

RANCANG BANGUN APLIKASI TRANSLITERASI AKSARA LATIN MENJADI AKSARA SASAK MENGGUNAKAN ALGORITMA RULE BASED BERBASIS ANDROID

(Implementation Latin Script Transliteration to Sasak Script Using Rule Base Algorithm On Android)

Putu Wahyu Pratama, Arik Aranta, Fitri Bimantoro

Program Studi Teknik Informatika, Fakultas Teknik, Universitas Mataram
Jl. Majapahit 62, Mataram, Lombok NTB, INDONESIA

Email: putuwahyoepatama@gmail.com, [arikaranta, bimo]@unram.ac.id

Abstract

The Sasak script is one of the cultural heritage of the Sasak people on the island of Lombok. But lately, the Sasak script has begun to be rarely used among the Sasak people, especially among the younger generation. If this continues, the existence of the Sasak script is threatened with extinction. One of the efforts in preserving the Sasak script is digitizing the Sasak script to digital technology devices by utilizing Unicode. In this research, the author will make an android-based transliteration application and develop a rule-based algorithm that refers to the rules of the Sasak script. The string replacement method with hexadecimal addressing derived from the UTF-16 Unicode is used to adjust the Latin characters with the writing rules of the Sasak script. Digital writing of the Sasak script using 47 symbols from Unicode Balinese block on "1B00-1B7F". To transliterate Latin script into the Sasak script, the author has developed 1704 rule-based rules to apply all the rules of writing the Sasak script. The accuracy test towards on 3 data sources, namely testing of street signage in Mataram City obtained an accuracy value of 97.20% from 250 data test, testing of "Bali-Indonesian Latin and Balinese script" dictionary obtained an accuracy value of 96,375 from 800 data test, and testing of Lontar Doyan Neda (Dewi Anjani) obtained an accuracy value of 65.33% from 600 data test. So that the overall accuracy obtained is 85.39% from 1650 data test.

Keywords: aksara Sasak, transliterasi, rule based, string replacement, unicode

1. PENDAHULUAN

Aksara Sasak merupakan salah satu bentuk warisan budaya masyarakat suku Sasak yang perlu untuk dilestarikan. Dahulu masyarakat suku Sasak di Pulau Lombok menggunakan aksara Sasak dalam menuliskan informasi dalam bentuk tulisan. Namun belakangan ini, penggunaan dan pemanfaatan aksara Sasak dimasyarakat mulai berkurang terutama dikalangan generasi muda. Kurangnya pemanfaatan aksara Sasak di masyarakat, dikhawatirkan aksara Sasak akan mengalami kepunahan. Pelestarian aksara Sasak di masyarakat merupakan sesuatu yang perlu untuk diperhatikan. Bukan hanya sebagai ciri khas budaya suku Sasak, tetapi juga sebagai bagian dari identitas masyarakat suku Sasak itu sendiri. Mengingat pentingnya upaya pelestarian aksara Sasak, maka dibutuhkan inovasi dan upaya-upaya dalam melestarikan aksara Sasak untuk mencegah kepunahan.

Ditengah perkembangan teknologi digital saat ini, salah satu upaya pelestarian aksara Sasak dapat dilakukan dengan digitalisasi aksara dengan menerapkan aksara Sasak pada perangkat teknologi digital. Penerapan aksara Sasak pada perangkat teknologi digital dapat dilakukan dengan memanfaatkan *unicode* yang dikembangkan oleh *Unicode Consortium*. *Unicode* merupakan standar pengkodean universal untuk karakter tulisan dan teks yang mengizinkan suatu aksara atau karakter dapat ditampilkan secara digital[1]. Aksara Sasak dapat menggunakan blok *unicode balinese* pada "1B00-1B7F"[2]. Hal ini dapat dilakukan karena kekerabatan dan bentuk aksara yang sama antara aksara Sasak dengan aksara Bali. Namun perbedaannya terdapat pada penerapan aturan yang digunakan dalam penulisan aksara. Dengan diterapkannya aksara Sasak dalam bentuk *unicode*, masyarakat dapat memanfaatkan aksara Sasak pada perangkat teknologi digital untuk keperluan seperti teks dalam foto, teks

dalam video ataupun pesan teks dalam bermedia sosial sehingga meningkatkan pemanfaatan aksara Sasak ditengah masyarakat.

