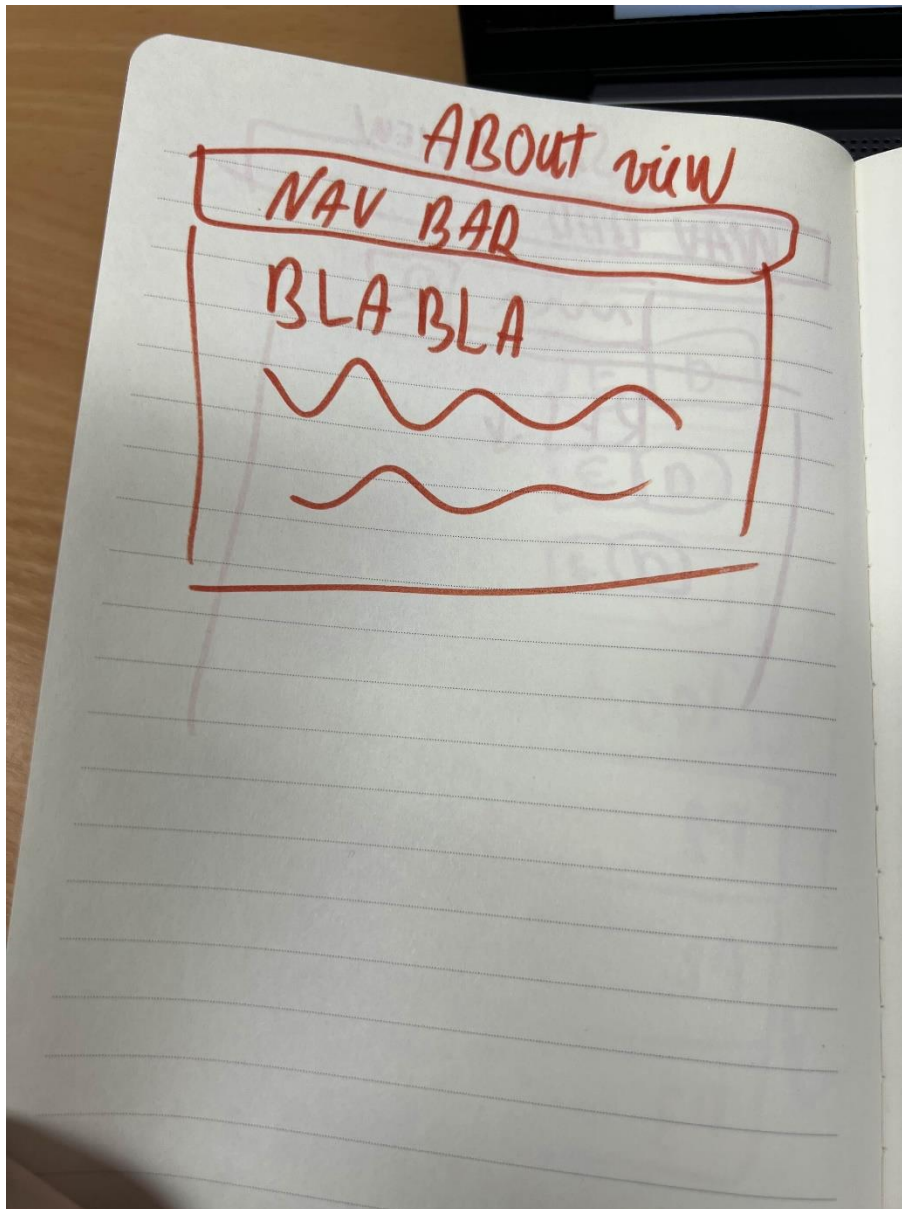


Figure 5 About view wireframe

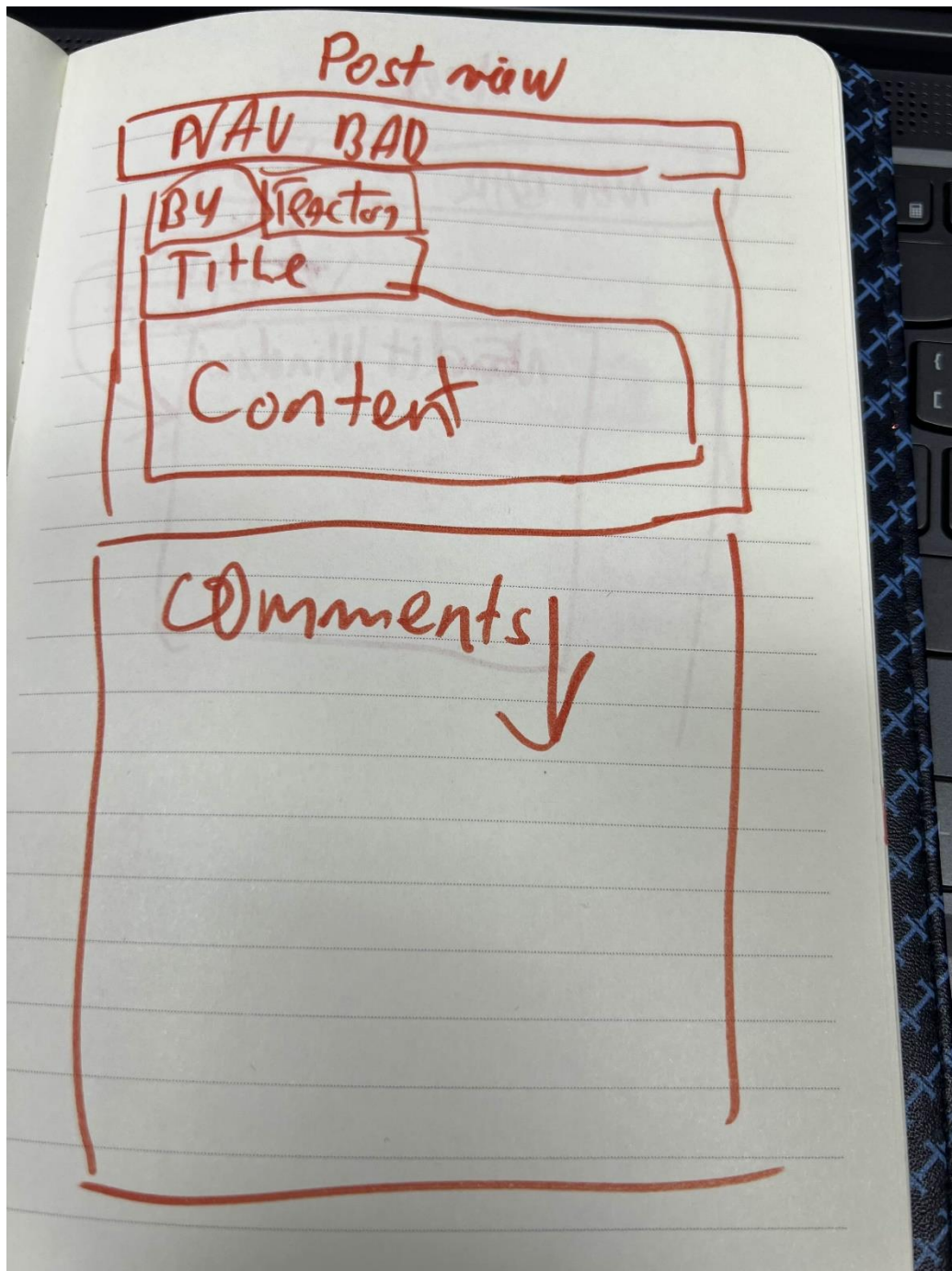


Figure 6 Post View wireframe

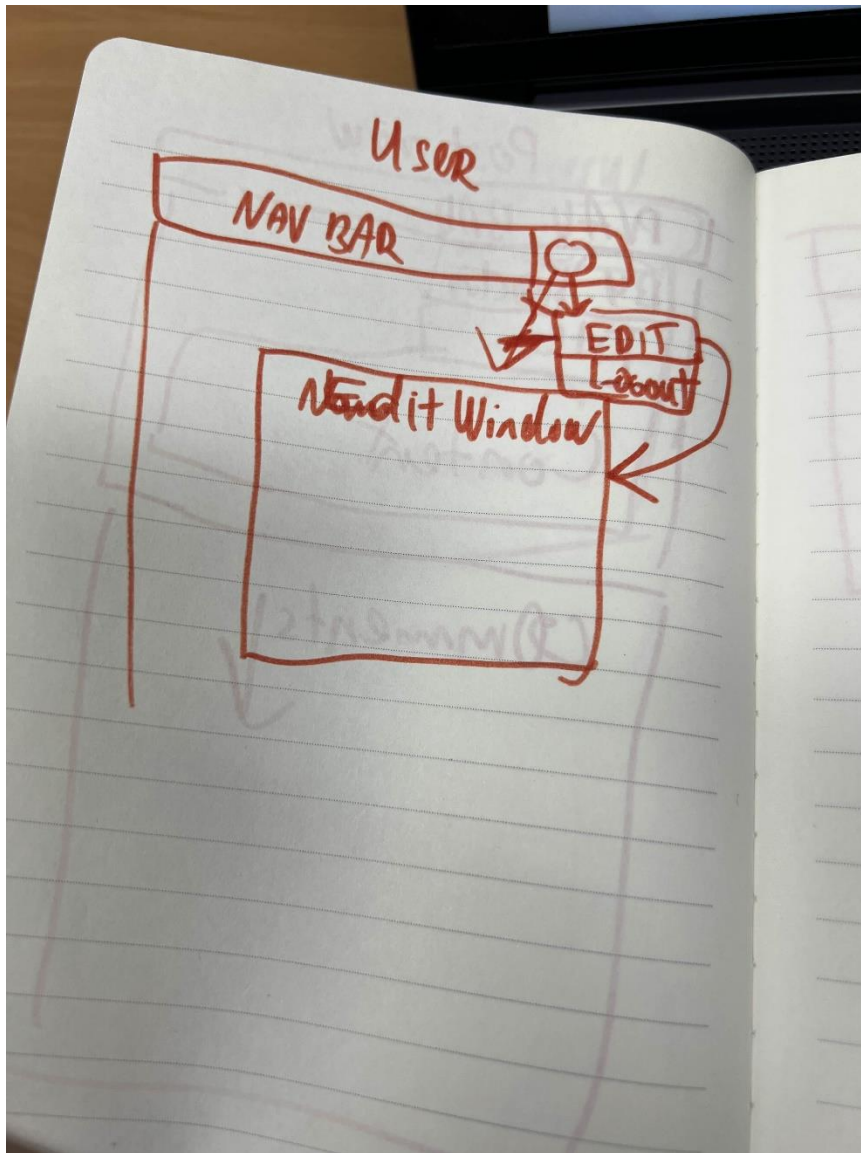


Figure 7 User edit modal wireframe

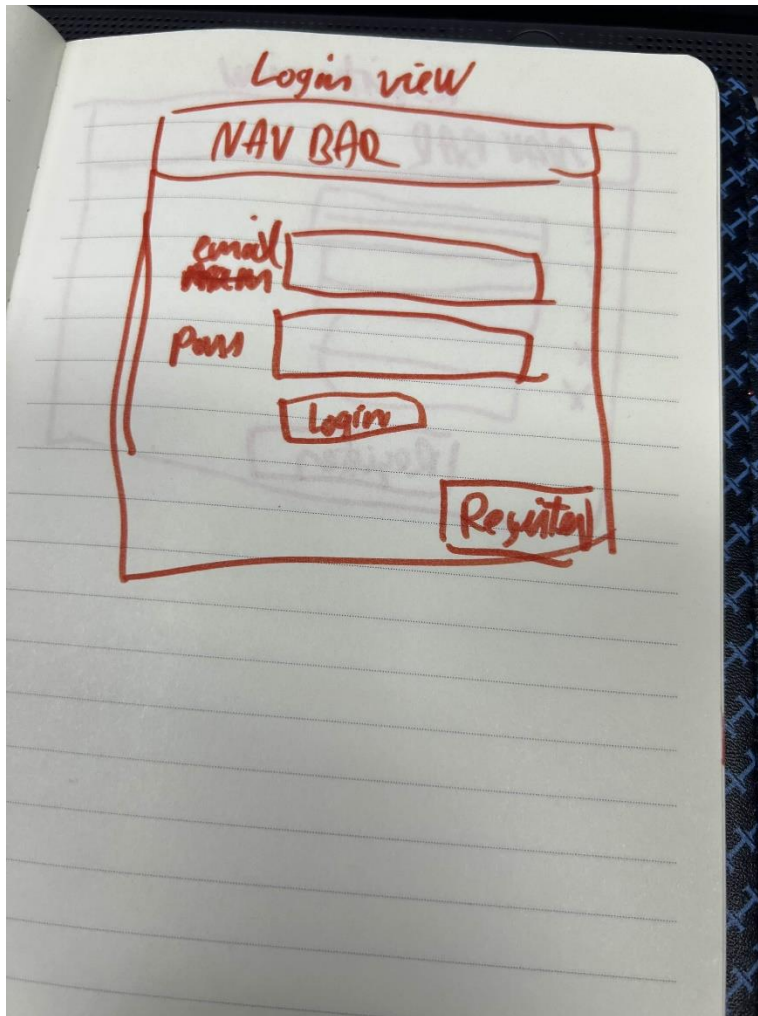


Figure 8 Login wireframe

