

Implementasi CLIPS dalam Pengambilan Keputusan Berdasarkan Hasil Baca Sensor untuk Monitoring Ruang Budidaya Jamur Tiram

Adha Mashur Sajiah, Noor Akhmad Setiawan, Oyas Wahyunggoro, Alfadoni Prisantama

Departemen Teknik Elektro dan Teknologi Informasi, Fakultas Teknik,
Universitas Gadjah Mada
Jalan Grafika No. 2 Yogyakarta—55281, Indonesia
adha.m.sajiah@gmail.com

Abstrak

Jamur tiram termasuk "tanaman" heterotropik yang hidupnya bergantung pada lingkungan. Sekarang ini sudah banyak usaha pembudidayaan jamur tersebut. Karena lingkungan idealnya adalah di dataran tinggi maka usaha pembudidayaan jamur tiram di dataran rendah memerlukan penjagaan nilai optimal dari suhu, kelembaban dan pencahayaan ruang budidayanya. Sayangnya *monitoring* tanaman jamur kebanyakan masih manual. Oleh karena itu dalam penelitian ini dilakukan pengembangan sistem *monitoring* budidaya jamur tiram dengan mengendalikan suhu, kelembaban dan cahaya optimal dengan pengambilan keputusan cerdas terotomatisasi. Sistem pengambilan keputusan ini dibuat menggunakan bahasa pemrograman CLIPS yaitu bahasa pemrograman yang dibuat khusus untuk sistem pakar. Sintaks-sintaksnya yang memang dibuat berdasarkan model penarikan kesimpulan IF-THEN. Selain CLIPS sistem juga digabungkan dengan sistem Java yang bertugas membaca input data sensor. Sistem ini berhasil dibuat dengan menggunakan 10 aturan aksi dan 5 kondisi sensor. Dengan sistem ini diharapkan pembudidaya jamur tiram dapat menggunakannya untuk memantau dan menjaga kondisi ideal jamur tiram.

Kata Kunci: CLIPS, jamur tiram, *monitoring*, pengambilan keputusan.

1. Pendahuluan

Saat ini, pembudidayaan jamur tiram mengalami perkembangan yang pesat. Budidaya jamur tiram yang pada umumnya dilakukan di daerah dataran tinggi yang memiliki suhu yang dingin, saat ini sudah banyak dibudidayakan di dataran rendah. Agar pertumbuhan jamur pada daerah dataran rendah dapat optimal maka suhu, kelembaban, dan pencahayaan ruang budidaya jamur harus dijaga sesuai dengan kondisi idealnya (Martawijawa dan Nurjayadi, 2010).

Sayangnya, *monitoring* tanaman masih dilakukan secara manual, sehingga waktu penyiraman hanya mengandalkan termometer ruangan dan cukup mengurus tenaga pembudidaya jamur tiram karena harus bolak-balik menyiram jamur agar suhu dan kelembabannya sesuai dengan kebutuhan jamur tiram (Istuti dan Nurbana, 2006). Ini artinya banyak sumber daya manusia dan waktu yang dibutuhkan hanya untuk melakukan *monitoring* ruang budidaya jamur. Oleh karena itu, para dibutuhkan suatu sensor yang dapat membantu meringankan kegiatan *monitoring* ruang budidaya jamur tersebut (Susilowati dan Rahardjo, 2010).

Sensor tersebut bekerja secara otomatis sehingga pengambilan keputusan yang tepat dapat dilakukan dengan berdasarkan suhu,

kelembaban, dan intensitas cahaya yang masuk pada ruang budidaya jamur, dimana kondisi ideal untuk pertumbuhan jamur tiram adalah suhu $\pm 28^{\circ}\text{C}$, kelembaban 80% - 90% RH, dan intensitas cahaya yang masuk 40 lux (Agromedia, 2006). Untuk menjaga kondisi ruang budidaya jamur agar dalam kondisi yang ideal, dapat digunakan sensor suhu, sensor kelembaban, dan sensor cahaya, dengan aktuator blower untuk menjaga suhu, sprayer untuk menjaga kelembaban, dan lampu untuk mengatur intensitas cahaya yang masuk.

