

OYSTER MUSHROOM'S GREENHOUSE REMOTE MONITORING SYSTEM BASED ON RASPBERRY PI

Martanto¹⁾, Elisabet Yustia²⁾

Jurusan Teknik Elektro

Fakultas Sains dan Teknologi, Universitas Sanata Dharma
Kampus III, Paingan, Maguwoharjo, Depok, Sleman, Yogyakarta 56281
Email: ¹martanto@usd.ac.id

Abstract

In the agricultural sector there are many mushroom farmers who do manual cultivation. Environmental factors that influence the growth of oyster mushrooms include temperature and humidity. This system was made to maintain the stability of temperature and humidity in oyster mushrooms' greenhouse remotely. Communication of remote data exchange between Raspberry-Pi and Zelio using TCP/IP networks was successfully carried out. The system managed to run the features in the setting menu which is able to set the temperature and humidity set point values, set the on/off condition of the actuators, display the local system set point and set the on/off condition of the enable button in the GUI. Temperature and humidity values, and actuator status can displayed in monitoring menu. In the logging menu, the data was saved in csv format. Automatic file naming based on date was successfully done, the csv file was successfully opened and displayed in graphical plot.

Keyword: Setting, Remote Monitoring, Logging, GUI, Raspberry Pi

1. Pendahuluan

Teknologi turut berperan penting dalam perkembangan peradaban manusia. Teknologi adalah keseluruhan sarana untuk menyediakan barang-barang yang diperlukan bagi kelangsungan dan kenyamanan hidup manusia[1]. Manusia menciptakan dan mengembangkan teknologi untuk membuat kehidupan manusia menjadi lebih baik.

Dengan kehadiran teknologi yang semakin modern, manusia bisa bertukar informasi dalam hitungan detik. Bandingkan dengan media yang diperlukan untuk bertukar informasi seperti surat, yang bisa memakan waktu berhari-hari, bahkan berminggu-minggu untuk bisa diterima. Hal ini menunjukkan bahwa semakin berkembang teknologi, maka hidup manusia semakin mendapat kemudahan.

Dengan semakin tingginya laju mobilitas manusia, maka perlu diimbangi dengan teknologi yang bisa mengiringinya. Teknologi jarak jauh menjadi salah satu jawabannya, yaitu teknologi yang memungkinkan manusia untuk selalu terhubung dengan yang lainnya, kapanpun dan dimanapun berada. Teknologi jarak jauh ini telah diterapkan dalam berbagai bidang, yaitu baik dalam bidang pertahanan, keamanan, ekonomi, industri, maupun pertanian.

Dalam bidang pertanian secara khususnya, teknologi ini telah dipakai untuk memonitor dan mengontrol irigasi, pengendalian proses manufaktur pupuk, serta pengendalian dan pemantauan sistem dalam manufaktur tepung [2]. Bidang pertanian menghasilkan produk-produk yang menjadi konsumsi utama manusia, contohnya padi, kedelai, gandum, jamur. Secara khusus untuk produk jamur, peminatan masyarakat terhadap produk pertanian ini meningkat dari tahun ketahun. Banyaknya orang yang telah terjun ke dalam dunia budidaya jamur ternyata tetap belum bisa mengimbangi permintaan pasar yang tinggi terhadap produk jamur. Hal ini dikarenakan masih banyak petani jamur yang melakukan pembudidayaan jamur dengan pemeliharaan secara manual [3].

Dalam pembudidayaan jamur, khususnya pada jamur tiram, pemeliharannya memerlukan perlakuan yang khusus. Ada beberapa faktor lingkungan yang mempengaruhi tumbuh kembang jamur tiram, diantaranya adalah suhu dan kelembaban. Pemeliharaan yang tidak maksimal terhadap jamur tiram akan menyebabkan jamur tiram tidak bertumbuh dan berkembang dengan maksimal, dan hal ini dapat mengakibatkan turunnya kualitas dan kuantitas produksi jamur. Pemeliharaan dengan cara manual adalah kurang efektif, karena petani harus sering mengecek kondisi suhu dan kelembaban, kemudian melakukan penanganan manual terhadap kondisi yang terjadi tersebut, yaitu menghidupkan aktuator secara manual. Hal ini tentu saja akan membuang waktu, dan tenaga.

Pada penelitian sebelumnya, telah ada penelitian yang menjaga suhu dan kelembaban pada kumbung jamur tiram menggunakan Zelio Smart Relay. Zelio ini kemudian akan mengontrol keluaran aktuator secara otomatis berdasarkan suhu dan kelembaban yang terjadi [4], namun sistem kendali dalam penelitian tersebut masih bersifat lokal, yaitu belum bisa dimonitor dan diatur dari jarak jauh. Sistem lokal ini masih menggunakan tombol-tombol yang harus ditekan dari jarak dekat, dan belum memiliki tampilan yang *user friendly* untuk mengetahui kondisi-kondisi yang sedang terjadi dalam kumbung jamur tiram.