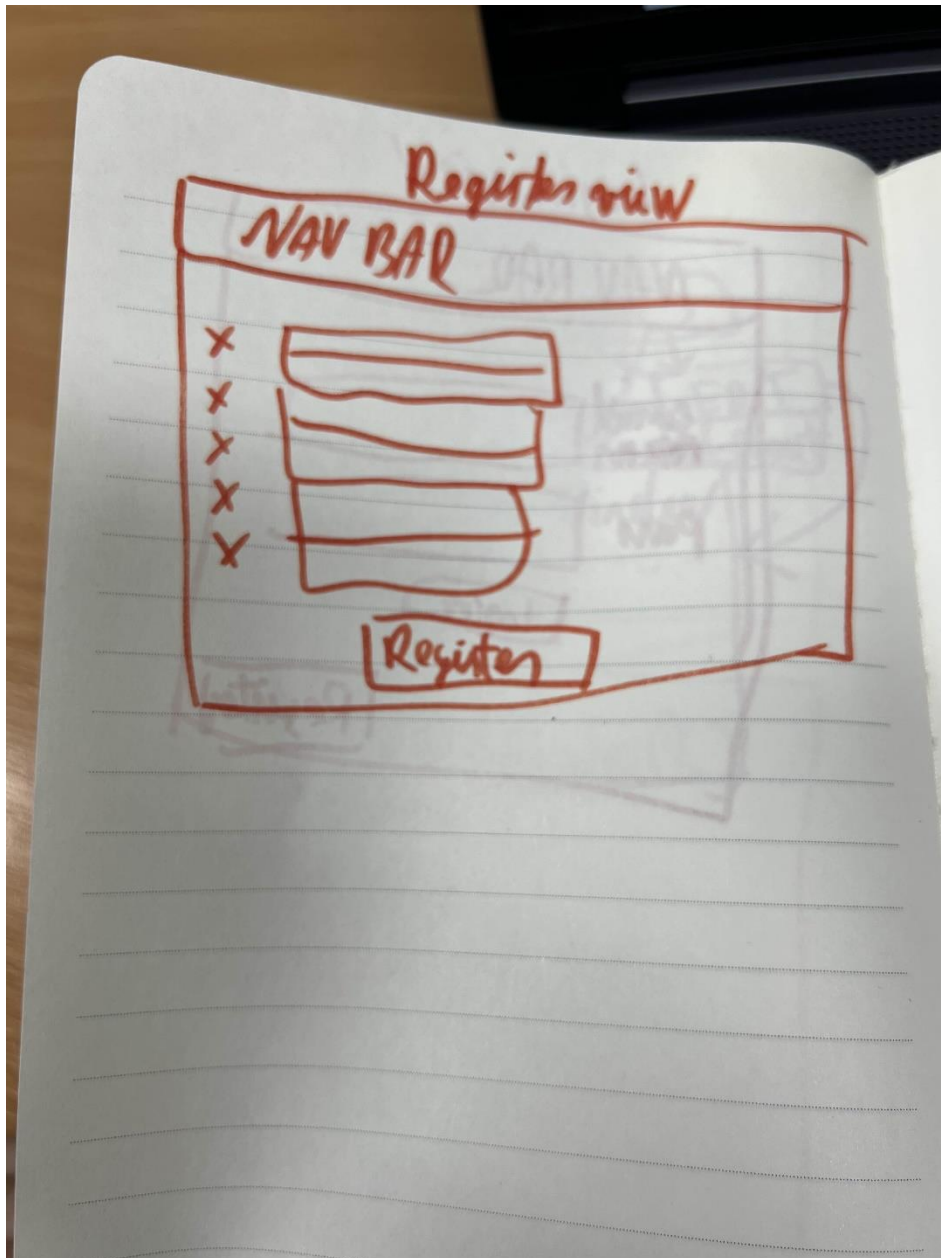


Figure 9 Register wireframe

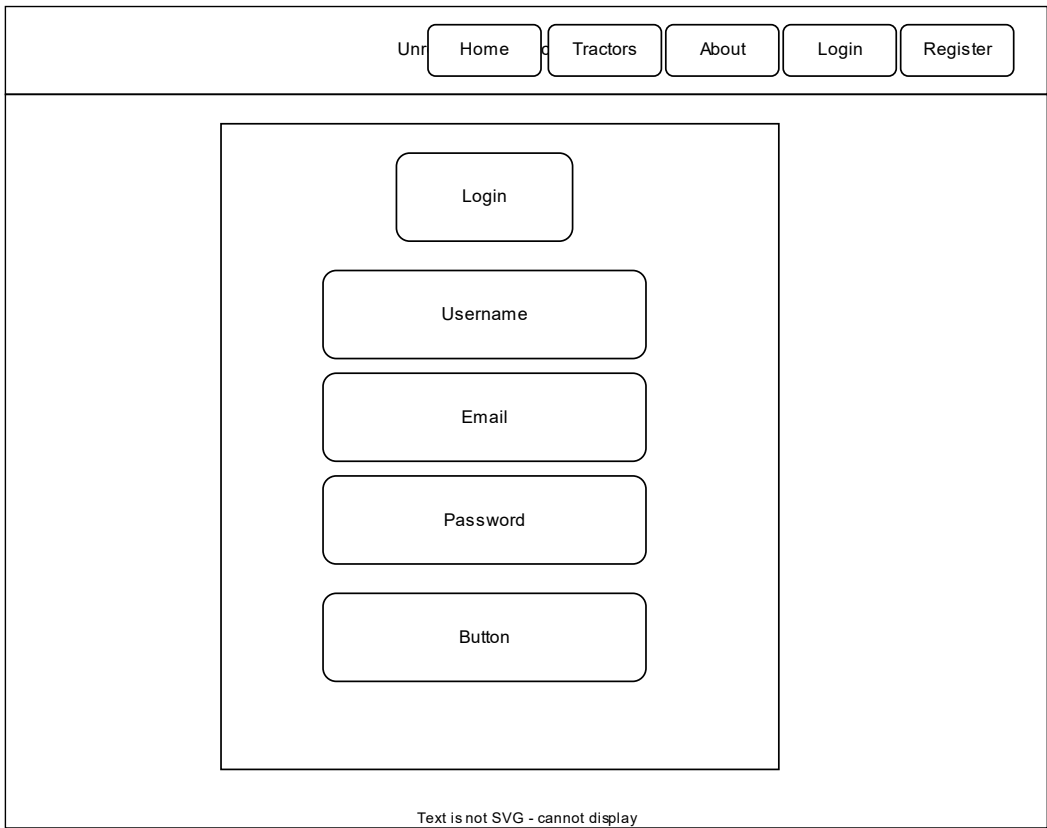


Figure 10 Login wireframe

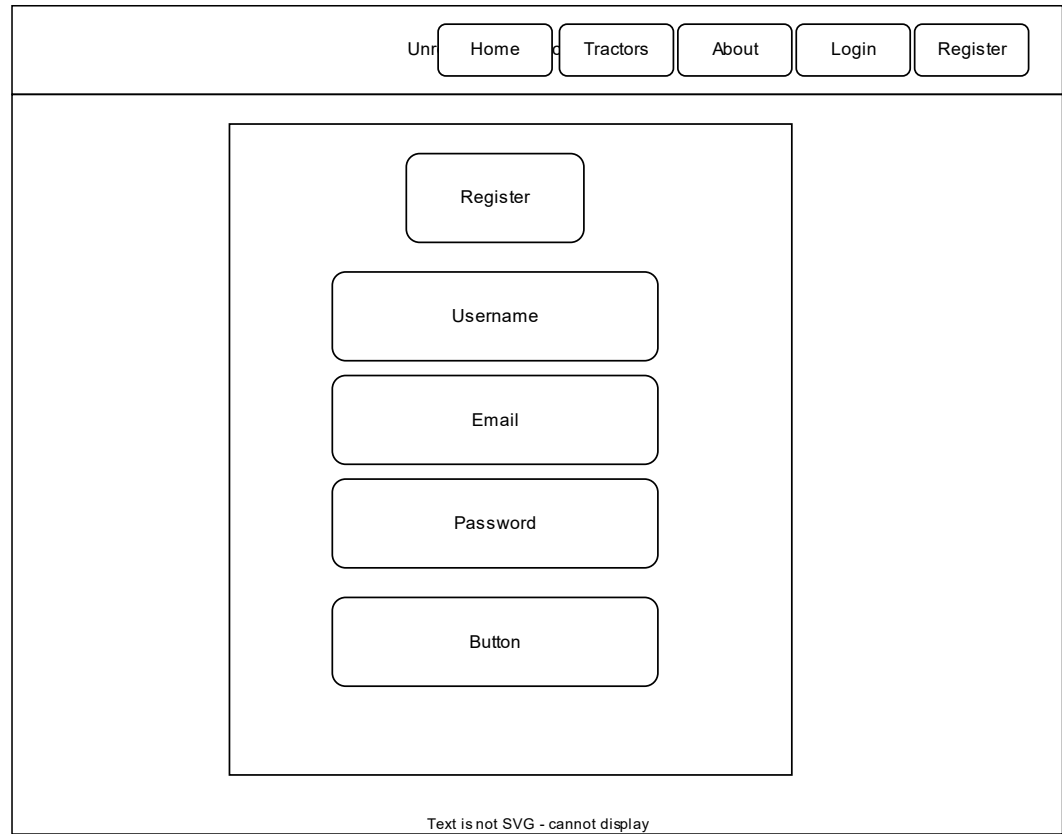


Figure 11 Register wireframe

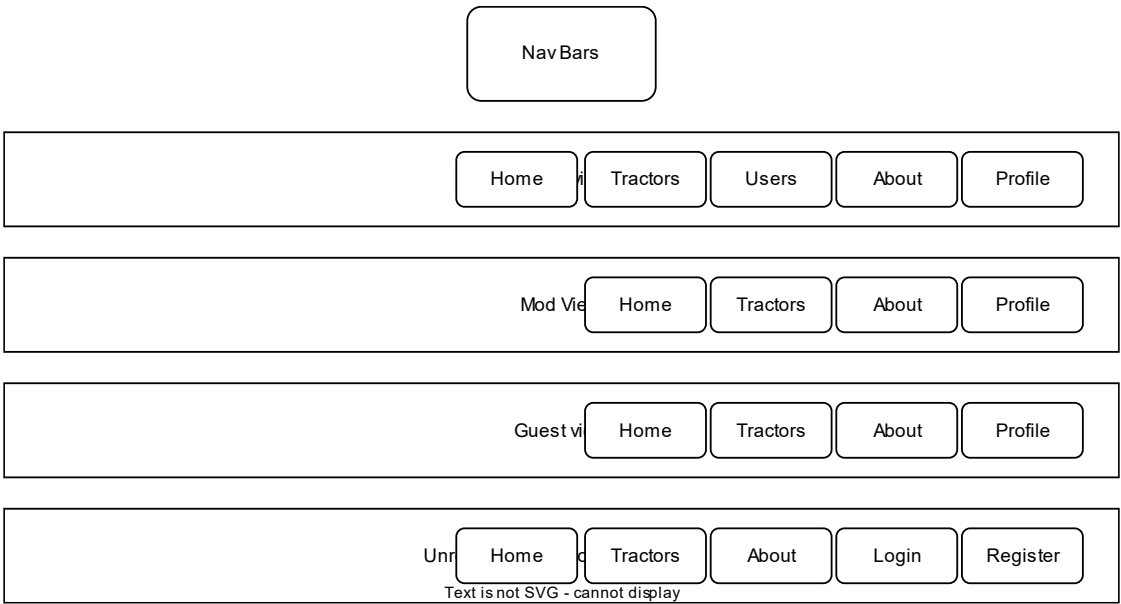


Figure 12 NavBar wireframes

