**JELGAVAS TEHNIKUMS**

**Aleks Markuss Balodis**

**Niks Barčausks**

**Laikapstākļu datu ievākšana un vizualizācija izmantojot mikrokontrolierus.  
  
  
Kvalifikācijas darbs  
kvalifikācijas ieguvei  
programmēšanas tehniķis**

**Darba izpildītāji:**

**410.gr.izgl. Aleks Markuss Balodis \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**410.gr.izgl. Niks Barčausks \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Jelgava  
2022**

**ANOTĀCIJA**

Aleks Markuss Balodis un Niks Barčausks – kvalifikācijas darbs.

Jelgava 2022

Mikrokontrolieri plaši tiek izmantoti mūsdienās dažādās ierīcēs, lai veiktu vienkāršas funkcijas, kur nav nepieciešama liela veikstpēja. Darba mērķis ir ievākt datus par laikapstākļiem ar mikrokontrolieriem un izmantojot bezvadu komunikāciju nosūtīt tos uz serveri, kas tos saglabās datubāzē un attēlos lietotājam viegli pārredzamā veidā mājaslapā.

Mikrokontrolieri paši nespēj izmerīt temperatūru, gaismas daudzumu vai nosūtīt informāciju bez vadiem, bet vien apstrādāt loģiku, tāpēc tiek pievienoti dažādi moduļi vai komponentes. Komponentes kā LED tiek pievienotas izmantojot GPIO kontaktus, bet moduļi kā, piemēram nRF24L01 bezvadu modulis, izmanto komunikācijas protokolu SPI.

Darba veikšanai nepieciešams būs izmantot dažādas programmēšanas un iezīmju valodas kā C, JavaScript un HTML, CSS. Koda rediģēšanai tiks izmantots Visual Studio Code un Arduino IDE. Versiju kontroles sistēmai tiks izmantots GIT.

**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 Mikrokontrolieri

3.1.1 Mikrokontroliera koda funkcijas

3.2 Datubāze

3.3 WEB

5. Nobeigums

5.1 Secinājumi