CLIPS merupakan singkatan dari 'C Language Integrated Production System'. CLIPS adalah salah satu alat perangkat lunak pemrograman yang dikhususkan untuk membangun sistem pakar. Perangkat lunak ini digunakan secara luas dalam sistem pakar (Di Stefano, 2005). CLIPS memiliki kelebihan dalam representasi pengetahuan, portabilitas dan ekstensibilitas.

Penelitian ini berupaya mengimplementasikan CLIPS untuk mengambil keputusan berdasarkan hasil baca sensor suhu, kelembaban, dan intensitas cahaya yang masuk pada ruang budidaya jamur, dalam bentuk *expert system*. Hal ini dilakukan untuk mempermudah *monitoring* ruang budidaya jamur.

Tujuan penelitian ini diantaranya adalah:

- a. Mengimplementasikan parameter ideal dan tidak ideal dari suhu, kelembaban dan intensitas cahaya yang masuk pada ruang budidaya jamur.
- b. Mengembangkan sistem yang dapat mengambil keputusan berdasarkan hasil baca sensor suhu, kelembaban dan intensitas cahaya yang masuk pada ruang budidaya jamur.

2. Metode

Pada bagian ini dijelaskan struktur CLIPS, diagram alir sistem sistem pengambil keputusan pada ruang *monitoring* jamur, *rule* yang digunakan sebagai basis pengetahuan sistem dan implementasi sistem dalam bahasa pemrograman CLIPS.

2.1 CLIPS

Sistem yang diusulkan pada makalah ini diimplementasikan dengan CLIPS. CLIPS merupakan alat yang dikenal secara luas dalam sistem pakar.

CLIPS adalah alat pengembangan sistem pakar yang menyediakan lingkungan menyeluruh untuk pembuatan *rule* dan objek berbasis sistem pakar. Fitur utama dari CLIPS adalah:

1. *Knowledge representation*: CLIPS menyediakan alat padu untuk menangani berbagai jenis pengetahuan dengan dukungan tiga paradigma pemrograman berbeda, yaitu: berbasis aturan, berorientasi objek dan prosedural. Pemrograman berbasis aturan memungkinkan pengetahuan dinyatakan dalam heuristik, yang merinci kumpulan aksi yang harus dilakukan pada situasi tertentu. Pemrograman berorientasi objek memungkinkan sistem kompleks dinyatakan dalam komponen modular. Sedangkan kemampuan pemrograman prosedural mirip seperti bahasa pemrograman yang ada pada C++, Java dan Python.
2. Portabilitas: CLIPS ditulis dalam bahasa C untuk portabilitas dan kecepatan serta telah dipasang pada banyak sistem operasi yang berbeda tanpa perubahan kode. Sistem operasi dimana CLIPS telah diujikan adalah Windows XP, MacOS X dan Unix. CLIPS dapat ditransfer ke sistem manapun yang mempunyai compiler C atau C++ standar ANSI.
3. Integrasi/Ekstensibilitas: CLIPS dapat dibenamkan dalam kode prosedural, yang disebut *subroutine* dan diintegrasikan dengan bahasa lain seperti C, Java, FORTRAN dan ADA.

