

## 1. Pendahuluan

Perkembangan ilmu pengetahuan dan teknologi informasi sekarang ini sudah semakin maju dan berkembang, dimana manusia terus berinovasi dalam menciptakan sesuatu yang dapat memudahkan aktivitas dan pekerjaan manusia. Salah satu contohnya dalam bidang industri perfilman yaitu bioskop.

Bioskop merupakan suatu tempat untuk menonton pertunjukan film. Guna membantu dalam hal pemilihan film oleh konsumen dan penjualan tiket di bioskop maka dibuatlah suatu sistem informasi yang dapat memberikan informasi tentang film apa saja yang tersedia, waktu film tersebut diputar dan juga tempat duduk yang tersedia di dalam teater bioskop.

Teori Bahasa dan Automata merupakan sebuah metode yang memiliki model dan gagasan dalam perancangan sebuah sistem. *Finite state automata* menerapkan model matematika dalam pengambilan keputusan ketika menerima *input* berupa bahasa yang dimengerti atau dikenali oleh mesin dan menghasilkan *output* berdasarkan aturan-aturan yang ditetapkan.

Berdasarkan latar belakang yang ada, maka dilakukan penelitian yang bertujuan membuat perancangan sistem untuk pembelian tiket bioskop dan divalidasi menggunakan *Finite State Automata*. *Finite state Automata* digunakan sebagai alur logis untuk validasi proses perancangan sistem yang dibangun. Penelitian yang dilakukan diharapkan dapat membuktikan bahwa sistem yang dihasilkan sesuai dengan rancangan yang dilakukan.

## 2. Tinjauan Pustaka

Sudah banyak penelitian tentang penerapan atau penggunaan *Finite State Automata*, salah satu contoh penelitian yang berjudul Validasi Aplikasi Diagnosa Penyakit Kanker Serviks Menggunakan *Finite State Automata*, membahas tentang membuat perancangan sistem untuk mendiagnosa penyakit kanker serviks dan divalidasi menggunakan *Finite State Automata*. *Finite state Automata* digunakan sebagai alur logis untuk validasi proses perancangan sistem yang dibangun, dan dapat disimpulkan bahwa validasi yang dilakukan menggunakan *Finite State Automata* dapat membuktikan alur logis dari sistem diagnosa penyakit kanker serviks [1].

Penelitian yang kedua berjudul Penerapan *Finite State Automata* Pada Layanan Reservasi Tiket Bis, membahas tentang bagaimana *Finite State Automata* digunakan untuk memetakan bagaimana proses berjalannya sistem reservasi tiket bis sehingga dapat memudahkan penyelesaian sistem ini [2].

Penelitian yang ketiga berjudul Penerapan Konsep *Finite State Automata* (FSA) Pada Mesin Pembuat Minuman Kopi Otomatis, membahas tentang aplikasi simulasi mesin kopi otomatis yang dapat melakukan proses pembuatan 7 jenis pilihan minuman. Dalam aplikasi ini konsep *Finite State Automata* (FSA) diterapkan untuk mengenali dan menangkap pola pada proses pembuatan minuman kopi [3].

Berdasarkan penelitian-penelitian sebelumnya terkait dengan penerapan *Finite State Automata*, maka dilakukan penelitian yang membahas tentang validasi alur logis sistem pembelian tiket bioskop menggunakan *finite state automata*.

*Finite State Automata* (FSA) disebut juga *Finite Automata* merupakan suatu model matematika dari suatu sistem yang menerima *input* dan menghasilkan *output* diskrit. *Finite state automata* memiliki *state* yang banyaknya berhingga (terbatas), dan dapat berpindah-pindah dari satu *state* ke *state* lain. Perubahan *state* ini dinyatakan dengan fungsi transisi. *State* adalah kondisi atau keadaan atau kedudukan [4].

Prinsip kerja *Finite Automata* adalah sebagai berikut:

- Menerima *input* string.
- Membaca (menyerap substring) karakter awal dengan kontrol berada pada *state* awal.
- Dengan kontrol dan karakter awal yang telah dibaca tadi, *state* akan berpindah ke *state* baru.
- Proses berlanjut sampai semua string terserap habis.
- Jika *state* akhir yang ditempati saat string habis tersebut berada dalam himpunan *final state* yang telah ditentukan, maka string tersebut diterima atau dikenali oleh *Finite Automata* tersebut. Jika tidak, maka string tersebut ditolak atau tidak dikenali oleh *Finite Automata* itu.