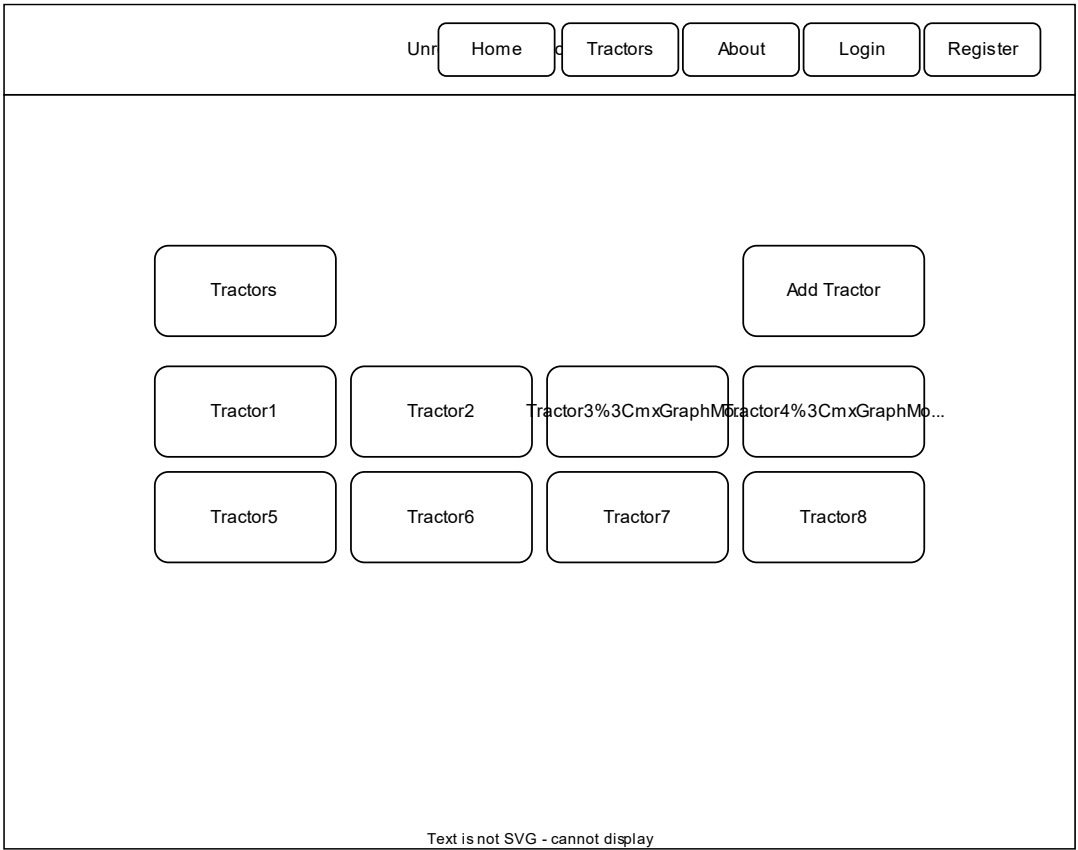


Figure 13 Tractors page wireframe

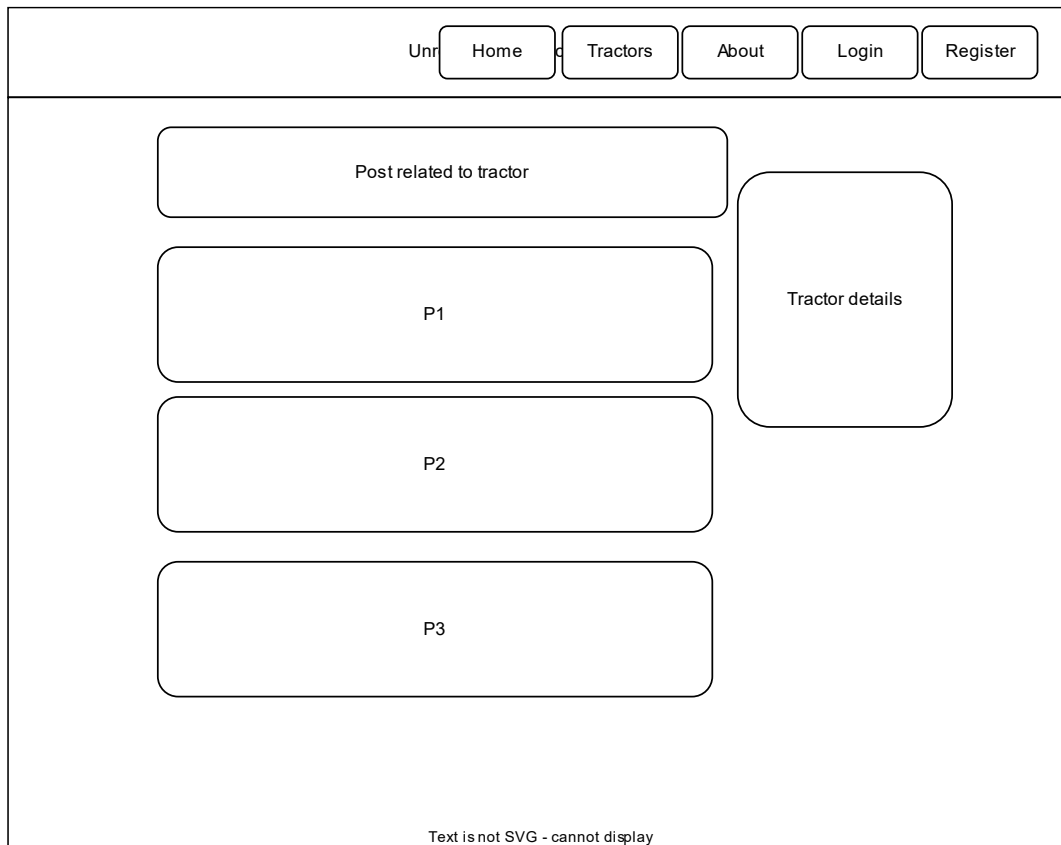


Figure 14 Tractor details view

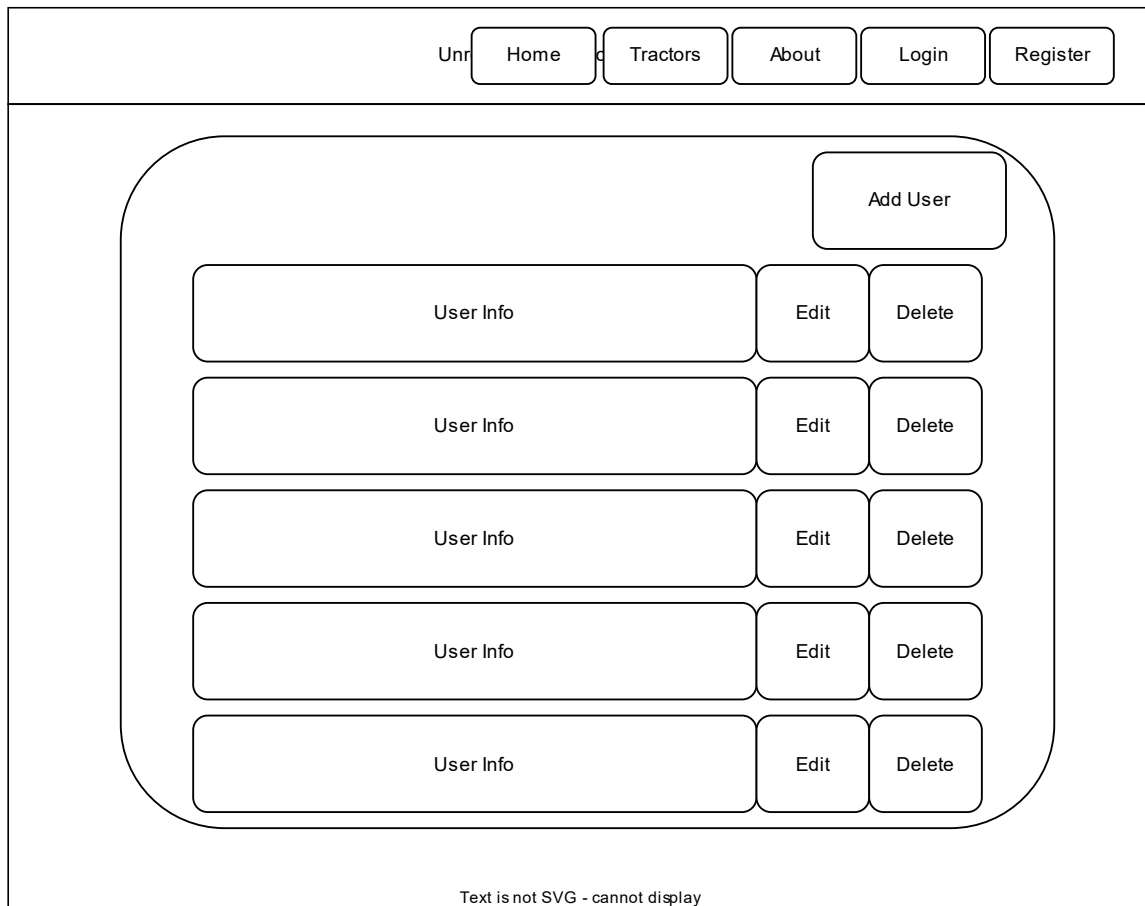


Figure 15 Users view

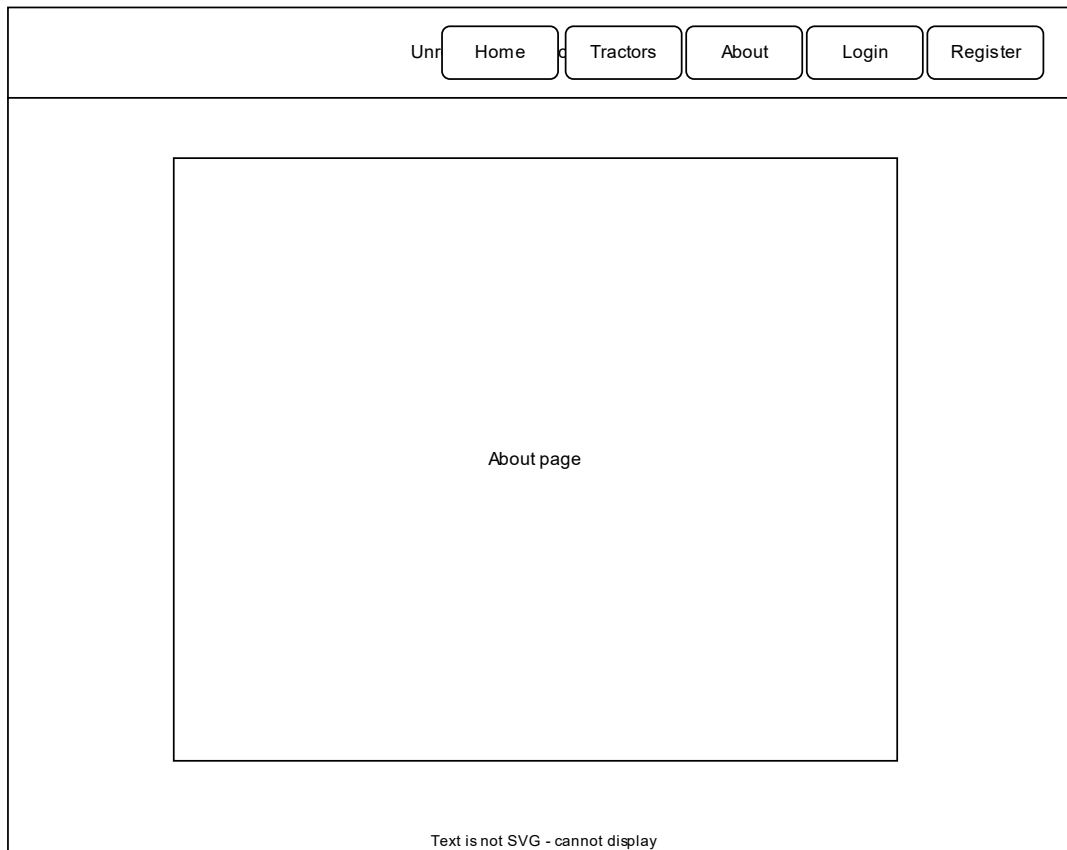