Raspberry Pi adalah suatu teknologi berukuran mini yang memiliki fungsi utama sebagai komputer, namun juga bisa dipakai sebagai mikrokontroler [5]. Di dalam Raspberry Pi ini terdapat layanan-layanan untuk membuat program berbasis grafis, termasuk di dalamnya pembuatan *Graphical User Interface* (GUI). GUI ini kemudian dapat dipakai oleh *user* untuk mengoperasikan sistem dari jarak jauh.

Dari permasalahan di atas, maka dirancanglah pengembangan dari metode sistem kendali lokal Zelio tersebut. Pengembangan ini adalah sebuah sistem berbasis Raspberry Pi 3 yang dapat memonitor dan termasuk di dalamnya dapat melakukan pengaturan terhadap nilai *set point* suhu dan kelembaban serta aktuator dalam kumbung jamur tiram dan dapat melakukan perekaman data. Dengan adanya sistem monitor ini, suhu dan kelembaban serta aktuator dapat dimonitor, direkam, dan diatur dari jarak jauh.

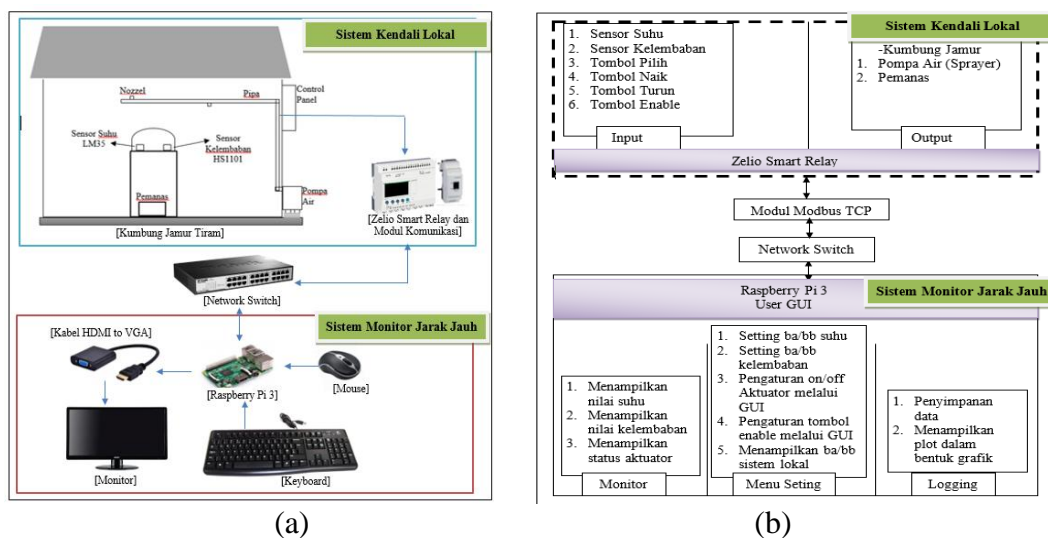
Tujuan dari penelitian ini adalah menciptakan alat untuk memonitor sistem secara jarak jauh pada kumbung jamur tiram dengan menggunakan Raspberry Pi 3. Manfaat yang ingin dicapai dari penelitian ini yaitu memudahkan petani jamur tiram untuk memonitor kondisi suhu, kelembaban dan status aktuator pada kumbung jamur tiram, memudahkan petani jamur untuk melakukan pengaturan terhadap nilai *set point* suhu dan kelembaban, serta aktuator pada kumbung jamur tiram, memudahkan petani jamur untuk melakukan perekaman data dan melihat data dalam plot grafik, mengembangkan penelitian tentang sistem monitor pada kumbung jamur tiram yang sudah ada.

Untuk menghindari kompleksnya permasalahan yang muncul, maka dibatasi sebagai berikut. Sistem kendali lokal kumbung jamur yang digunakan memiliki jangkauan pengukuran batas atas dan batas bawah suhu 0°C - 40°C dan kelembaban 0% - 100% [4]. Prototipe kumbung jamur akan menggunakan prototipe yang sudah ada. Menggunakan Zelio Smart Relay untuk mengendalikan sistem lokal kumbung jamur. Menggunakan Modbus TCP/IP sebagai protokol komunikasi antara Raspberry Pi dengan Zelio. Perancangan GUI untuk memonitor dan melakukan penyetelan (*setting*) berbasis Raspberry Pi 3. Monitoring akan mencakup pemantauan terhadap nilai suhu, nilai kelembaban dan status aktuator. Penyetingan akan mencakup pengaturan nilai suhu batas atas (ba) dan batas bawah (bb) pada suhu dan kelembaban, pengaturan perintah on/off aktuator dari GUI, pembacaan ba dan bb sistem lokal Zelio, serta pengaturan on/off tombol *enable* dari GUI. Perekaman data dapat disimpan dalam bentuk *file* csv dan dapat ditampilkan dalam bentuk plot grafik.