Sebuah *Finite Automata*  $M$  dinyatakan dengan lima tupel, yaitu  $(Q, \Sigma, \delta, S, F)$ , di mana:

$Q$  = himpunan *state*

$\Sigma$  = himpunan alfabet masukan

$\delta$  = fungsi transisi

$S$  = *state* awal (*initial state*)

$F$  = himpunan *state* akhir (*final state*)

Misal diketahui sebuah *state diagram* seperti pada Gambar 1.



**Gambar 1** Contoh *State Diagram Finite Automata*

Keterangan Gambar 1:

- Gambar lingkaran menyatakan *state*
- Label pada lingkaran adalah nama *state* tersebut
- Busur panah menyatakan transisi atau perpindahan *state*
- Gambar lingkaran yang didahului sebuah busur panah tanpa label menyatakan *state* awal
- Gambar lingkaran ganda menyatakan *final state*

Maka:

$$Q = \{A, B\}$$

$$\Sigma = \{0,1\}$$

$$S = \{A\}$$

$$F = \{B\}$$

$\delta$  = fungsi transisi

$$\delta(A, 0) = A$$

$$\delta(A, 1) = B$$

$$\delta(B, 0) = B$$

$$\delta(B, 1) = A$$

Berdasarkan fungsi transisi tersebut, dapat dibuat tabel transisi seperti pada Tabel 1.

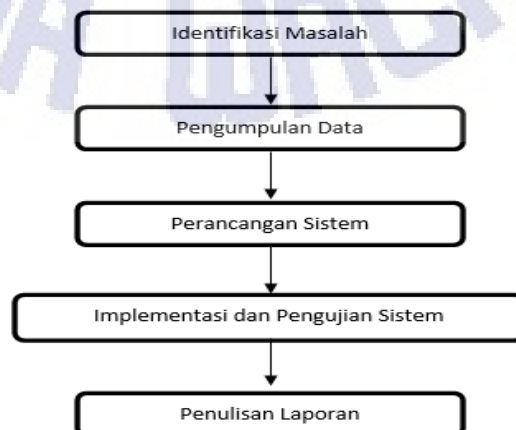
**Tabel 1** Tabel Transisi

$\delta$	0	1
A	A	B
B	B	A

Contoh bila string yang masuk adalah 1011, maka string tersebut bergerak dari *Start* ke *state* A, kemudian membaca karakter '1' dan berpindah ke *state* B, yang merupakan *state* tujuan dari hasil pembacaan karakter '1'. Kemudian string selanjutnya yang dibaca adalah '0'. Karena *state* tujuan dari pembacaan karakter '0' adalah B sendiri, maka *state* tidak berpindah. Selanjutnya membaca karakter '1', dari *state* B berpindah ke *state* A yang merupakan *state* tujuan setelah membaca karakter '1'. Setelah itu, karakter '1' dibaca dan *state* berpindah ke *state* B. Pembacaan string berhenti karena karakter sudah habis. *State* terakhir yang ditempati adalah *state* B. Karena *state* B berada dalam himpunan *final state*, maka string '1011' diterima oleh *Finite Automata* tersebut [4].

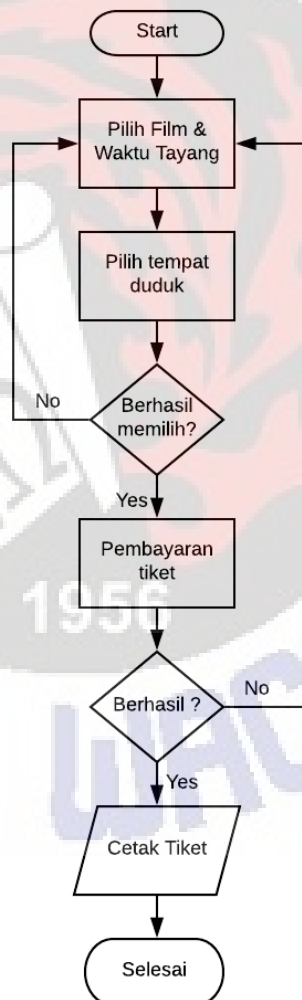
### 3. Metode dan Perancangan Sistem

Tahapan yang dilakukan dalam penelitian ini terdiri dari 5 (lima) tahapan yaitu: (1) Identifikasi masalah, (2) Pengumpulan data, (3) Perancangan sistem, (4) Implementasi dan pengujian sistem, (5) Penulisan laporan.



