

Ovládanie teplovzdušných dúchadiel

Návrh implementácie

Karin Kubinová, Marek Dinka, Tomáš Hruškovíc, Ivan Böhman

Obsah

1. Úvod
 - 1.1. Účel dokumentu
 - 1.2. Rozsah využitia systému
 - 1.3. Prehľad nasledujúcich kapitol
2. Podrobná špecifikácia vonkajších interfejsov
 - 2.1. Komunikácia s inými súbormi
 - 2.2. Komunikácia s inými zariadeniami
3. Dátový model
4. Používateľské rozhranie
5. UML diagramy
 - 5.1. UML component diagram
 - 5.2. UML class diagram
 - 5.3. UML sequence diagram
6. Plán implementácie

1. Úvod

1.1. Účel dokumentu

Tento dokument vznikol v rámci predmetu Tvorba informačných systémov v školskom roku 2022/2023 a slúži ako kompletný a detailný návrh systému na ovládanie teplovzdušných dúchadiel. Obsahuje všetky informácie potrebné pre vysvetlenie a pochopenie funkcionality ako aj spôsobu implementácie systému. Tento dokument je primárne určený pre vývojárov. Obsah v tomto dokumente zahŕňa všetky požiadavky z katalógu požiadaviek.

1.2. Rozsah využitia systému

Pre prácu s týmto dokumentom sa vyžaduje oboznámenie sa s katalógom požiadaviek, ktorý s návrhom úzko súvisí. Tento dokument špecifikuje všetky požiadavky z katalógu požiadaviek. Ďalej špecifikuje vonkajšie interfejsy, formáty súborov a komunikačné protokoly. Obsahuje taktiež celkový návrh používateľského prostredia vrátane vizualizácie. Nachádzajú sa tu aj diagramy, ktoré bližšie popisujú implementáciu systému.

1.3. Prehľad nasledujúcich kapitol

Nasledujúce kapitoly sa venujú špecifikácii vonkajších interfejsov, dátovému modelu perzistentných údajov, používateľskému rozhraniu a jeho vizualizácií, návrhu a plánu implementácie.

2. Podrobná špecifikácia vonkajších interfejsov

2.1. Komunikácia s inými súbormi

Naša aplikácia načítava .xml súbor nahratý používateľom pomocou grafického rozhrania. Následne vytvorí kópiu na rovnaké miesto kde je aj originál, pričom sa k názvu pripíše `_temp_control` a vykonajú sa nasledujúce zmeny.

Na začiatok pripíše nový blok, ktorý bude obsahovať .exe súbor s jediným parametrom, cestou k .xml, `.../path/to/xml.xml`.

<ACTIONS>

```
<COM APP="F:\burniee\EXECPP.exe" CMD="RUNAPP"
PARAMS="'F:\burniee\Program.xml'"/>
```

</ACTIONS>

Ďalej ku každému bloku / fáze sa pridá blok na spúšťanie rovnakého .exe s jediným parametrom, cestou k .xml, `.../path/to/xml.xml`, s tým rozdielom, že sa pred cestu pridá špeciálny znak `^`. Pričom použité cesty k .xml musia byť celé, nestačia relatívne.

<ACTIONS>

```
<COM APP="F:\burniee\EXECPP.exe" CMD="RUNAPP"
PARAMS="'^F:\burniee\Program.xml'"/>
```

</ACTIONS>

Každé meno fázy doplníme o teplotu a ID dúchadiel, ktoré užívateľ zvolí v GUI pri uploadovaní .xml, v tvare @temperature#id1#id2...

Ak

- a) názvy blokov v XML už obsahujú @ . . . doplnia sa iba o zvolené ID dúchadiel (ak boli nejaké zvolené)
- b) neobsahujú @ . . . doplnia sa o template @temperature a následne ID dúchadiel. V tomto prípade treba všetky výskyty temperature prepísať na želanú teplotu.

Toto ale neplatí pre bloky v Measurement, kde sa upraví iba samotný názov Measurement, pričom nastavená teplota a dúchadlá platia pre všetky bloky v ňom.

NAME="Start@temperature#id01"

Aplikácia vytvorí dva súbory na účely logovania.