2. Metodologi Penelitian

Metode yang digunakan untuk membuat sistem monitor jarak jauh adalah sebagai berikut. Tahap pertama adalah perancangan sistem yang meliputi perancangan perangkat keras dan perangkat lunak. Tahap kedua adalah mengimplementasikan rancangan sistem untuk membentuk sistem keseluruhan. Tahap ketiga adalah pengujian dan pengambilan data. Tahap terakhir adalah melakukan analisis data dan pengambilan kesimpulan.

Secara umum, *plant* dari sistem monitoring kumbung jamur tiram ditunjukkan dalam gambar gambar 1. Gambar 1(a) menunjukkan *plant* sistem monitoring kumbung jamur secara keseluruhan, terdiri atas sistem kendali lokal berbasis Zelio Smart Relay, dan sistem monitor jarak jauh berbasis Raspberry Pi. Dalam perancangan perangkat keras sistem jarak jauh, Raspberry Pi 3 akan dihubungkan ke monitor untuk menampilkan GUI dengan kabel HDMI to VGA. Port Ethernet dari Raspberry Pi 3 akan dihubungkan ke Switch. Modul Modbus TCP dari Zelio Smart Relay juga akan dihubungkan ke Network Switch untuk komunikasi antara Raspberry Pi 3 dan Zelio. Keyboard dan Mouse akan dihubungkan ke Raspberry Pi 3.



Gambar 1. *Plant* Sistem Monitor pada Kumbung Jamur Tiram

Model sistem gambar 1(b) adalah mengenai sistem kumbung jamur kendali lokal dan sistem monitor jarak jauh. Sistem kumbung jamur kendali lokal yang telah dibuat, kerja sistem sepenuhnya dikendalikan oleh Zelio Smart Relay. *User* dapat mengubah nilai *set point* untuk suhu dan kelembaban dengan menggunakan beberapa tombol yang disediakan, yaitu tombol pilih, tombol naik, tombol turun dan tombol *enable*. Tombol pilih digunakan untuk memilih pengaturan jenis *set point* yaitu suhu atau kelembaban, dan memilih pengaturan *set point* batas atas dan batas bawah yang akan digunakan. Tombol naik dan turun digunakan untuk menaikkan dan menurunkan nilai *set point*. Sedangkan tombol *enable* digunakan untuk mengaktifkan kerja dari tombol-tombol pengaturan yang telah disebutkan sebelumnya. Jika tombol *enable* tidak aktif, maka tombol-tombol lain tidak akan berkerja. Keluaran yang dikendalikan dari sistem lokal ini adalah berupa pemanas dan pompa air.

Pada bagian monitoring jarak jauh, pemrosesnya berbasis Raspberry Pi 3. Untuk dapat melakukan monitoring jarak jauh pada sistem lokal, maka Zelio akan dihubungkan ke Raspberry Pi 3 dengan modbus TCP. Fungsi dari Raspberry Pi 3 adalah sebagai komputer yang menyediakan perangkat lunak untuk membuat sistem monitor jarak jauh, yang akan ditampilkan oleh GUI. *User* kemudian dapat melakukan pengawasan, penyetingan suhu dan kelembaban, dan penyimpanan dari GUI seperti yang ditunjukkan dalam model sistem pada gambar 1(b) bagian sistem monitor jarak jauh.

Perancangan perangkat lunak terdiri dari dua hal yaitu perancangan data antarmuka dan perancangan menu-menu dengan GUI. Bahasa pemrograman yang digunakan adalah python yang berorientasi pada obyek yang merupakan bahasa pemrograman interpreter mutiguna [6]. GUI menu akan dibuat menggunakan Tkinter yang adalah aplikasi GUI standar dari *python interface*.