Melalui penelitian sebelumnya tentang penggunaan aksara Sasak dalam keseharian masyarakat, diketahui bahwa salah satu faktor kurangnya pemanfaatan aksara Sasak adalah kurangnya pengetahuan masyarakat tentang aksara Sasak karena aksara Sasak susah untuk dipelajari[3]. Seperti yang kita ketahui, aksara Sasak memiliki aturan-aturan dalam menyusun tulisannya, sehingga masyarakat yang ingin menulis dengan aksara Sasak dituntut untuk memahami aturan penulisan aksara Sasak. Bagi masyarakat yang belum memiliki pengetahuan tetapi ingin menulis dengan aksara Sasak, menuliskan kata atau kalimat dengan aksara Sasak tentu menjadi problem yang tidak mudah. Oleh karena itu, transliterasi dibutuhkan untuk menyelesaikan masalah tersebut. Transliterasi dilakukan dalam rangka untuk mengubah aksara Latin menjadi aksara Sasak. Beberapa daerah lain di Indonesia seperti Bima, Bali dan Jawa telah mengembangkan aplikasi transliterasi untuk melestarikan aksara daerahnya[4]–[7]. Sementara untuk saat ini, aksara Sasak belum memiliki aplikasi transliterasi yang dapat mengubah aksara Latin menjadi aksara Sasak ataupun sebaliknya.

Berdasarkan permasalahan yang ada, maka peneliti bermaksud melakukan perancangan dan penelitian tentang aplikasi transliterasi berbasis *android* yang dapat melakukan alih aksara dari aksara Latin ke aksara Sasak. *Android* dipilih sebagai platform aplikasi ini berdasarkan tingkat kepopuleran *android* yang memiliki lebih dari 2,5 miliar pengguna dari seluruh dunia[8]. Dalam penelitian ini, algoritma *rule based* digunakan dalam menerapkan aturan penulisan aksara Sasak menggunakan metode *string replacement*. Dengan penerapan algoritma *rule based*, diharapkan nantinya aplikasi yang dihasilkan mampu melakukan transliterasi dengan akurasi yang baik.

2. TINJAUAN PUSTAKA

Penelitian-penelitian sebelumnya yang dijadikan sebagai rujukan dalam proses pembuatan aplikasi transliterasi aksara Latin menjadi aksara Sasak dengan algoritma *rule based* dapat diuraikan sebagai berikut.

Penelitian pertama yang berjudul “Penerapan Algoritma *Rule Base* Dengan Pendekatan *Hexadesimal* Pada Transliterasi Aksara Bima Menjadi Huruf Latin”. Penelitian ini menerapkan metode *string replacement* dalam mengubah dan menyesuaikan karakter aksara Bima menjadi aksara Latin. Dari penelitian yang

dilakukan memperoleh akurasi sebesar 90,64% dari 171 aturan yang diterapkan[5].

Penelitian kedua berjudul “*Transliteration Balinese Latin Text Becomes Aksara Bali Using Rule Base and Levenshtein Distance Approach*”. Penelitian ini dilakukan guna untuk memberikan pengalaman menulis lontar dengan menggunakan aplikasi (*software*). Pendekatan *Levenshtein Distance* digunakan untuk membandingkan kesesuaian kata yang di-*input*-kan dengan kata yang ada di dalam sistem *database* transliterasi. Metode *rule based* digunakan untuk menyesuaikan kata atau kalimat masukkan sehingga dapat sesuai dengan aturan yang ada untuk diubah ke dalam aksara Bali. Pengujian dilakukan dengan menggunakan 20 lembar lontar bertuliskan aksara Bali dan hasilnya memperoleh akurasi rata-rata sebesar 99,09%[6].

Penelitian ketiga berjudul “Transliterasi Huruf Latin Bahasa Indonesia ke Aksara Jawa Menggunakan Metode *Snakecut*”. Penelitian ini mengubah citra tulisan Latin berbahasa Indonesia menjadi teks berbahasa dan beraksara Jawa. Citra tulisan latin dikenali dan diubah ke dalam teks beraksara Latin dengan metode *snakecut* untuk kemudian diubah ke dalam teks beraksara Jawa. Hasil dari pengujian yang dilakukan diperoleh persentase keberhasilan sebesar 85% dari 20 sampel uji[7].