Seperti bahasa sistem pakar lain, CLIPS juga berurusan dengan *rule* dan fakta (*fact*). Berbagai jenis fakta dapat membuat *rule* berlaku. Sebuah

rule yang berlaku kemudian ditegaskan (diaktifkan). Fakta dan *rule* dibuat dengan mendefinisikan mereka, seperti berikut ini:

```
1 (deftemplate car_problem
2   (slot name)
3   (slot status)
4 )
5
6 (deffacts trouble_shooting
7   (car_problem (name ignition_key)
8   (status on))
9   (car_problem (name engine)
10  (status wont_start))
11  (car_problem (name headlights)
12  (status work))
13 )
14
15 (defrule rule1
16   (car_problem (name ignition_key)
17   (status on))
18   (car_problem (name engine) (status
19   wont_start))
20   =>
21   (assert (car_problem (name starter)
22   (status faulty)))
23 )
```

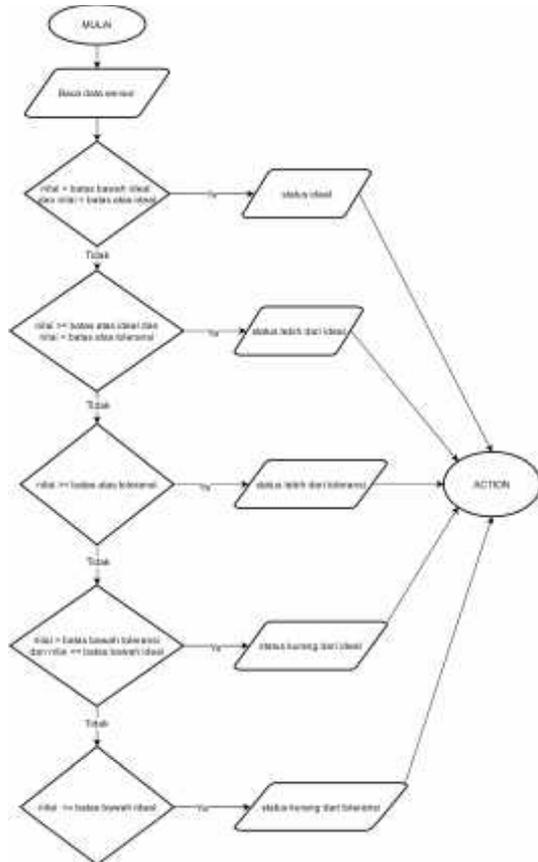
Pada listing program di atas baris 1-4 merupakan perintah mendeklarasikan sebuah tipe objek dengan *deftemplate*. Objek tersebut bernama *car_problem*. *car_problem* memiliki dua atribut yaitu *name* dan *status*.

Pada baris 6-10 merupakan perintah definisi fakta atau kejadian yang menjadi dasar aktifnya *rule*. Baris 7-8 menyatakan bahwa ada fakta *car_problem* dengan *name* adalah *ignition_key* dan *status* adalah *on*. Fakta ke dua adalah ada *car_problem* dengan *name* adalah *engine* dan *status* adalah *wont_start*. Sedangkan pada baris 15-2 merupakan definisi *rule* menjadi basis pengetahuan untuk melakukan inferensi keputusan. Bentuk *rule* ini adalah IF-THEN. Sebelah kiri dari simbol => adalah bagian IF (*antecedent*) dan sebelah kanan dari simbol => adalah bagian THEN (*consequent*).