Perancangan data antarmuka terdiri dari dua bagian yaitu konfigurasi alamat IP dan konfigurasi alamat dan perintah di Zelio Smart Relay. IP (*Internet Protocol*) *address* adalah alamat yang digunakan untuk menghubungkan satu perangkat ke perangkat lainnya. Dalam hal ini, IP *address* diperlukan untuk menghubungkan Raspberry dan Zelio. Pada komunikasi melalui protokol TCP/IP ini, IP yang akan digunakan adalah IP statis. IP statis ini bersifat tetap atau tidak berubah-ubah. Pada Zelio, konfigurasi jaringan network statis ini meliputi pengaturan terhadap IP *address* dan *subnet mask address*. Perancangan IP *address* untuk Zelio adalah pada alamat 192.168.1.10, dengan subnet mask pada alamat 255.255.255.0. Switch kemudian akan dengan mudah mengenali IP *address* yang bersifat tetap tersebut. Dalam mode *Function Block Diagram* di Zelio, modul Modbus TCP menggunakan tipe SR3NET01 dengan input extensi dari J1XT1 sampai J4XT1, dan output extensi dari O1XT1 sampai O4XT1[7][8]. Setiap input atau output extensi ini adalah dalam bentuk 16 bit Modbus word. Modbus TCP register untuk input J1XT1 adalah dari register 16 sampai register 19, register ini akan digunakan untuk menulis data, sedangkan register untuk O1XT1 sampai O4XT1 adalah dari register 20 sampai register 23, register ini akan digunakan untuk membaca data. Variabel yang akan dimonitor dan di-*set* akan ditentukan terlebih dahulu alamat registernya.

```

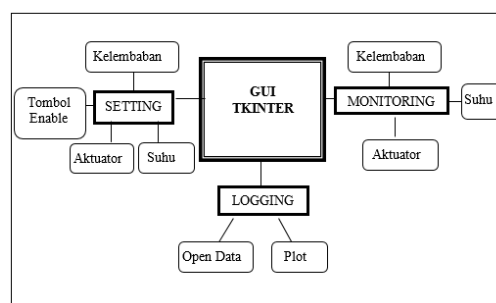
from pyModbusTCP.client import ModbusClient

zel = ModbusClient()
server_host="192.168.1.10"
server_port=502
zel.host(server_host)
zel.port(server_port)

if not zel.host("192.168.1.10"):
    print("host error")
if not zel.port(502):
    print("port error")
zel= ModbusClient (host = "192.168.1.10", auto_open=True)
    
```

Gambar 2. Listing Init Pemrograman Modbus TCP

Gambar 2 menunjukkan listing inisialisasi pemrograman modbus TCP. Modul *ModbusClient* dipanggil dari paket *library* modul *pyModbusTCP*. *ModbusClient* ini adalah perangkat yang bertindak sebagai client, dalam hal ini adalah Zelio. Inisialisasi dari *ModbusClient* dilakukan dengan fungsi *host/port return None if error*, atau fungsi yang mengembalikan nilai *none* jika tidak sesuai dengan *host/port* yang telah ditentukan. IP *address* Zelio yang sudah ditetapkan kemudian dimasukkan dalam syntax fungsi tersebut, yaitu 192.168.1.10, dan port default untuk ModbusTCP yaitu 502. Kode program *auto_open=True* adalah untuk membuat sambungan dari TCP agar tetap terbuka.



Gambar 3. Fitur Menu GUI

Perancangan menu dengan GUI terdapat menu utama dan submenu, yaitu menu utama, menu *setting*, menu monitoring dan menu *logging*. Gambar 4 menunjukkan fitur-fitur yang terdapat dalam menu-menu GUI. Menu utama adalah untuk membuka submenu. Dalam menu *setting* terdapat pengaturan nilai suhu batas atas (ba) dan batas bawah (bb) pada suhu dan kelembaban, pengaturan perintah on/off aktuator dari GUI, pembacaan ba dan bb suhu dan kelembaban sistem lokal Zelio, serta pengaturan on/off tombol *enable* dari GUI. Dalam menu monitoring terdapat pemantauan terhadap nilai suhu, nilai kelembaban dan status pemanas dan pompa yang terjadi pada saat itu berdasar pada interval waktu. Dalam menu *logging* terdapat penyimpanan file berdasar pada interval perekaman data, dan pembukaan data serta penampilan file dalam bentuk plot grafik.

```

from Tkinter import*
from tkMessageBox import*
import tkMessageBox
import tkFont
import numpy as np
import matplotlib.pyplot as plt
from tkFileDialog import askopenfilename
import tkFileDialog
import csv
import Tkinter, Tkconstants, tkFileDialog
from datetime import datetime
from dateutil import parser

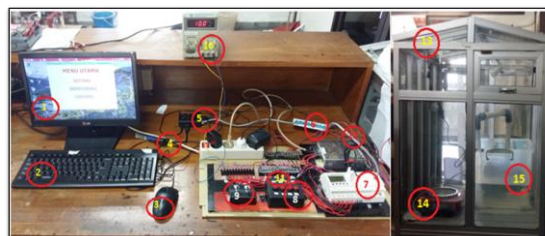
import matplotlib
matplotlib.use('TkAgg')
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg, NavigationToolbar2TkAgg