Prvý, typu .csv, kam sa budú počas behu aplikácie zapisovať logy teploty. Tento bude mať formát:

čas, meno_fázy, id_dúchadla, aktuálna_teplota, cieľová_teplota
pričom, ak je dúchadiel viac, posledné tri stĺpce sa zopakujú pre ďalšie dúchadlo.

2023-01-26 01:46:51.71, Start, id02, 26.75, 70, id03, 37.0, 90

Tento súbor bude jedinečný pre každý spustený projekt a bude sa ukladať do priečinku temperature_logs, ktorý vytvoríme na mieste, kde sa nachádza práve spúšťané .xml.

Druhý súbor log.txt kam sa budú zapisovať relevantné udalosti, ako napr. bežné požiadavky nastavenia teploty dúchadla interpretovaním skriptu, prechod na ďalšiu fázu, prípadne akékoľvek neočakávané udalosti – rôzne výnimky, ručné riadenie teploty, alebo zastavenie projektu. A tento bude formátu:

čas

správa

Napr. 2023-01-28 19:39:19.772

Search for new blowers was successful

A ten sa bude ukladať na miesto, odkiaľ je práve spúšťaný server, resp. GUI. V prípade, že tento súbor presiahne veľkosť 1000kB, tak (pri štarte servera) sa premenuje na logOld.txt a nové sa logujú do log.txt. V prípade, že logOld.txt už existuje, tak sa zmaže.

Používateľ bude môcť uložiť nastavenia GUI, konkrétne cestu k .exe, ktoré sa bude počas testovania spúšťať, a port, na komunikáciu so serverom. Toto nastavenie sa uloží tam, kde je uložené GUI, do jedného GUIconfig.txt súboru formátu:

cesta k .exe

Port

2.2. Komunikácia s inými zariadeniami

S dúchadlom komunikuje náš systém, bežiaci na mikrokontrolére, pomocou DAC prevodníku MCP4725. Mikrokontrolér číta aktuálnu teplotu prostredníctvom pripojeného termočlánku pripojeného na čip MAX6675.

3. Dátový model

Medzi GUI a serverom prebieha obojstranná komunikácia pomocou TCP. Medzi serverom a mikroprocesorom prebieha UDP komunikácia.

4. Používateľské rozhranie

Táto časť obsahuje návrh používateľského rozhrania. Detailný popis funkcií je rozpísaný v katalógu požiadaviek.

1. BLOWERS

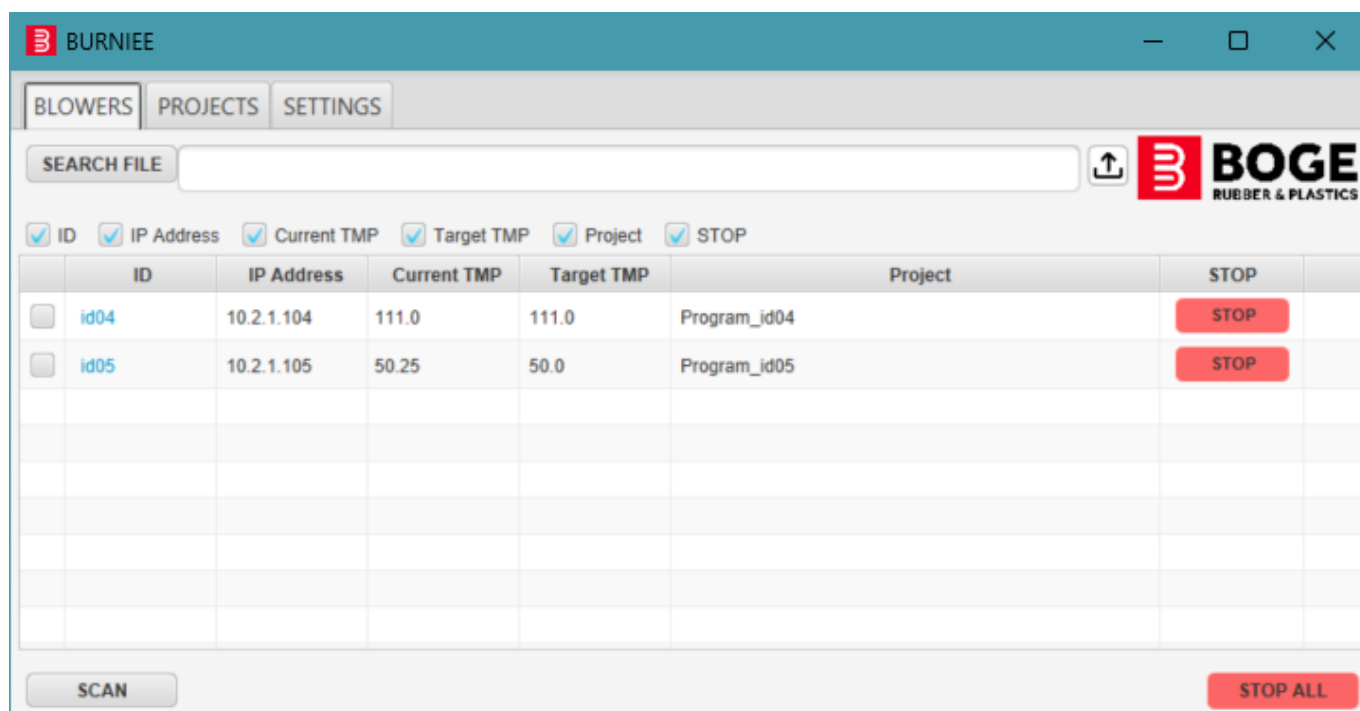
Tu sa zobrazujú aktuálne pripojené dúchadlá.