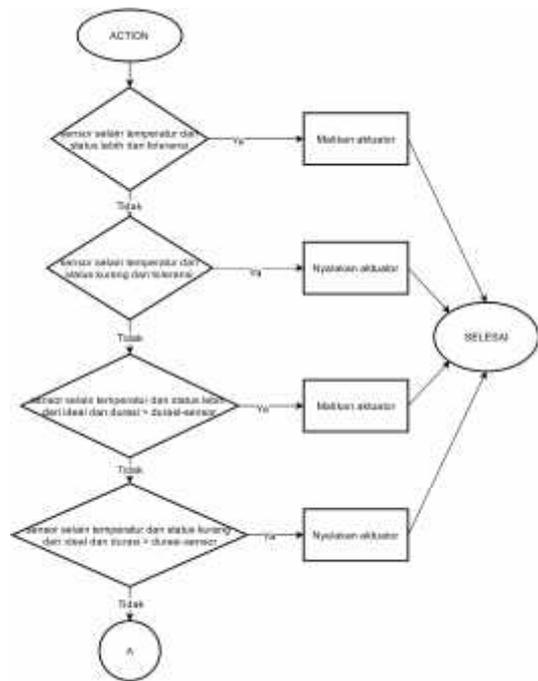
2.2 Diagram Alir Sistem

Berdasarkan parameter pada Bab 2, maka dapat dibangun *rule* sekuensial untuk membangun sistem pengambilan keputusan berdasarkan hasil baca sensor untuk monitoring ruang budidaya. Sistem yang dirancang pada penelitian ini memiliki diagram alir seperti pada Gambar 1-3. Gambar 1. menjelaskan bahwa sistem dimulai dengan pembacaan data sensor (yaitu sensor suhu, kelembaban dan cahaya). Selanjutnya setelah pembacaan data sensor data diperiksa jika berada di status ideal, lebih dari ideal, lebih dari toleransi, kurang dari ideal dan kurang dari toleransi. Masing-masing keadaan tersebut akan berujung pada aksi untuk aktuatur. Untuk

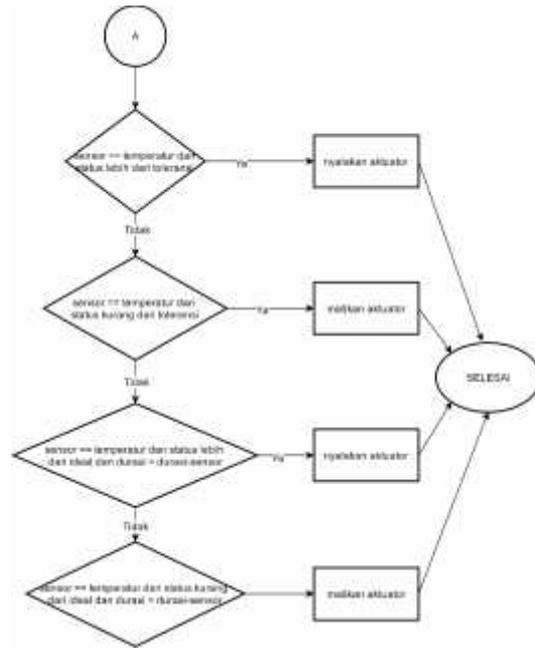
memeriksa masing-masing *state* tersebut digunakan *rule* yang dijelaskan pada sub bab selanjutnya.



Gambar 1. Diagram Alir Sistem 1 (States)



Gambar 2. Diagram Alir Sistem 2 (Action 1)



Gambar 3. Diagram Alir Sistem 3 (Action 2)

Gambar 2. menunjukkan langkah-langkah yang dilakukan sistem ketika merespon data yang masuk. Ketika memulai pengecekan sistem memeriksa apakah sensor selain temperatur dan status sistem lebih dari toleransi. Jika 'ya' maka aktuator (*blower* dan lampu) akan dimatikan sedangkan jika sebaliknya maka akan diperiksa dengan uji selanjutnya. Uji selanjutnya adalah apakah sensor selain temperatur dan status kurang dari toleransi. Jika 'ya' maka aktuator akan dinyalakan. Sedangkan jika 'tidak' maka diuji lanjutan. Pada proses ini diperiksa apakah sensor selain temperatur dan status lebih dari ideal serta durasi melebihi durasi sensor. Jika 'ya' maka aktuator dimatikan sedangkan jika tidak lanjut ke uji selanjutnya. Pada bagian ini diuji apakah sensor selain temperatur dan status kurang dari ideal dan durasi melebihi durasi sensor. Jika 'ya' maka aktuator harus dinyalakan dan jika tidak maka lanjut ke proses selanjutnya yang dijelaskan pada Gambar 3.

Gambar 3. Merupakan proses yang menjelaskan pemeriksaan data sensor dan keputusan yang harus diambil berkaitan dengan sensor suhu. Gambar ini merupakan kelanjutan proses Gambar 2. Ditandai dengan konektor proses. Pada dasarnya prosesnya sama dengan sensor selain temperatur (Gambar 2.) tetapi aksi yang dilakukan oleh temperatur berkebalikan pada kondisi yang sama.