```

Gambar 4. Listing Pemrograman Pemanggilan Modul

Gambar 4 memperlihatkan mengenai listing pemanggilan modul atau import *package* pemrograman dalam python. Modul yang dipanggil adalah modul Tkinter, tkMessageBox, tkFont, numpy, tkFileDialog, csv, datetime, dateutil, dan matplotlib. Modul Tkinter dipanggil untuk pemrograman GUI. Modul tkMessageBox adalah modul yang digunakan untuk pemrograman beberapa pesan, contohnya pesan peringatan. Modul tkFont adalah pemanggilan untuk jenis huruf, besar huruf, dan ketebalan huruf. Modul numpy adalah library yang di dalamnya terdapat fungsi matematika untuk melakukan perhitungan. Modul tkFileDialog adalah package library untuk menampilkan dialog window dalam suatu window. Modul csv adalah modul yang dipakai untuk melakukan penyimpanan data dan pembacaan data ke dalam suatu file yang bertipe csv. Modul datetime adalah modul yang menampilkan waktu berupa tanggal, hari, jam, menit, dan detik. Modul dateutil adalah untuk pemanggilan ekstensi dari modul datetime standard, dalam hal ini berkaitan dengan parser yaitu untuk mengurai suatu format untuk mempresentasikan waktu. Modul matplotlib adalah modul untuk melakukan pemrograman terhadap plot, di dalamnya terdapat pembuatan label nama terhadap garis x dan y, navigasi toolbar, dan menampilkan plot dalam suatu canvas.

3. Hasil dan Pembahasan



Gambar 5. *Plant* Sistem Keseluruhan

Hasil racangan *plant* sistem secara keseluruhan dapat dilihat pada gambar 5. Pada sistem kendali lokal, dua potensiometer (pengganti sensor) dan tombol-tombol push button dihubungkan ke pin input Zelio Smart Relay. Potensiometer ini menerima tegangan dari Power Supply 10V. Tombol-tombol push button tersebut adalah tombol pilih yang berfungsi

untuk memilih jenis pengaturan *set point* suhu dan kelembaban, tombol *up* dan *down* untuk menaikkan dan menurunkan nilai *set point*, serta tombol *enable* yang berfungsi untuk mengaktifkan kerja dari semua tombol push button. LED adalah sebagai indikator dari tombol *enable*, dan jenis pengaturan *set point* yang dipilih, yaitu batas atas (ba) dan batas bawah (bb) suhu, serta batas atas dan bawah kelembaban. Kelima LED ini dihubungkan ke pin output Zelio. Keluaran yang dikendalikan dari sistem lokal adalah berupa pemanas dan pompa air. Pemanas dan pompa air ini dihubungkan ke pin output Zelio.

Pada sistem monitor jarak jauh, Keyboard, Mouse, kabel LAN dan kabel HDMI to VGA dihubungkan ke Raspberry Pi. Keyboard berfungsi untuk mengetik nilai-nilai *set point*. Mouse berfungsi untuk menggerakkan kursor dan melakukan pemilihan di window menu. Kabel HDMI to VGA juga dihubungkan ke monitor, kabel ini membuat grafis tertampil di layar monitor. Raspberry Pi berkomunikasi dengan Zelio dengan modul komunikasi Ethernet. Hal ini dilakukan dengan cara menghubungkan kabel LAN dari Raspberry Pi ke Network Switch, dan juga menghubungkan kabel LAN dari Modul Komunikasi Zelio ke Network Switch.

3.1. Pengujian Menu Setting

Pengujian pada menu *setting* dilakukan dengan menguji entry ba dan bb suhu dan kelembaban, tombol set pemanas, tombol set pompa, tombol on dan off, entry ba dan bb suhu dan kelembaban sistem lokal serta menguji tombol on dan off untuk *enable*. Gambar 6 menunjukkan mengenai tampilan menu *setting*.

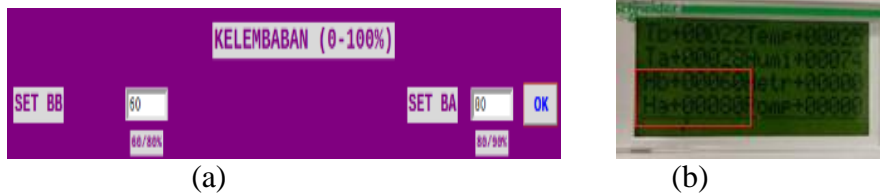


Gambar 6. Tampilan GUI Menu Setting

Pengujian data dilakukan dengan memasukkan nilai *set point* suhu dan kelembaban di GUI yaitu memasukkan nilai ba ke entry ba dan nilai bb ke entry bb suhu, kemudian tombol ok ditekan. Gambar 7(a) menunjukkan ketika nilai ba dan bb suhu dimasukan kedalam entry GUI, yaitu sebagai contoh nilai bb adalah 22, dan nilai ba adalah 28. Dalam Zelio pada gambar 7(b) dapat dilihat bahwa nilai Tb dan Ta sudah berubah menjadi Tb adalah 22 dan Ta adalah 28. Hal ini menunjukkan bawah nilai yang dimasukan dalam entry GUI suhu tersebut berhasil dikirim ke Zelio. Gambar 8(a) menunjukkan ketika nilai ba dan bb kelembaban dimasukan kedalam entry GUI, yaitu sebagai contoh nilai bb adalah 60, dan nilai ba adalah 80. Dalam Zelio pada gambar 8(b) dapat dilihat bahwa nilai Tb dan Ta sudah berubah menjadi Tb adalah 60 dan Ta adalah 80. Hal ini menunjukkan bawah nilai yang dimasukan dalam entry GUI kelembaban tersebut berhasil dikirim ke Zelio.