Penelitian keempat berjudul “Penerapan *Rule Based* dalam Membangun *Transliterator Jawatex*”. Penelitian ini melakukan transliterasi dokumen teks dengan aksara Latin menjadi aksara Jawa. Dokumen teks Latin diolah menjadi daftar pola pemenggalan *string* Latin yang lebih kecil menggunakan metode *rule based*. Kemudian hasil pemenggalan *string* diolah ke dalam skema pemetaan ke dalam aksara jawa menggunakan metode *Pattern Matching*. Dalam proses transliterasi menggunakan 177 model pola pemenggalan *string* Latin dan 138 model pola pemetaan. Hasil dari pengujian transliterasi ini yaitu teks Latin dapat ditransliterasikan menjadi teks beraksara jawa dengan baik dan disimpan dalam bentuk dokumen PDF[9].

Penelitian kelima berjudul “Sistem Penerjemah Bahasa Jawa-Aksara Jawa Berbasis *Finite State Automata*”. Dalam penelitian ini transliterasi dilakukan dengan mengubah teks berbahasa Jawa dengan aksara Latin menjadi teks beraksara Jawa dan sebaliknya. Proses transliterasi menggunakan *finite state automata* (FSA) untuk mengenali dan mencari pola suku kata. FSA menentukan jenis setiap huruf dalam bahasa Jawa sehingga mempermudah konfigurasi penyusunan aksara Jawa yang dihasilkan.

Dalam proses transliterasi aksara Latin ke aksara Jawa diperoleh persentase keberhasilan sebesar 92% dari 50 sampel uji. Sedangkan proses transliterasi aksara Jawa ke aksara Latin memperoleh persentase keberhasilan sebesar 93,8% dari 32 sampel uji[10].

Penelitian keenam berjudul *“Utilization of Hexadecimal Numbers In Optimization of Balinese Transliteration String Replacement Method”*. Penelitian ini mengubah aksara Latin menjadi aksara Bali dengan menggunakan pendekatan *string replacement* berbasis *android*. Dalam penelitian ini setiap huruf dari kata yang di-*input*-kan diubah ke dalam bilangan *hexadesimal* untuk memperoleh alamat uniknnya. Berdasarkan hasil pengujian, diperoleh persentase ketelitian sebesar 93% dari 151 data uji dan 92,72% dari 3915 data uji yang diambil dari Kamus Bali-Indonesia[11].

Penelitian ketujuh berjudul *“Balinese Latin Text Becomes Aksara Bali Using Rule Base Method”*. Aplikasi transliterasi yang dihasilkan dalam penelitian ini mampu mengubah teks Latin menjadi aksara Bali seperti pada dokumen lontar asli dengan akurasi mencapai 90,67%. Pengujian dilakukan dengan mencocokkan hasil transliterasi dengan 10 dokumen lontar beraksara Bali[12].

Penelitian kedelapan berjudul *“Pengembangan Aplikasi Mobile Pembelajaran Transliterasi Tulisan Latin Ke Aksara Bali”*. Aplikasi android yang dihasilkan dalam penelitian ini mampu menangani perilaku kompleks transliterasi berdasarkan dokumen *“The Balinese Alphabet”* dengan tingkat akurasi mencapai 98% dari 151 sampel uji[13].

Berdasarkan tinjauan pustaka diketahui bahwa terdapat beberapa penelitian yang memiliki keterkaitan dengan pembuatan aplikasi transliterasi, namun belum ada penelitian tentang transliterasi aksara Latin menjadi aksara Sasak. Maka dalam hal ini, penulis melakukan penelitian untuk membuat aplikasi transliterasi aksara Latin menjadi aksara Sasak menggunakan algoritma *rule based* dengan memanfaatkan bilangan heksadesimal sebagai pengalamanan karakter dalam menyesuaikan aturan penulisan aksara Sasak dengan aksara Latin. Penelitian ini mengimplementasikan aksara *rekan* dan variasi penulisan aksara Sasak yang tidak digunakan pada transliterasi aksara Bali dalam penelitian sebelumnya[11]. Sehingga kata-kata dalam bahasa Sasak yang akrab dengan penggunaan huruf “q” pada akhir kata dapat diimplementasikan.

