**JELGAVAS TEHNIKUMS**

**Niks Barščausks**

**Laikapstākļu datu ievākšana un vizualizācija lietotājsaskarsmē.  
  
  
Kvalifikācijas darbs ieguvei kvalifikācijā -  
programmēšanas tehniķis**

## Darba vadītājs:

## Uldis Grunde-Zeiferts \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Darba izpildītājs:

## 410.gr.izgl. Niks Barščausks \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Jelgava  
2022**

**ANOTĀCIJA**

Niks Barčausks – kvalifikācijas darbs.

Jelgava 2022

Šīs kvalifikācijas darba mērķis ir izveidot majas lapu kas uzkrās un vizualizēs laikapstākļu datus kas tiks ievākti sadarbībā ar Alekša Baloža darbu kur viņs ievāc laikapstakļu datus izmantojot mikrokontrolierus un nosūtīs datus uz manu serveri izmantojo RestAPI JSON formātā. Tiks pielietots ReactJS priekš lietotāj interfeisa un ExpressJS priekš servera interfeisa. Kā arī tiks izmantots MongoDB Cloud priekš datu uzkrāšanas JSON formātā. ReactJS sazināsies ar ExpressJS caur RestAPI.

**SATURS**

Ievads

1.Programmatūras prasību specifikācija

1.1 Ievads

1.1.1 Dokumentācijas mērķis

1.1.2 Produkta sfēra

1.1.3 Dokumentācijas pārskats

1.2 Vispārīgs apraksts

1.2.1 Produkta funkcijas

1.2.2 Produkta pielietojums

1.2.3. Mērķauditorija

1.3 Funkcionālās prasības

2. Izstrādei izmantotās tehnoloģijas un programmatūra

3. Produkta komponenšu apraksts

3.1 ReactJS front-end

3.1.1 ReactJS komponešu saraksts

3.2 Datubāzes struktūra

3.3 ExpressJS back-end

3.3.1 ExpressJS API saraksts

3.3.2 ExpressJS funkcijas

5. Nobeigums

5.1 Secinājumi