Gambar 7. (a) Pengujian *Set point* Suhu di GUI, dan (b) hasil di Zelio



Gambar 8. (a) Pengujian *Set point* Kelembaban di GUI dan (b) hasil di Zelio

Gambar 9 menunjukkan mengenai pengujian pembacaan nilai batas atas dan batas bawah suhu dan kelembaban sistem lokal. Pengujian data dilakukan dengan memasukan nilai *set point* dari sistem lokal Zelio. Nilai-nilai yang telah dimasukan tersebut berhasil dikirimkan ke GUI, hasil ini dapat dilihat pada gambar 10. Gambar 9 menunjukkan nilai ba dan bb suhu dan kelembaban yang diatur dari sistem lokal Zelio. Di Zelio tersebut, Tb bernilai 16, Ta bernilai 22, Hb bernilai 80, dan Ha bernilai 90. Nilai yang dibaca di GUI seperti yang ditunjukkan pada gambar 10. juga menunjukkan nilai yang sama, yaitu bb suhu bernilai 22, ba suhu bernilai 16, bb kelembaban bernilai 80 dan ba kelembaban berniali 90. Hal ini menunjukkan bawah nilai yang dimasukan dari sistem lokal Zelio berhasil dikirim dan dibaca di GUI.



Gambar 9. Pengujian Pembacaan *Set point* Suhu dan Kelembaban di Sistem Lokal



Gambar 10. Hasil Uji Pembacaan *Set point* Suhu dan Kelembaban di GUI

Tabel 1. Set Pemanas dan Pompa Kondisi On

Tombol Set Pemanas dan Pompa	Kondisi	Keadaan Pemanas dan Pompa	Indikator	
			Tombol On	Tombol Off
0	Awal	1	Relief /Biru	Relief/Biru
1	Awal	1	Sunken /Hijau	Relief /Biru
1	Tombol Off	0	Relief /Biru	Sunken /Hijau
1	Tombol On	1	Sunken /Hijau	Relief /Biru

Keadaan pemanas dan pompa adalah bernilai 1 atau kondisi on. Ketika tombol set pemanas atau pompa tidak ditekan atau kondisi 0, maka tombol on dan off tidak akan bisa mengirimkan perintah ke Zelio. Di GUI, indikator dari tombol on dan off adalah relief atau berwarna biru, ini menandakan bahwa ketika tombol ini ditekan, tidak akan ada aksi yang terjadi. Hasil di Zelio adalah tetap sama seperti sebelumnya, yaitu kondisi pemanas atau pompa adalah 1.

Ketika tombol set pemanas atau pompa ditekan, maka tombol on dan off dapat beroperasi. Kondisi awal pemanas atau pompa adalah 1, maka tombol on menjadi sunken atau berwarna hijau, sedangkan tombol off tetap relief atau berwarna biru. Ketika kondisi pemanas awal atau pompa bernilai 1 atau on ini dipaksa menjadi 0 atau off, maka tombol off ditekan atau diberi kondisi 1. Tombol off kemudian berubah menjadi berwarna hijau atau menjadi sunken, sedangkan tombol on akan menjadi relief/berwarna biru karena kondisi 0. Pemanas atau pompa di Zelio nilainya dari kondisi awal 1 berhasil dipaksa menjadi kondisi 0. Ketika kondisi pemanas awal 1, dan tombol on ditekan, maka kondisi pemanas akan tetap

bernilai 1. Hasil data uji tabel 1 ini menunjukkan bahwa rancangan indikator tombol relief/sunken dan perintah pemaksaan set aktuator pemanas dan pompa berhasil dilakukan.

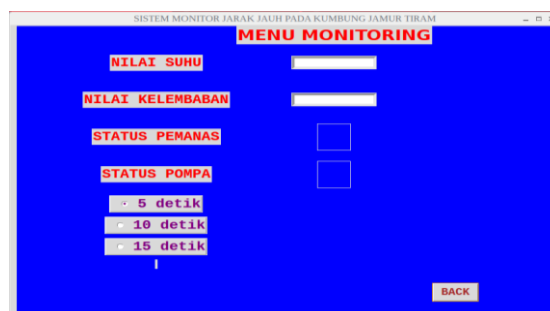
Tabel 2. Pengujian Bagian Tombol *Enable*

Tombol On	Tombol Off	Lokal	GUI
0	1	0	1
1	0	1	1

Pengujian pada bagian tombol *enable* ditunjukkan pada tabel 2. Jika tombol on di GUI tidak ditekan, maka sistem monitor jarak jauh hanya bisa diatur dari GUI saja. Jika tombol on di GUI ditekan, maka sistem bisa diatur dari GUI dan dari lokal. Tombol-tombol yang berada disistem lokal dapat dioperasikan dan digunakan untuk input pengaturan sistem lokal.