Figure 16 About page wireframe

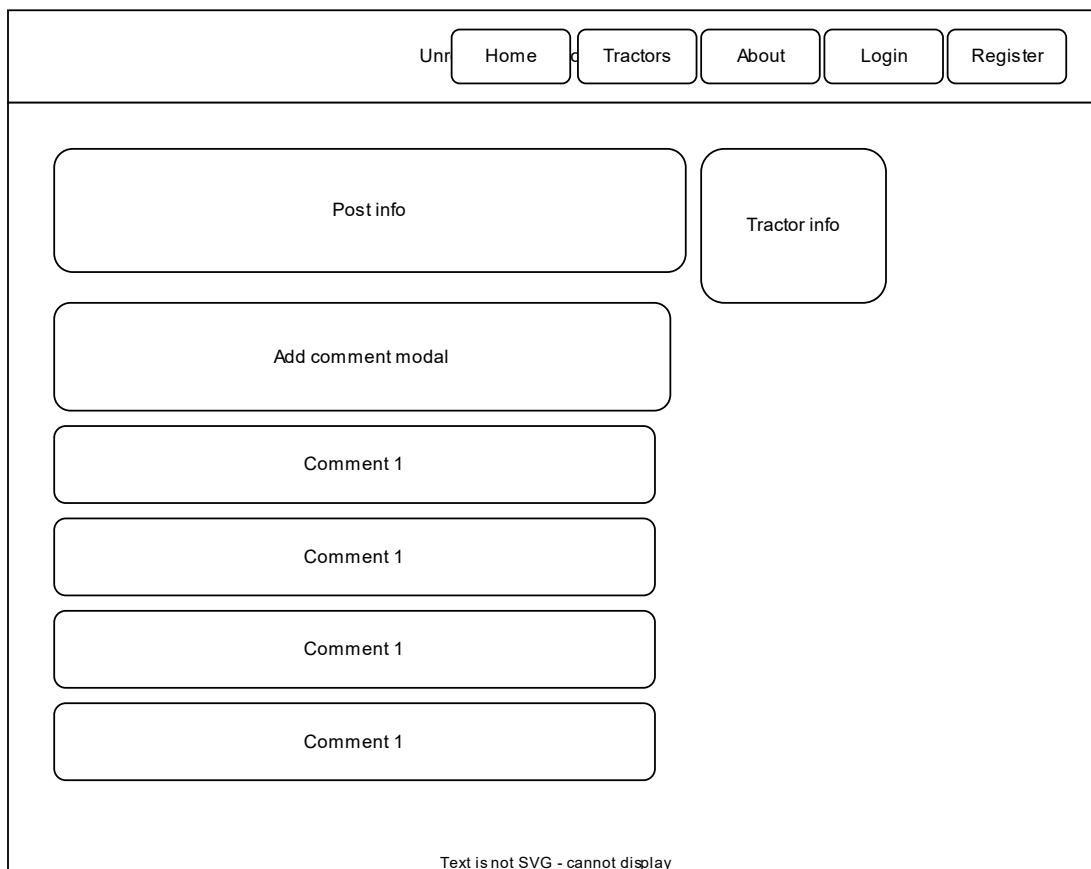


Figure 17 Post details wireframe

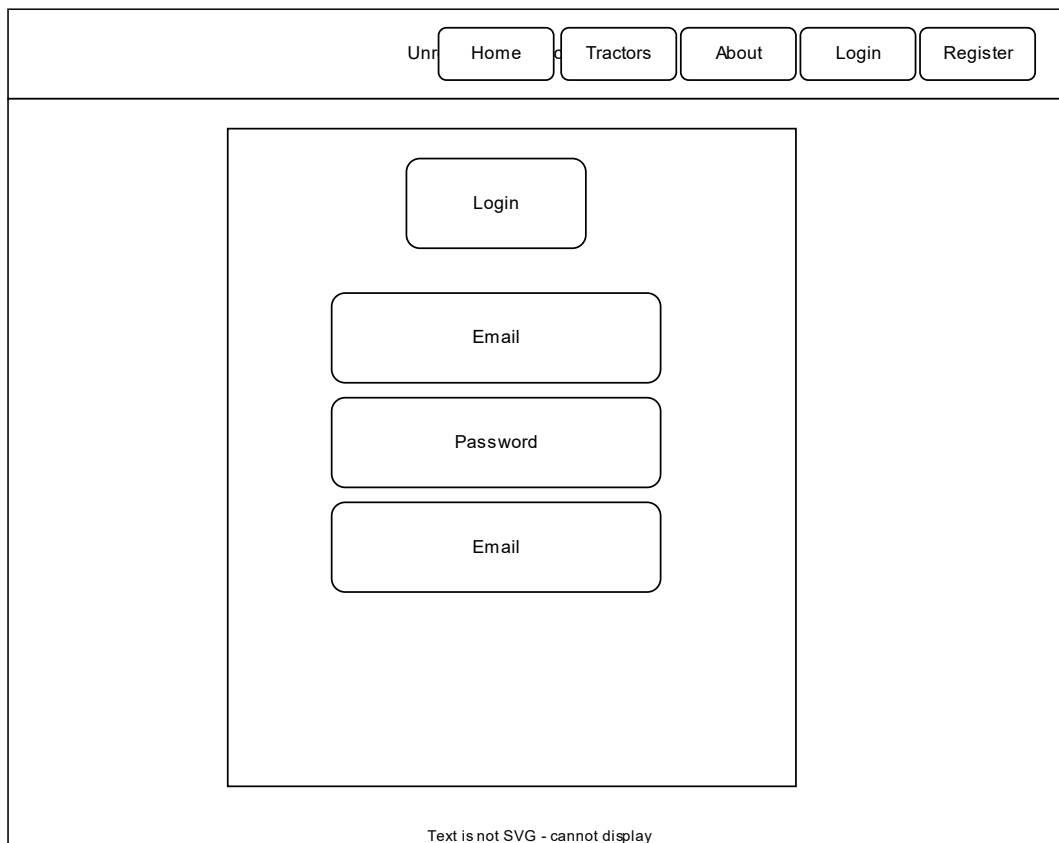


Figure 18 Login wireframe

5. Atliktos užduoties demonstracija ir dokumentacija

Git saugyklos nuorodos:

Visa informacija apie Backend API maršrutus pateikta GitHub saugyklos readme.md faile, kur galėsite rasti išsamią informaciją apie visas API užklausas, metodus ir atsakymus.

Backend API maršrutai: [Backend – readme.md](#)

Visa informacija apie Frontend aplikaciją ir jos struktūrą taip pat pateikta GitHub saugykloje, įskaitant naudojamąs bibliotekas, komponentų struktūrą, autentifikacijos procesą ir daugiau.

Frontend aplikacija: [Frontend – readme.md](#)

Svetainės nuoroda:

[Kletis](#)

6. Išvados

Sukurta Sistema pasižymi greitąveika bei funkcionalumu, kuris patenkina daugumos naudotojų poreikius.

Pagrindiniai darbo rezultatai:

- Užtikrinta ganėtinai patogi naudotojo sąsaja.
- Backend yra lankstus, palaikantis įvairias užklausas ir užtikrinantis duomenų saugumą.
- Pasirinktos technologijos, tokios kaip Vue.js ir Node.js, suteikė sistemai modernumo ir efektyvumo.
- Sistemos architektūra užtikrina gerą komponentų atskyrimą, palengvinant priežiūrą ir plėtrą.

Sistemos tobulinimui ribų nėra – galima tobulinti dizainą, sukurti paieškos langą, failų ir nuotraukų pridėjimą ir forumą, papildomų filtrų, atrankų galimybes, personalizavimo galimybes naudotojams.