**Gambar 2** Tahapan penelitian

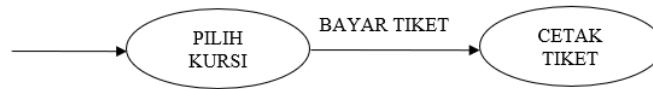
Tahapan penelitian pada Gambar 2 dapat dijelaskan sebagai berikut. Pada tahap identifikasi masalah dilakukan identifikasi terhadap permasalahan sistem pembelian tiket menggunakan *finite state automata*. Pengumpulan data pada tahapan ini dilakukan pengumpulan data dari jurnal-jurnal maupun buku-buku yang berkaitan dengan pembahasan penelitian melalui studi *literature*. Pada tahap pembuatan perancangan sistem diterapkan penggunaan *finite state automata* untuk melakukan validasi pada sistem pembelian tiket bioskop. Implementasi dan pengujian sistem pada tahap ini dilakukan implementasi sistem ke dalam bentuk program dan dilakukan pengujian sistem dari implementasi sistem yang sudah dibuat, apakah sudah sesuai dengan konsep pembelian tiket film bioskop menggunakan *finite state automata*, jika masih terdapat kesalahan maka dilakukan perbaikan sehingga mendapatkan hasil yang lebih baik. Penulisan laporan pada tahap ini membuat laporan terkait hasil dari pengujian yang telah dilakukan.



**Gambar 3** Proses Pembelian Tiket Bioskop

Gambar 3 merupakan rancangan proses pembelian tiket bioskop. Rancangan proses ini merupakan alur yang digunakan untuk membuat sistem hingga menghasilkan *output* terakhir yaitu cetak tiket. Dalam perancangan

diagram pembelian tiket bioskop digunakan *Non Deterministic Finite Automata* (N-DFA).



**Gambar 4** Diagram FSA untuk Pembelian Tiket Bioskop.

#### 4. Hasil dan Pembahasan

Berdasarkan hasil analisis pembelian tiket melalui diagram *finite state automata*, maka dapat dibuat sebuah rancangan algoritma yang dapat dijadikan dasar pembuatan program pembelian tiket bioskop. Algoritma proses pembelian tiket bioskop adalah sebagai berikut:

1. Tampilkan pertama yaitu menu daftar film, harga tiket dan waktu *show*
2. Tampilkan kedua yaitu *form* pilih kursi
3. Jika berhasil memilih kursi maka akan masuk ke menu pembayaran
4. Jika gagal memilih kursi maka kembali ke menu daftar film
5. Tampilkan ketiga yaitu menu pembayaran tiket
6. Jika gagal membayar tiket maka akan kembali ke menu daftar film
7. Jika berhasil membayar tiket maka tiket akan tercetak

Algoritma proses pembelian tiket bioskop dibuat berdasarkan pada rancangan *Flowchart Non Deterministic Finite Automata* (N DFA) yang telah dibuat sebelumnya. Algoritma tersebut berupa langkah-langkah untuk menyelesaikan masalah berupa sistem pembelian tiket bioskop. Algoritma ini akan digunakan untuk mempermudah penyusunan *pseudocode* sebelum membuat ke dalam bentuk program. *Pseudocode* dari proses pembelian tiket bioskop adalah sebagai berikut :

```

INPUT USERNAME
READ USERNAME
INPUT PASSWORD
READ PASSWORD
if user benar memasukkan username dan password
  then tampil daftar film

if user pilih waktu show 1
  then tampil form pilih kursi
  if kursi telah dipilih
    then tampilan menu pembayaran
    if transaksi berhasil
      then cetak tiket

else if user pilih waktu show 2
  then tampil form pilih kursi
  if kursi telah dipilih
    then tampilan menu pembayaran
    if transaksi berhasil
      then cetak tiket
  
```

```

else if user pilih waktu show 3
    then tampil form pilih kursi
    if kursi telah dipilih
        then tampilan menu pembayaran
    if transaksi berhasil
        then cetak tiket
end.