Kliknutím na ID sa dostaneme na webové rozhranie controllera, kde vieme nastaviť jeho údaje = ID, IP adresu, PID... či zapnúť manuálne riadenie dúchadla.

Kliknutím na STOP pri dúchadle (a potvrdením v potvrdzovacom okne) zastavíme ohrev dúchadla. Teplota teda začne klesať, avšak dúchadlo stále vníma jeho cieľovú teplotu.

Kliknutím na ⚠ dúchadlo znova začne hriať na cieľovú teplotu.

Kliknutím na STOP ALL (emergency button) všetky dúchadlá ihneď prestanú hriať a testovacie projekty sa zrušia.

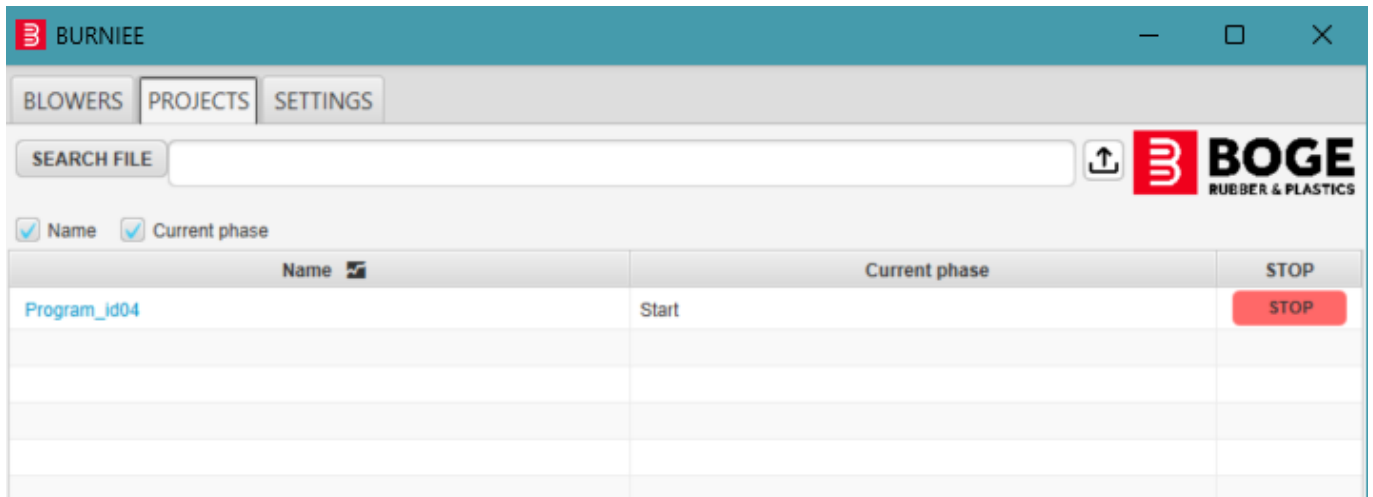


2. PROJECTS

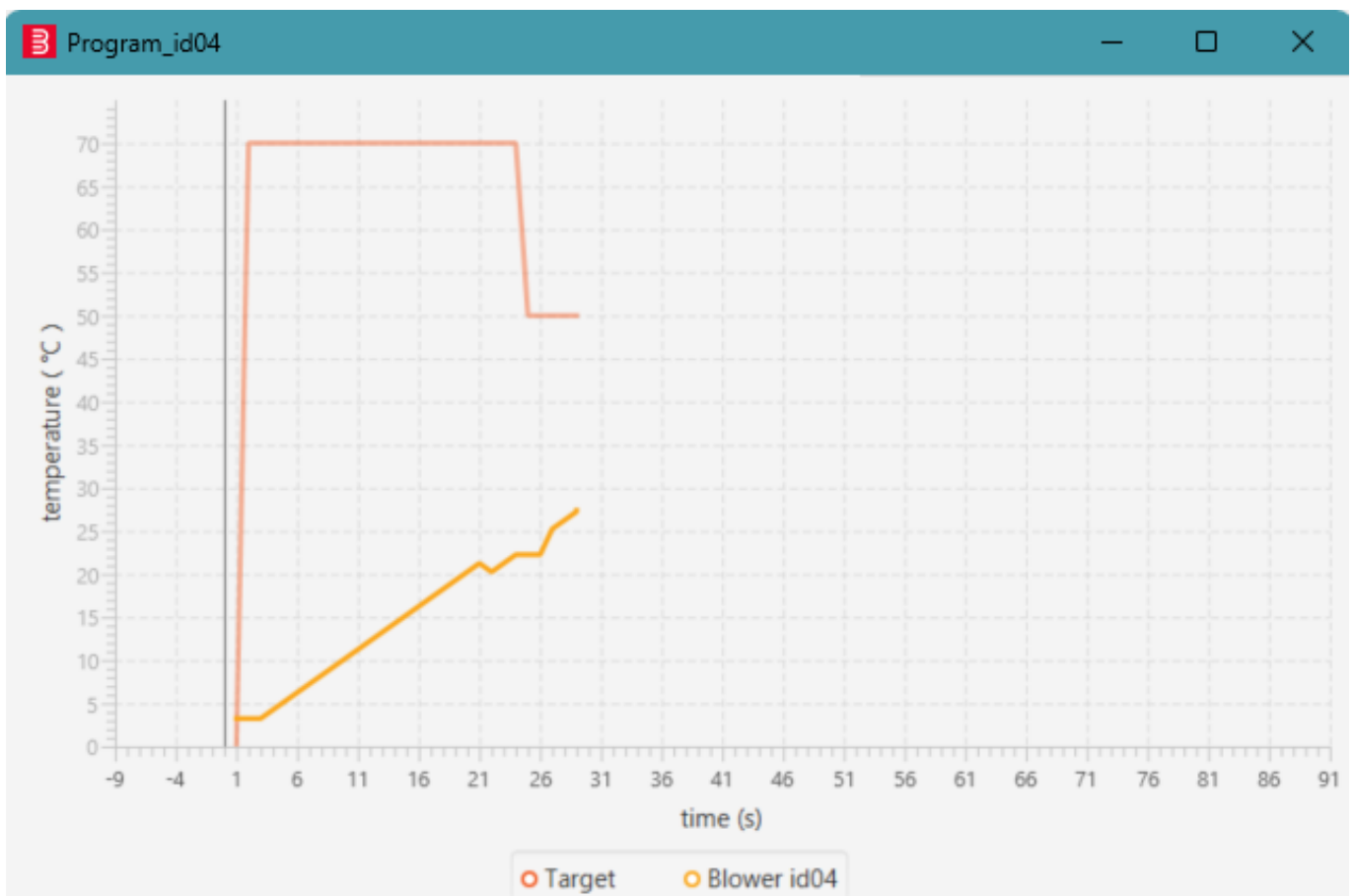
Tu sa zobrazujú aktuálne bežiace projekty.