2.3 Rule Pada Sistem

Sistem secara berurutan akan mengecek parameter suhu, kelembaban, dan intensitas cahaya yang masuk dengan *rule* sebagai berikut.

< NT | < NI | I | > NI | > NT

dengan NT: Nilai Toleransi, NI: Nilai Ideal dan I: Ideal.

Suhu

< NT	< NI	I	> NI	> NT
20	22	26	30	

- Jika suhu lebih dari nilai suhu ideal, maka akan diberikan peringatan sampai 3 kali *cycle*, setelah mencapai 3 kali *cycle* maka *blower* otomatis dinyalakan.
- Jika suhu kurang dari nilai suhu ideal, maka akan diberikan peringatan sampai 3 kali *cycle*, setelah mencapai 3 kali *cycle* maka *blower* otomatis dimatikan.
- Jika suhu lebih dari nilai toleransi suhu, maka *blower* otomatis dinyalakan.
- Jika suhu kurang dari nilai toleransi suhu, maka *blower* otomatis dimatikan.

Kelembaban

< NT	< NI	I	> NI	> NT
60	80	85	90	

- Jika kelembaban lebih dari nilai kelembaban ideal, maka akan diberikan peringatan sampai 5 kali *cycle*, setelah mencapai 5 kali *cycle* maka *sprayer* otomatis dimatikan.
- Jika kelembaban kurang dari nilai kelembaban ideal, maka akan diberikan peringatan sampai 5 kali *cycle*, setelah mencapai 5 kali *cycle* maka *sprayer* otomatis dinyalakan.
- Jika kelembaban lebih dari nilai toleransi kelembaban, maka *sprayer* otomatis dimatikan.
- Jika kelembaban kurang dari nilai toleransi kelembaban, maka *sprayer* otomatis dinyalakan.

Intensitas Cahaya

< NT	< NI	I	> NI	> NT
40	50	300	310	

- Jika intensitas cahaya lebih dari nilai intensitas cahaya ideal, maka akan diberikan peringatan sampai 4 kali *cycle*, setelah mencapai 4 kali *cycle* maka lampu otomatis dimatikan.
- Jika intensitas cahaya kurang dari nilai intensitas cahaya ideal, maka akan diberikan peringatan sampai 4 kali *cycle*, setelah mencapai 4 kali *cycle* maka lampu otomatis dinyalakan.
- Jika intensitas cahaya lebih dari nilai toleransi intensitas cahaya, maka lampu otomatis dimatikan.

- Jika intensitas cahaya kurang dari nilai toleransi intensitas cahaya, maka lampu otomatis dinyalakan.

2.4 Implementasi Rule dengan CLIPS

Berikut *source code* CLIPS dan Java untuk mengambil keputusan berdasarkan hasil baca sensor:

```

;;; PENENTUAN PARAMETER IDEAL, KURANG IDEAL, DAN TIDAK IDEAL
(defacts MAIN::sensor-information
  (sensor (name temperature) (device blower)
    (less-than-acceptable 20)
    (less-than-normal 22)
    (more-than-normal 26) (more-than-acceptable 30))
  (sensor (name humidity) (device sprayer)
    (less-than-acceptable 60)
    (less-than-normal 80)
    (more-than-normal 85) (more-than-acceptable 90))
  (sensor (name light) (device light)
    (less-than-acceptable 40)
    (less-than-normal 50)
    (more-than-normal 300) (more-than-acceptable 310)))
...

;;; AKSI SETELAH ANALISIS PEMBACAAN DATA SENSOR (OUTSIDE NORMAL)(SELAIN TEMPERATUR)
(defmodule ACTION (import MAIN ?ALL))

(defrule ACTION::Shutdown--More-Than-Acceptable
  (cycle ?time)
  (sensor-trend
    (name ?sensor&~temperature)
    (state ?state&more-than-acceptable))
  (sensor (name ?sensor) (device ?device))
  ?on <- (device (name ?device) (status on))
  =>
  (printout t "Cycle " ?time " - ")
  (printout t "Sensor " ?sensor " reading is " ?state crlf)
  (printout t " Shutting down device " ?device
    crlf)
  (bind ?txt-info (str-cat "Cycle " ?time " - " "Sensor " ?sensor " reading is " ?state))
  (bind ?txt-action-info (str-cat "-->Shutting down device " ?device))
  (assert (actuator-info(info ?txt-info)))
  (assert (actuator-info(info ?txt-action-info)))
  )
...