```

### Kode Program 1 Perintah untuk Fungsi Menyimpan Data Tempat Duduk yang Dipilih.

```

private void btn_EnterActionPerformed(java.awt.event.ActionEvent evt) {
// fungsi tombol enter
if (!tampungSeat.isEmpty()) {
int harga = konversiKeInt(hrg);
panel_Pembayaran.setVisible(true); //tampilkan panel pembayaran
panel_Seat.setVisible(false); //tampilkan panel pemilihan kursi
lbl_JudulFilm.setText(lbl_Layar.getText());
lbl_Seat.setText(tampungSeat.toString()); //menampilkan data kursi yang dipesan
lbl_Jumlah.setText(String.valueOf(tampungSeat.size())); //menampilkan jumlah data kursi
double r = harga*tampungSeat.size(); //konversi total ke tipe data double
lbl_Total.setText(KonversiKeRP(r));
} else {
JOptionPane.showMessageDialog(null, "ANDA BELUM MEMILIH KURSI", "INFORMASI", JOptionPane.WARNING_MESSAGE);
}
}

```

Kode Program 1 menjelaskan tentang proses validasi pemesanan kursi. Apabila kursi telah dipilih maka akan masuk ke proses pembayaran, dan jika kursi belum terpilih maka akan tampil pesan pemberitahuan user belum memilih kursi.

### Kode Program 2 Untuk melakukan proses pembayaran.

```

1100 private void txt_BayarFocusGained(java.awt.event.FocusEvent evt) {
1101 // TODO add your handling code here:
1102 Action action = new AbstractAction()
1103 {
1104 @Override
1105 public void actionPerformed(ActionEvent e)
1106 {
1107 int totalHarga = konversiKeInt(lbl_Total.getText());
1108 int dibayar = Integer.parseInt(txt_Bayar.getText());
1109 if (dibayar >= totalHarga) {
1110 double kembalian = dibayar-totalHarga;
1111 lbl_Kembalian.setText(KonversiKeRP(kembalian));
1112 btn_CetakTkt.setEnabled(true);
1113 txt_Bayar.setEnabled(false);
1114 } else {
1115 JOptionPane.showMessageDialog(null, "Yang Dibayar Kurang Dari Total Harga");
1116 }
1117 }
1118 };
1119 txt_Bayar.addActionListener( action );
1120 }
1121 }
1122

```

Kode Program 2 merupakan proses penghitungan pembayaran. Jika uang yang diinputkan kurang maka sistem akan memberi pemberitahuan berupa pesan “Yang dibayar kurang dari total harga”.

### Kode Program 3 Perintah Untuk Mencetak Tiket.

```
ArrayList data;
private void cetakTiket() throws FileNotFoundException, IOException{
    File fout = new File(FILENAME);
    FileOutputStream fos = new FileOutputStream(fout);
    try (BufferedWriter bw = new BufferedWriter(new OutputStreamWriter(fos))) {
        int i = 1;
        for (Object tampungSeat1 : tampungSeat) {
            System.out.println();
            bw.write("Tiket Ke-" + i++);
            bw.newLine();
            bw.write("=====");
            bw.newLine();
            bw.write("Judul : "+lbl_JudulFilm.getText());
            bw.newLine();
            bw.write("Studio : "+lbl_Studio.getText());
            bw.newLine();
            bw.write("Tanggal : "+lbl_Tanggal.getText());
            bw.newLine();
            bw.write("Waktu : "+lbl_Waktu.getText());
            bw.newLine();
            String kursi = String.valueOf(tampungSeat1);
            bw.write("Row : "+kursi.charAt(0));
            bw.newLine();
            bw.write("Seat : "+kursi.charAt(1));
            bw.newLine();
            bw.write("=====");
            bw.newLine();
            bw.newLine();
            bw.newLine();
            bw.newLine();
        }
        File f = new File("TiketXXI.txt");
        if(f.exists()) {
            JOptionPane.showMessageDialog(null, "Tiket Berhasil Dicetak!");
            Form_Menu fm = new Form_Menu();
            dispose();
            fm.setVisible(true);
        }
        else{
            JOptionPane.showMessageDialog(null, "Tiket Gagal Dicetak!");
            Form_Menu fm = new Form_Menu();
            dispose();
            fm.setVisible(true);
        }
    }
}
```