Kliknutím na **NAME** sa zobrazí nové okno s grafom projektu, kde sa zobrazuje celý priebeh testovania, teda teplota jednotlivých dúchadiel a aj ich požadovaná teplota. Graf sa posúva automaticky, ale dá sa tu posúvať scrollovaním myškou.

Kliknutím na **STOP** sa testovanie zastaví a všetky dúchadlá k nemu priradené prestanú ohrievať.



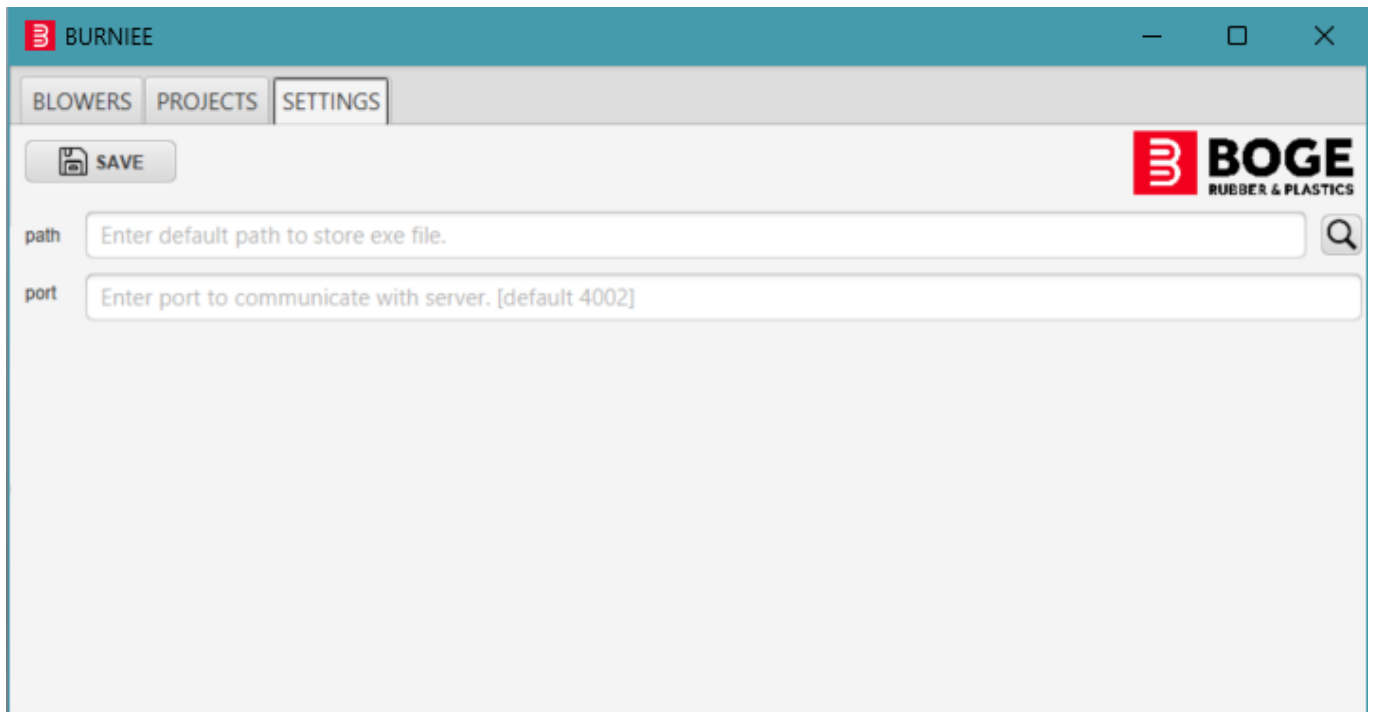
Name	Current phase	STOP
Program_id04	Start	STOP