2.1. Aksara Sasak

2.1.1. Aksara *baluq olas*

Aksara Sasak merupakan aksara yang digunakan untuk menulis dan menyampaikan informasi oleh masyarakat suku Sasak pada masa lampau. Aksara Sasak menggunakan sistem penulisan *abugida*. *Abugida* adalah sistem aksara yang huruf-hurufnya berupa suku kata (silabel) dengan vokal bawaan huruf “a”[1]. Aksara utama dalam aksara Sasak biasanya dikenal dengan aksara *Baluq Olas* atau *Wiyanjana*. Aksara Sasak memiliki hubungan kekerabatan dengan aksara Bali dan aksara Jawa karena sama-sama berasal dari aksara Kawi yang berkembang di Pulau Jawa. Berbeda dengan aksara Jawa yang memiliki 20 buah aksara utama, aksara Sasak dan aksara Bali memiliki 18 buah aksara utama. Aksara Sasak menghilangkan aksara “*dha*” dan “*tha*” pada aksara Jawa sebagai aksara utama[14]. Macam-macam bentuk aksara utama dalam aksara Sasak dapat dilihat pada Gambar 1.

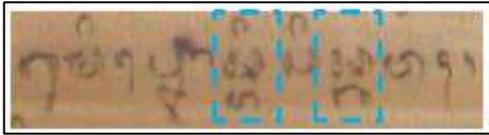
Ha ᮊ / ᮃ	Na ᮊ / ᮄ	Ca ᮊ / ᮅ	Ra ᮊ / ᮆ	Ka ᮊ / ᮇ
Da ᮊ / ᮈ	Ta ᮊ / ᮉ	Sa ᮊ / ᮏ	Wa ᮊ / ᮐ	La ᮊ / ᮑ
Ma ᮊ / ᮒ	Ga ᮊ / ᮓ	Ba ᮊ / ᮔ	Nga ᮊ / ᮕ	
Pa ᮊ / ᮖ	Ja ᮊ / ᮗ	Ya ᮊ / ᮘ	Nya ᮊ / ᮙ	

Gambar 1. Aksara *Baluq Olas*.

2.1.2. Aksara serapan (*rekan*)

Aksara Sasak memiliki hubungan kekerabatan paling dekat dengan aksara Bali. Mulai dari bentuk aksara hingga sistem penulisan aksara Sasak hampir sama dengan aksara Bali. Namun, aksara Sasak dan aksara Bali memiliki perbedaan mendasar dalam sistem penulisannya. Pertama dari segi bahasa, penggunaan bahasa Sasak yang akrab dengan penggunaan huruf “q” pada akhir kata membuat penulisan aksara Sasak berbeda dengan aksara Bali. Aksara Bali tidak mengenal huruf “q” dalam penulisan aksaranya. Berikut ini merupakan contoh perbedaan penulisan aksara Sasak dengan aksara Bali menggunakan kalimat *“gumi pnoq isiq gawah,”* yang diambil dari Lontar Doyan Neda (Dewi Anjani). : *“gumi pnoq isiq gawah,”*.

Penulisan aksara Sasak dalam Lontar Doyan Neda (Dewi Anjani):



Gambar 2. Penulisan pada Lontar Doyan Neda (Dewi Anjani).

Hasil transliterasi aksara Bali melalui aplikasi “Nulis Aksara Bali”:



Gambar 3. Contoh penulisan aksara Bali.

Kemudian yang kedua karena adanya pengaruh budaya Arab yang masuk dan berkembang di Lombok pada abad ke-16[15], membuat masyarakat suku Sasak melakukan penyesuaian dalam sistem penulisannya. Huruf-huruf yang biasa ditemukan dalam bahasa Arab seperti “f”, “v”, “q”, “z” yang tidak ada dalam aksara Bali dapat ditulis dalam aksara Sasak dengan menggunakan aksara *rekan*. Hal ini menunjukkan bahwa aksara memiliki sifat yang dinamis dan selalu berkembang.

TABEL I. BENTUK AKSARA SERAPAN.

Nama	Penggunaan Unicode	Aksara	Gantungan
Kha / Qa	U+1B13 (ka) dan U+1B34 (<i>rekan</i>)	ꦏꦲ	ꦏꦲꦠꦺ
Dza	U+1B24 (da) dan U+1B34 (<i>rekan</i>)	ꦢꦲ	ꦢꦲꦠꦺ
Fa	U+1B27 (pa) dan U+1B34 (<i>rekan</i>)	ꦥꦲ	ꦥꦲꦠꦺ
Va	U+1B2F (wa) dan U+1B34 (<i>rekan</i>)	ꦮꦲ	ꦮꦲꦠꦺ
Za	U+1B1A (ja) dan U+1B34 (<i>rekan</i>)	ꦗꦲ	ꦗꦲꦠꦺ
Gha	U+1B15 (Ga) dan U+1B34 (<i>rekan</i>)	ꦒꦲ	ꦒꦲꦠꦺ
'a	U+1B15 (akara) dan U+1B44 (<i>paten</i>)	ꦲꦶ	

Di masyarakat Sasak beberapa penulisan aksara serapan dapat dilihat pada penulisan papan nama jalan di Kota Mataram. Contoh penggunaan aksara serapan dapat dilihat pada Gambar 2.