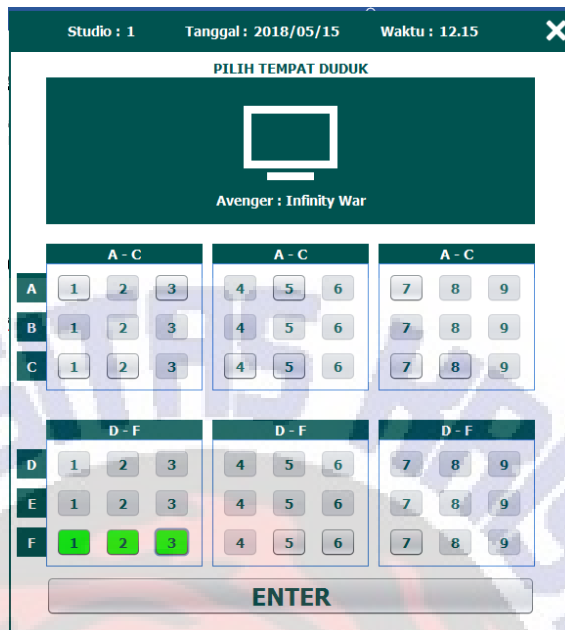
Kode Program 3 menjelaskan pencetakan tiket. Jika proses pembayaran telah berhasil maka akan muncul pesan “Tiket Berhasil Dicetak”. Jika proses pembayaran gagal maka akan muncul pesan “Tiket gagal dicetak”.



STUDIO	JUDUL FILM	HARGA TIKET	SHOW 1	SHOW 2	SHOW 3
1	Avenger : Infinity War	Rp.50.000,00	12.15	15.15	19.15
2	Fifty Shades of Darker	Rp.45.000,00	12.30	15.30	19.30

Gambar 5 Tampilan Halaman Awal Menu Daftar Film Bioskop

Gambar 5 merupakan tampilan awal menu daftar film yang diputar. Pada menu ini *user* dapat memilih film melalui jam tayang film tersebut.



**Gambar 6** Tampilan Halaman Pemilihan Tempat Duduk

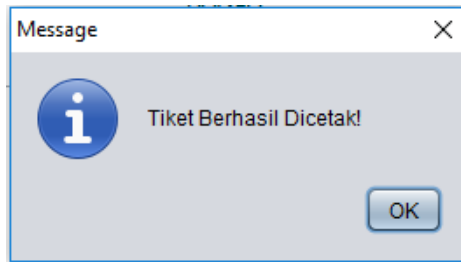
Gambar 6 merupakan tampilan *output* program saat melakukan pemilihan tempat duduk, dimana *user* dapat memilih posisi tempat duduk yang diinginkan.



**Gambar 7** Tampilan Menu Pembayaran

Gambar 7 merupakan tampilan *output* program menu pembayaran dimana menampilkan jumlah tiket yang dibeli dan total harga dari transaksi pembelian tersebut.





**Gambar 8** Tampilan Pesan Jika Tiket Berhasil Dicetak

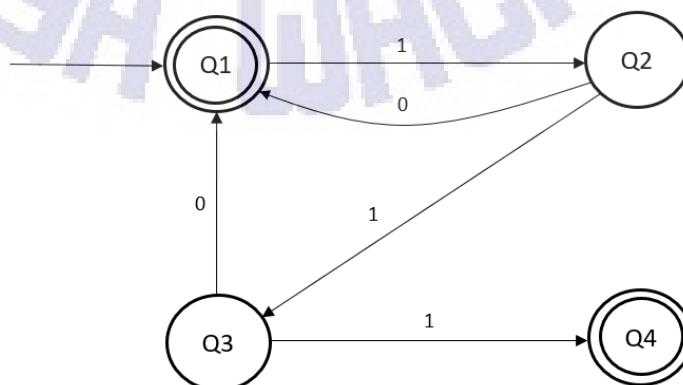
Gambar 8 merupakan tampilan *output* pesan pemberitahuan jika tiket berhasil dicetak.



**Gambar 9** Tampilan *Output* Tiket Yang Tercetak

Gambar 9 merupakan tampilan hasil tiket yang tercetak, yang menampilkan judul, studio, tanggal, waktu dan tempat duduk dari film yang dipilih.

Sistem yang dihasilkan untuk pembelian tiket bioskop, divalidasi menggunakan diagram *finite automata* N-DFA. Diagram *state* N-DFA dapat dilihat pada Gambar 10.



**Gambar 10** Diagram *State* N-DFA untuk Validasi Sistem Pembelian Tiket Bioskop

Pada diagram N-DFA untuk pembelian tiket bioskop terdapat abjad {0,1} simbol, 0 untuk tidak berhasil atau salah dan 1 untuk berhasil atau benar.