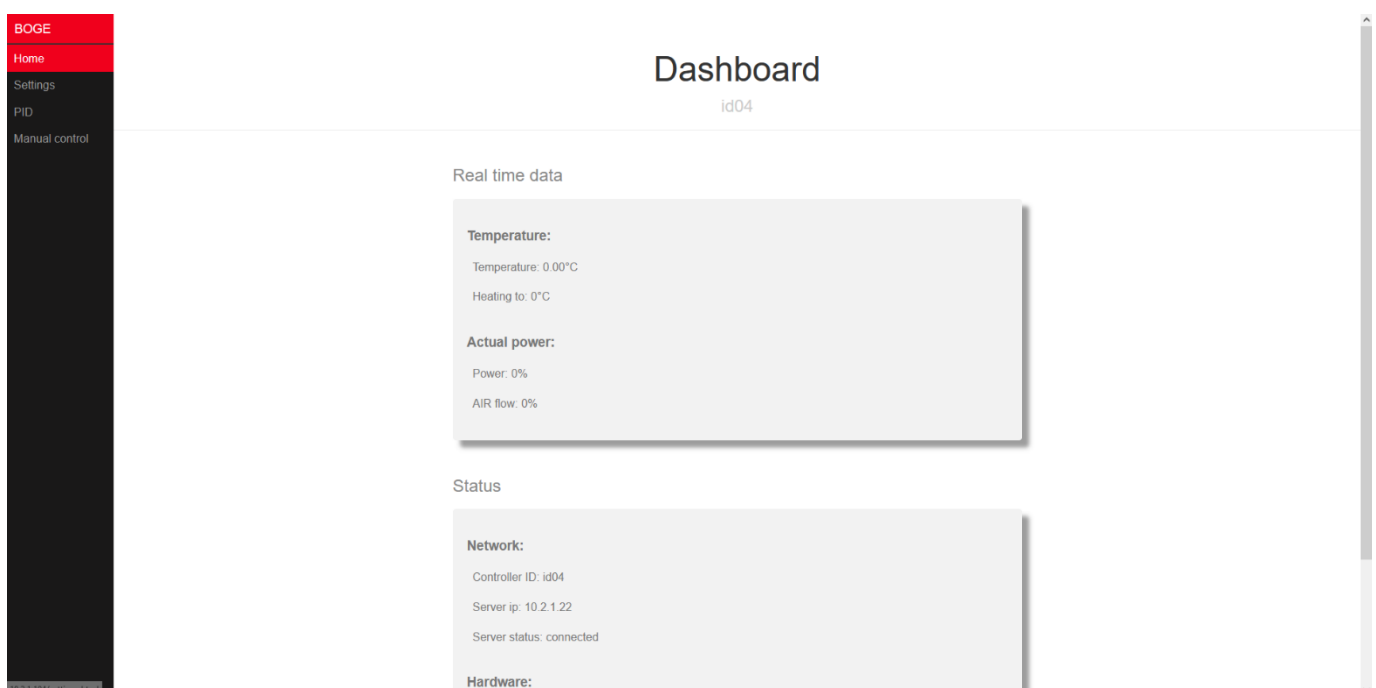
3. SETTINGS

Tu vieme nastaviť cestu k EXE (kliknutím na lupu 🔍 sa otvorí prieskumník súborov) a port na komunikáciu so serverom (defaultne nastavený na 4002).

Kliknutím na **SAVE** sa nastavenia uložia a pri ďalšom spustení GUI sa načítajú.



Webové rozhranie controllera vyzerá nasledovne.



Settings

id04

Controller setup

Controller IP:

10.2.1.104

Controller Gateway:

10.2.1.1

Controller Network mask:

255.0.0.0

Controller ID:

id04

Save

PID setup

P:

0.04

I:

0.0006

D:

0.005

Alpha:

1

Delay (ms):

250

Δt :

0

Save

Control

id04

Manual control

Manual control



PID control

Temperature:

80

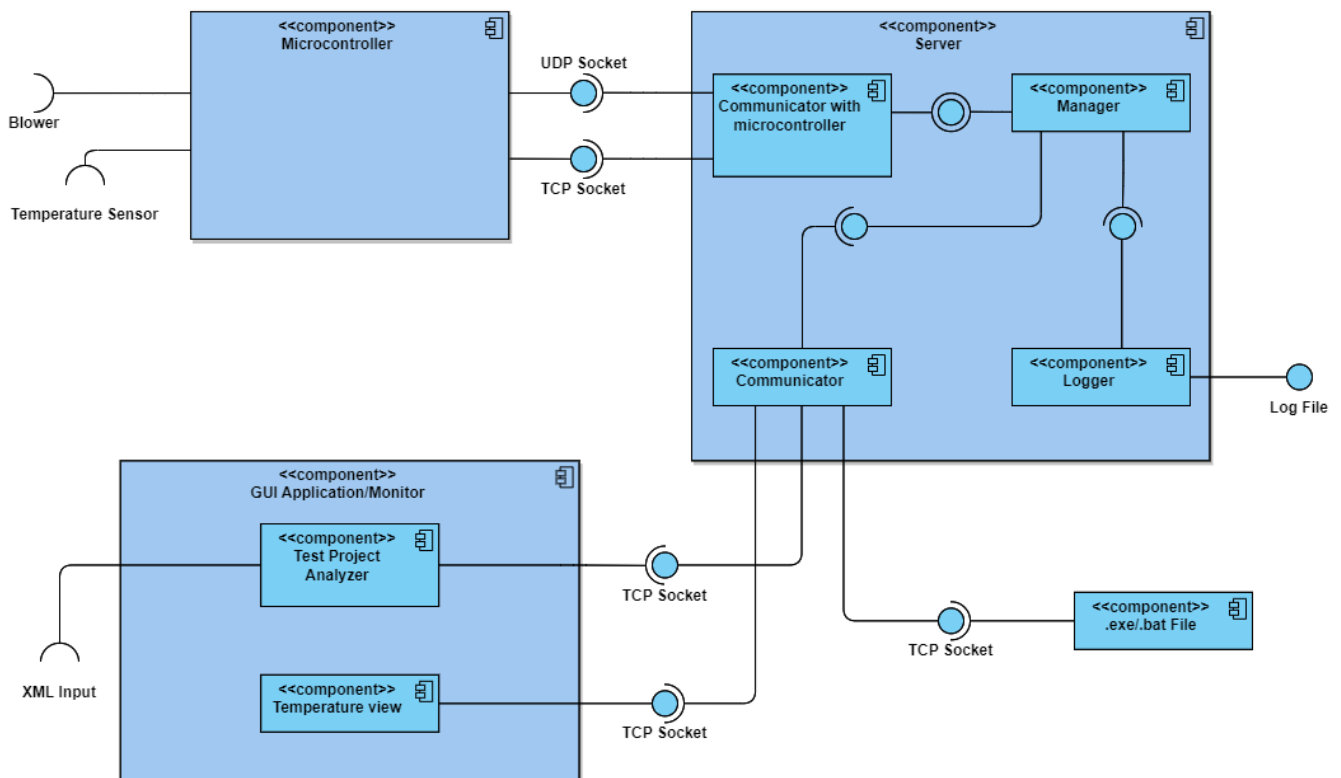
AirFlow:

100

Maintains temperature

5. UML diagramy

5.1. UML component diagram



5.2. UML class diagram

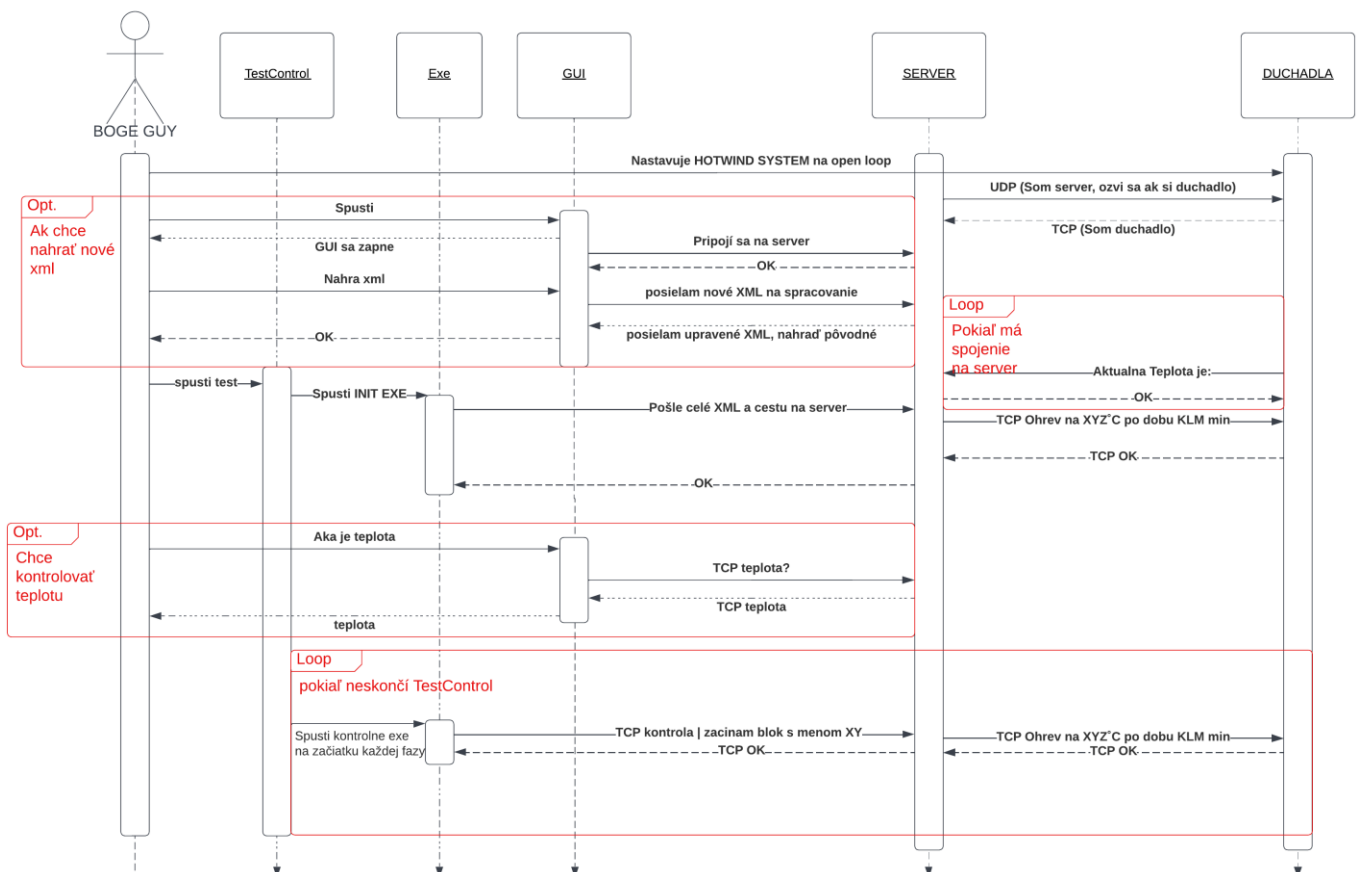
GUI



Server



5.3. UML sequence diagram



6. Plán implementácie

Táto časť obsahuje postupný plán implementácie projektu.

- Vytvoriť komunikačné komponenty
 - Komunikácia medzi arduinom a dúchadlom pomocou DAC kontroleru
 - Komunikácia medzi arduinom a teplomerom
 - Komunikácia medzi arduinom a serverom
 - Komunikácia medzi GUI aplikáciou a serverom
- Testovanie ovládania dúchadla pomocou arduina
- Pripraviť načítavanie vstupného XML súboru (transformácia, extrakcia fáz)
- Vytvorenie riadiaceho komponentu (manager) servera
- Vytvoriť EXE súbor, ktorý pošle správu serveru
- Vytvoriť základ GUI aplikácie bez dôrazu na dizajn, ktorý umožní nahrať XML súbor
- Vytváranie logovacieho komponentu servera
- Úprava vizuálu GUI aplikácie do finálnej podoby
- Testovanie aplikácie a validácia podľa katalógu požiadaviek