Gambar 4. Papan nama jalan di Kota Mataram.

2.1.3. Sandangan

a. *Sandangan Swara*

Sandangan swara digunakan untuk mengubah pelafalan dengan bunyi vokal dari suatu aksara utama. Seperti yang diketahui, aksara Sasak menggunakan sistem penulisan *abugida* yang menggunakan kombinasi huruf konsonan yang diikuti dengan huruf vokal “a”. Dalam aksara Sasak *sandangan swara* terdiri dari 5 huruf vokal yaitu “u”, “i”, “e”, “ē” dan “o”. Terdapat juga aksara khusus yang digunakan untuk menuliskan suku kata “re” dan “le”. Bentuk dari sandangan swara dapat dilihat pada Tabel II.

TABEL II. SANDANGAN SWARA.

Bunyi	Nama	Unicode	Aksara
u	<i>Suku</i>	U+1B38	ꦱꦸ
i	<i>Wulu</i>	U+1B36	ꦱꦸꦭꦸ
e	<i>Pepet</i>	U+1B42	ꦱꦺꦥꦺꦠ
ē	<i>Taleng</i>	U+1B3E	ꦱꦺꦠꦺꦁ
o	<i>Taleng tarung</i>	U+1B40	ꦱꦺꦠꦺꦁꦠꦂꦸꦁ
re	<i>Re Reppa</i>	U+1B0B	ꦱꦺꦫꦺꦫꦺꦥꦥ
le	<i>Le Lenga</i>	U+1B0D	ꦱꦺꦫꦺꦁ

b. *Sandangan Paten*

Sandangan paten merupakan aksara yang ditambahkan pada aksara utama untuk menghilangkan bunyi atau huruf vokal dari suatu aksara utama. Untuk menjadikan sebuah aksara utama menjadi konsonan, maka dapat menggunakan *paten* (ꦲꦶ). Namun tidak berlaku untuk huruf “r”, “h” dan “ng”, huruf-huruf tersebut memiliki aksara

tersendiri dalam penulisan aksara Sasak. Bentuk *sandangan paten* dapat dilihat pada Tabel III.

TABEL III. SANDANGAN PATEN.

Huruf	Nama	Unicode	Bentuk Aksara
(Konsonan)	<i>Paten</i>	U+1B44	ʔ
h	<i>Wisah</i>	U+1B04	ʔ
r	<i>Layar</i>	U+1B03	ʔ
ng	<i>Cecek</i>	U+1B02	ʔ

2.1.4. Aksara angka

Penulisan angka dalam aksara Sasak harus menggunakan simbol pemisah di awal dan di akhir penulisan angkanya, penulisannya menggunakan simbol *carik siki* (ʔ). Tanda *carik siki* digunakan untuk menghindari kerancuan dalam penulisan aksara, mengingat miripnya bentuk aksara huruf dan angka[14]. Bentuk angka dalam aksara Sasak dapat dilihat pada Tabel IV.

TABEL IV. AKSARA ANGKA.

Angka	Unicode	Bentuk Aksara
0	U+1B50	o
1	U+1B51	ʔ
2	U+1B52	ʔ
3	U+1B53	ʔ
4	U+1B54	ʔ
5	U+1B55	ʔ
6	U+1B56	ʔ
7	U+1B57	ʔ
8	U+1B58	ʔ
9	U+1B59	ʔ

2.1.5. Tanda baca

Beberapa tanda baca yang biasa digunakan dalam penulisan aksara Sasak dapat dilihat pada Tabel 2.11.

TABEL V. AKSARA TANDA BACA.

Tanda	Unicode	Bentuk Aksara
Koma	U+1B5E	ʔ
Titik	U+1B5F	ʔ

2.2. Algoritma rule based

Algoritma *rule based* (*rule based system*) merupakan suatu program komputer yang memproses informasi yang terdapat di dalam *working memory* dengan sekumpulan aturan yang terdapat di dalam basis pengetahuan menggunakan mesin inferensi untuk menghasilkan informasi baru. *Rule based system* secara relatif adalah model sederhana yang bisa diadaptasi ke banyak masalah[16].