3.2. Pengujian Menu Monitoring

Gambar 11 menunjukkan tampilan menu monitoring. Gambar 12 memperlihatkan hasil uji pertukaran data yang terjadi dari Zelio ke GUI Raspberry Pi. Dalam monitoring ini, dapat dilihat pada gambar 12(a) bahwa di Zelio nilai suhu sekarang adalah 25, kelembaban sekarang 74, status pemanas adalah 0, dan status pompa adalah 0. Data dari Zelio ini berhasil dikirimkan ke GUI dan di menu monitoring tersebut tertampil nilai suhu adalah 25, nilai kelembaban adalah 74, status pemanas adalah berwarna merah atau kondisi 0, dan status pompa adalah berwarna merah atau kondisi 0. Hasil ini dapat dilihat pada gambar 12(b).



Gambar 11. Tampilan Menu Monitoring

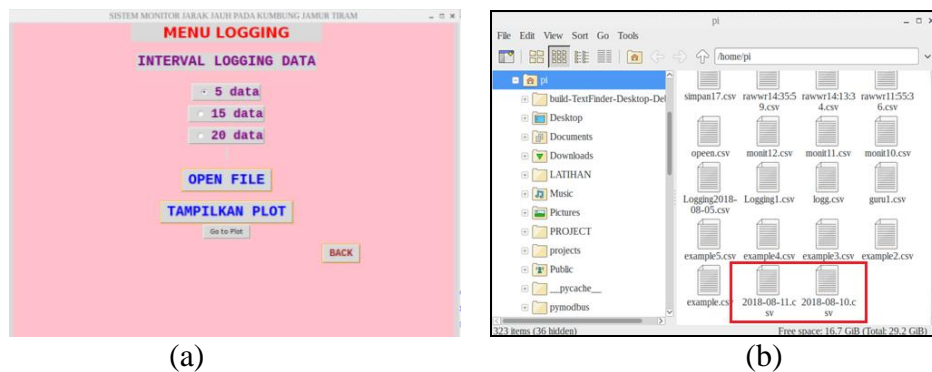


Gambar 12. (a) Data dari Zelio (b) Hasil Monitoring Data di GUI

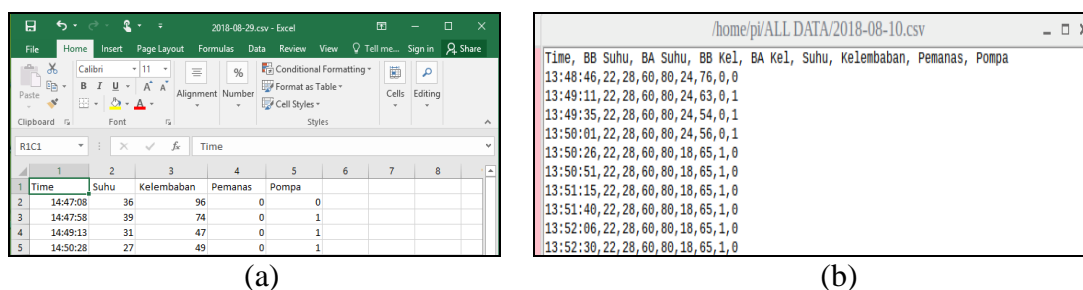
3.3. Pengujian Menu Logging

Gambar 13(a) menunjukkan mengenai tampilan menu *logging*. Gambar 13(b) menunjukkan bahwa penyimpanan otomatis nama file sesuai dengan tanggal berhasil dilakukan. Data-data yang dihasilkan dapat dilihat dalam lampiran 1, data-data-data ini adalah hasil dari kombinasi interval waktu 5 detik, 10 detik dan 15 detik dengan interval perekaman data 5 data, 15 data dan 20 data. Total data yang disimpan tersebut adalah 2545 data. Data-data ini diambil selama 27 jam dari tanggal 10-08-2018 jam 13:48:46 sampai

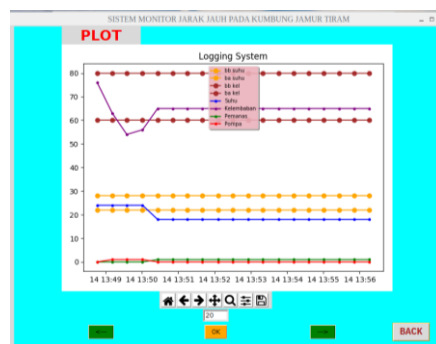
tanggal 11-08-2018 jam 17:11:38, dapat dilihat bahwa setelah jam 24.00 tanggal 10-08-2018, data-data masuk dalam file baru dengan nama file 11-08-2018 berhasil disimpan.



Gambar 13. (a) Tampilan Menu Logging, dan (b) Penyimpanan Nama File Otomatis.