```

Sementara berikut ini adalah file yang berisi hasil baca sensor yang akan dianalisis dan diambil suatu keputusan pada aktuator yang ada:

sense.txt
 temperature 19
 humidity 50
 light 20
 end-of-cycle
 temperature 21
 humidity 70
 light 45
 end-of-cycle
 temperature 25
 humidity 82
 light 100
 end-of-cycle
 temperature 27
 humidity 87
 light 305
 end-of-cycle
 temperature 32
 humidity 92
 light 315
 end-of-cycle
 temperature 20
 humidity 60
 light 40
 end-of-cycle

3. Hasil dan Pembahasan

Berikut ini adalah hasil eksekusi program *monitoring* ruang budidaya jamur tiram:



Gambar 4. Tampilan Awal Sistem Monitoring



Gambar 5. Tampilan Sistem Monitoring Saat Menjalankan File *sense.txt*

Gambar 4. menunjukkan hasil eksekusi dengan data masukan sebagai berikut:

temperature 19
 humidity 50
 light 20
 end-of-cycle
 temperature 21
 humidity 70
 light 45
 end-of-cycle
 temperature 25
 humidity 82
 light 100
 end-of-cycle

Dengan menggunakan ketentuan pada Subbab 2.2, didapatkan:

Cycle 1

Suhu

< NI		< NI	I		> NI		> NI
	20		22	26		30	
19							

Kelembaban

< NI		< NI	I		> NI		> NI
	60		80	85		90	
50							

Intensitas Cahaya

< NI		< NI	I		> NI		> NI
	40		50	300		310	
20							

Cycle 2

Suhu

< NI		< NI	I		> NI		> NI
	20		22	26		30	
		21					

Kelembaban

< NI		< NI	I		> NI		> NI
	60		80	85		90	
		70					

Intensitas Cahaya

< NI		< NI	I		> NI		> NI
	40		50	300		310	
		45					

Cycle 3

Suhu

< NI		< NI	I		> NI		> NI
	20		22	26		30	
			25				

Kelembaban

< NI		< NI	I		> NI		> NI
	60		80	85		90	
			82				

Intensitas Cahaya

< NT	< NI	I	> NI	> NT
40	50	300	310	
	100			



Gambar 6. Tampilan Sistem Monitoring Saat Menjalankan File sense2.txt

Sementara itu, Gambar 6. menunjukkan hasil eksekusi dengan data masukan sebagai berikut:

temperature 21
 humidity 70
 light 45
 end-of-cycle
 temperature 21
 humidity 70
 light 45
 end-of-cycle
 temperature 21
 humidity 70
 light 45
 end-of-cycle

Dengan menggunakan ketentuan pada Subbab 2.2, didapatkan:

Cycle 1

Suhu

< NT	< NI	I	> NI	> NT
20	22	26	30	
	21			

Kelembaban

< NT	< NI	I	> NI	> NT
60	80	85	90	
	70			

Intensitas Cahaya

< NT	< NI	I	> NI	> NT
40	50	300	310	
	45			

Cycle 2

Suhu

< NT	< NI	I	> NI	> NT
20	22	26	30	
	21			

Kelembaban

< NT	< NI	I	> NI	> NT
60	80	85	90	
	70			

Intensitas Cahaya

< NT	< NI	I	> NI	> NT
40	50	300	310	
	45			

Cycle 3

Suhu

< NT	< NI	I	> NI	> NT
20	22	26	30	
	21			

Kelembaban

< NT	< NI	I	> NI	> NT
60	80	85	90	
	70			

Intensitas Cahaya

< NT	< NI	I	> NI	> NT
40	50	300	310	
	45			

Terlihat bahwa suhu berada pada keadaan "Kurang dari Nilai Ideal" selama 3 cycle, sehingga blower dimatikan.