Simbol Q1-Q4 merupakan *state* yang ada, *state* tersebut dapat dijelaskan sebagai berikut:

- Q1 = Menu daftar film
- Q2 = Menu pilih kursi
- Q3 = Menu pembayaran tiket
- Q4 = Cetak tiket

Secara formal N-DFA pada Gambar 10 dinyatakan dalam 4 tupel atau M=(Q,  $\Sigma$ ,  $\delta$ , S, F) artinya:

Q= himpunan *state* / kedudukan

$\Sigma$ = himpunan simbol *input* / masukan / abjad

$\Delta$ = relasi transisi

S= *state* awal / kedudukan awal (*initial state*)

F= himpunan *state* akhir (*final state*)

Maka untuk menyatakan Gambar 10 adalah sebagai berikut:

Q = {Q1, Q2, Q3, Q4}

$\Sigma$  = {0, 1}

S = {Q1}

F = {Q4}

$\Delta$  = relasi transisi dapat dilihat pada Tabel 2.

**Tabel 2 Relasi Transisi**

$\Delta$	0	1
Q1	$\emptyset$	Q2
Q2	$\emptyset$	Q3
Q3	Q1	Q4
Q4	$\emptyset$	$\emptyset$

Relasi transisi juga dapat ditulis dengan cara:

$\Delta = \{((Q1, 0), \emptyset), ((Q1, 1), Q2), ((Q2, 0), \emptyset), ((Q2, 1), Q3), ((Q3, 0), Q1), ((Q3, 1), Q4), ((Q4, 0), \emptyset), ((Q4, 1), \emptyset)\}$ .

Pengujian ini dilakukan dengan mencocokkan hasil perancangan sistem yang sudah menerapkan *finite state automata*. Data yang diuji berjumlah 10 sampel data. Nilai keakuratan sistem memiliki dua bobot nilai yaitu 0 dan 1. Bernilai 0 jika hasil akhir sistem tidak sesuai dengan perancangan dan bernilai 1 jika hasil akhir sistem sesuai dengan perancangan.

**Tabel 3 Perbandingan Hasil Perancangan Sistem**

No	Perancangan	Sistem	Nilai keakuratan
1	Tiket berhasil dicetak	Tiket berhasil dicetak	1
2	Tiket berhasil dicetak	Tiket berhasil dicetak	1
3	Tiket berhasil dicetak	Tiket berhasil dicetak	1
4	Tiket tidak berhasil dicetak	Tiket tidak berhasil dicetak	1
5	Tiket berhasil dicetak	Tiket berhasil dicetak	1
6	Tiket berhasil dicetak	Tiket berhasil dicetak	1
7	Tiket berhasil dicetak	Tiket berhasil dicetak	1
8	Tiket berhasil dicetak	Tiket berhasil dicetak	1
9	Tiket tidak berhasil dicetak	Tiket tidak berhasil dicetak	1
10	Tiket berhasil dicetak	Tiket berhasil dicetak	1
Jumlah			10

Jika dihitung nilai probabilitasnya, maka:

$$\begin{aligned}\text{Nilai Akurasi} &= \frac{\text{Jumlah data akurat}}{\text{Jumlah seluruh data}} \times 100\% \\ &= \frac{10}{10} \times 100\% \\ &= 100\%\end{aligned}$$

Berdasarkan hasil pengujian tersebut dapat disimpulkan bahwa akurasi sistem pembelian tiket bioskop menggunakan *finite state automata* dengan 10 data yang diuji adalah 100% yang menunjukkan bahwa sistem ini dapat berfungsi dengan baik sesuai hasil perancangan sistem.

## 5. Simpulan

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat diambil kesimpulan bahwa : (1) Validasi yang dilakukan menggunakan *finite state automata* dapat membuktikan alur logis dari sistem pembelian tiket bioskop; (2) Dengan adanya perancangan sistem pembelian tiket bioskop ini dapat mempermudah user dalam melakukan pembelian tiket; (3) Program dirancang menggunakan *Java Netbeans*.

## 6. Daftar Pustaka

- [1] Paliyama, D. G., Pakereng, M. A. I., Oktober 2017. Validasi Aplikasi Diagnosa Penyakit Kanker Serviks Menggunakan *Finite State Automata*.
- [2] Manalu, Y. T., Pakereng, M. A. I., Mei 2017. Penerapan *Finite State Automata* Pada Layanan Reservasi Tiket Bis.
- [3] E.P, Rizky Indah Melly., Wamiliana., Kurniawan, Didik., Desember 2012. Penerapan Konsep *Finite State Automata* (FSA) Pada Mesin Pembuat Kopi Otomatis.
- [4] Utdirartatmo, F., 2001. *Teori Bahasa dan Otomata*. Yogyakarta: J & J Learning.