Gambar 14. (a) Isi file 29-08-2018.csv, dan (b) File CSV yang dibuka dari Menu Logging



Gambar 15. Tampilan Plot Grafik

Gambar 14(a) menunjukkan mengenai data-data yang berhasil disimpan ke dalam file berformat csv, data-data tersebut disimpan berdasar pada hasil uji variasi kombinasi interval waktu 5, 10 dan 15 detik dan interval penyimpanan data 5, 15, dan 20 data. Gambar 14(b) memperlihatkan mengenai file csv yang berhasil dibuka dari GUI. File csv ini dibuka dalam bentuk text. Gambar 15 menunjukkan mengenai hasil tampilan plot dalam bentuk grafik. Pada window plot tersebut terdapat *entry* untuk memasukkan jumlah data yang ingin ditampilkan, tombol ok untuk mengeksekusi nilai *entry* tersebut, serta tombol kanan (→) untuk menggeser data tampilan plot kekanan 10 data dan tombol kiri (←) untuk menggeser data tampilan plot kekiri 10 data. Dapat dilihat pada gambar x ini, nilai *entry* yang dimasukkan adalah 20, oleh karena itu plot yang ditampilkan adalah dari data 1 sampai data 20.

4. Kesimpulan

Berdasarkan pengujian sistem monitor jarak jauh pada kumbung jamur tiram berbasis Raspberry Pi, didapatkan kesimpulan sebagai berikut. Program aplikasi GUI berhasil diimplementasikan untuk mengoperasikan sistem jarak jauh, sistem berhasil membuka

window submenu *setting*, *monitoring* dan *logging* pada menu utama serta mengembalikan window submenu ke menu utama. Komunikasi pertukaran data secara jarak jauh antara Raspberry Pi dan Zelio menggunakan jaringan LAN atau TCP/IP berhasil dilakukan. Program pada Raspberry Pi berhasil melakukan pengiriman data nilai batas atas dan batas bawah suhu dan kelembaban, pengaturan on/off aktuator, dan pengaturan tombol *enable* ke Zelio. Program pada Zelio berhasil mengirimkan data nilai suhu dan kelembaban sekarang, nilai batas atas dan batas bawah suhu dan kelembaban dari sistem lokal, serta status pemanas dan pompa ke Raspberry Pi yang ditampilkan di GUI menu *setting* dan *monitoring*. Pada menu *monitoring*, implementasi untuk menampilkan nilai suhu dan kelembaban serta status pemanas dan pompa berdasar pada interval waktu 5, 10, dan 15 detik berhasil dilakukan. Pada menu *logging*, data berhasil disimpan dalam format csv sesuai dengan interval perekaman sebesar 5, 15 atau 20 data, file csv tersebut berhasil dibuka, dan implementasi penamaan file secara otomatis sesuai dengan tanggal berhasil dilakukan. Pada menu *logging*, plot berhasil ditampilkan dalam bentuk grafik sesuai dengan file csv yang dipilih. Sistem memiliki kekurangan yaitu pada perubahan pemilihan interval perekaman data dari pilihan besar ke pilihan kecil atau sebaliknya, data yang telah dikumpulkan sebelum perubahan interval akan direset kembali yang menyebabkan data yang telah dikumpulkan tersebut tidak akan disimpan.

Ucapan Terimakasih

Peneliti mengucapkan terimakasih kepada Laboratorium Kendali dan Instrumentasi Teknik Elektro Universitas Sanata Dharma, atas fasilitas alat dan ruang yang dapat digunakan oleh peneliti.

Daftar Pustaka

- [1] KBBI. Teknologi, online: <https://kbbi.web.id/teknologi>. Diakses 03 November 2017.
- [2] Scientific, Campbell. *Irrigation and Canal Control*, Campbellsci, online: <https://www.campbellsci.com.au/canal-control>. Diakses 03 November 2017.
- [3] Kabarnusa. Petani Jamur Tiram di Buleleng Kewalahan Penuhi Pesanan, online: <http://www.kabarnusa.com/2017/11/petani-jamur-tiram-di-buleleng.html>. Diakses 03 November 2017.
- [4] Nursyah, Riskian. (2018). Sistem Kendali Suhu Dan Kelembaban Pada Kumbung Jamur Berbasis Zelio Smart Relay. Yogyakarta: Universitas Sanata Dharma.
- [5] Raspberry. *What Is A Raspberry Pi*, online: <https://www.raspberrypi.org/help/faqs/RaspberryPiFoundation>. Diakses 8 November 2017.
- [6] Python, Belajar. Pengertian Python, online: <https://www.belajarpython.com/>. Diakses 10 November 2017.
- [7] Electric, Schneider. Zelio Soft, online: <https://www.schneider-electric.co.uk/en/product-range/542-zelio-soft/>. Diakses 11 November 2017.
- [8] Telemecanique. Zelio Logic - Communication Extension, online: http://www.electrocentr.com.ua/files/documentation/SE/plc/zelio/guide/ZelioLogic_Modbus_Ethernet_Extensions_2006_en.ppt. Diakses 28 November 2017.