4. Kesimpulan

Pada penelitian telah dilakukan beberapa hal sebagai berikut:

- Rule mengenai suhu, kelembaban, dan intensitas cahaya ideal dan tidak ideal pada ruang budidaya jamur tiram telah berhasil disimpan sebagai knowledge dalam sistem komputer dengan menggunakan "CLIPS".
- Pengambilan keputusan berdasarkan hasil baca sensor untuk monitoring ruang budidaya jamur tiram telah berhasil diimplementasikan dengan menggunakan "CLIPS".

Daftar Pustaka

A. Budiman, I. Islami, M. S. Hadi, S. Vokasi, P. Tanaman, S. Vokasi, and B. Perikanan, (2015). "AUTO HI-IS: Solusi Cerdas Budidaya Jamur Konsumsi dengan Automatic Humidity," Yogyakarta. Agromedia, (2006). Budi Daya Jamur Konsumsi. Jakarta: Agromedia Pustaka. Di Stefano, Antonella, (2005). "ERESYE: artificial intelligence in Erlang programs,"

- Tallin, Estonia: ACM. Pp. 62-71. ISBN 1-59593-066-3.
- E. I. Martawijawa and M. Y. Nurjayadi, (2010). *Bisnis Jamur Tiram di Rumah Sendiri*. Bogor, Indonesia: IPB Press.
- F. Budiawan, A. Jaya, and Irianto, (2010). "Pengaturan Suhu dan Kelembaban Pada Miniatur Kumbung untuk Meningkatkan Produktifitas Jamur Tiram," Surabaya, Indonesia.
- G. Ramadhani, (2015). "Sistem Pemantauan Kondisi Suhu dan Kelembapan pada Pembudidayaan Jamur Tiram Menggunakan Mikrokontroler Arduino dengan Sensor DHT11," Surabaya, Indonesia.
- J. Nugroho, (2014). "Sistem Monitoring Pendeteksi Suhu dan Kelembapan pada Rumah Jamur Berbasis Mikrokontroler AT-Mega 328," Universitas Muhammadiyah Ponorogo.
- L. Oktavia, T. Rifai, G. Sebastian, A. Gunawan, and Z. Agustina, (2012). "KELEMBAPAN TERHADAP JAMUR 'Studi kasus: Menentukan Material yang Baru untuk Rumah Jamur/Kumbung Jamur,'" Universitas Atma Jaya Yogyakarta.
- N. M. Djarijah and A. S. Djarijah, (2001). *Budidaya Jamur Tiram Putih*. Yogyakarta: Kanisius.
- Susilowati and B. Rahardjo, (2010). "Petunjuk Teknis Budidaya Jamur Tiram Putih (*Pleurotus ostreatus* var *florida*) yang Ramah Lingkungan".
- W. Istuti and S. Nurbana, (2006). *Budidaya Jamur Tiram*. Jawa Timur: Balai Pengkajian Teknologi Pertanian.
- Widodo, C. S. Prabowo, S. Winanti, and R. E. Juwanto, (2013). "Rancang Bangun Sistem Penyiraman Tiram Secara Otomatis Menggunakan Sensor Suhu Berbasis Mikrokontroler Atmega8," *J. Ris. Drh.* 2013, pp. 31-40.
- G. Riley, "CLIPS A Tool for Building Expert Systems," (2016). [Online]. Available: <http://www.clipsrules.net/>.