Hal yang harus dimiliki dalam membangun sebuah sistem berbasis aturan:

- Sekumpulan fakta yang berupa keadaan yang relevan dengan keadaan awal sistem bekerja.
- Sekumpulan aturan yaitu setiap kondisi yang mempengaruhi tindakan yang diambil dalam ruang lingkup permasalahan yang dibutuhkan.
- Kondisi yang menentukan bahwa solusi telah ditemukan atau tidak (*none exist*).

Algoritma *rule based* menggunakan teknik sederhana, proses dimulai dengan aturan dasar yang berisi semua pengetahuan dari suatu permasalahan yang dihadapi kemudian dikodekan ke dalam sebuah aturan IF-THEN dan sebuah basis data (tempat penyimpanan) yang mengandung data, pernyataan dan informasi awal. Sistem memeriksa semua aturan kondisi (IF) yang menentukan *subset*. Jika sebuah aturan kondisi ditemukan, maka sistem akan melakukan kondisi THEN. Pengecekan suatu aturan kondisi ini terus berlanjut hingga salah satu atau dua kondisi bertemu, jika aturan tidak ditemukan maka sistem tersebut harus keluar dari perulangan (*terminate*)[17].

2.3. Pengalamatan heksadesimal

Bilangan heksadesimal merupakan sistem bilangan dengan basis 16 yang biasa digunakan dalam sistem komputer. Bilangan heksadesimal menggunakan digit 0 sampai 9 dan huruf A, B, C, D, E dan F dalam membentuk sistem bilangannya. Keenam huruf ini memiliki harga desimal yaitu A = 10, B = 11, C = 12, D = 13, E = 14 dan F = 15[18]. Pengalamatan dengan bilangan heksadesimal pada proses transliterasi mengadopsi format bilangan *unicode* UTF-16 dengan menghapus \u. Pengalamatan suatu karakter harus terdiri dari 4 digit bilangan heksadesimal dengan jumlah yang sama. Apabila terdapat perbedaan 1 atau lebih jumlah alamat, maka akan mempengaruhi proses transliterasi yang dilakukan.

2.4. Metode string replacement

Metode *string replacement* merupakan suatu metode manipulasi *string* dalam pemrograman yang digunakan untuk mengubah suatu *string* menjadi *string* lainnya. Dalam penerapan algoritma *rule based*, metode *string replacement* digunakan untuk mengimplementasikan aturan-aturan dengan konsep IF-THEN dan merubahnya ke dalam bentuk *subset* (hasil yang diinginkan). Contoh proses penggunaan *string replacement* dapat dilihat pada Gambar 5. Pada proses tersebut terdapat perintah untuk mengganti kata “sulit” menjadi “mudah” pada kalimat *input*-an “Belajar Aksara Sasak itu sulit”. Sehingga kalimat yang dihasilkan berubah menjadi “Belajar Aksara Sasak itu mudah”.

```
01. input = "Belajar Aksara Sasak itu sulit"
02. a = (input.replace("sulit", "mudah"))
03. print("Output = "+a)

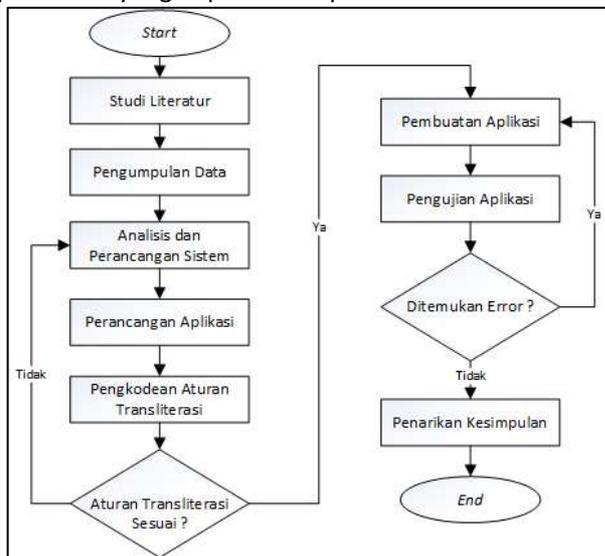
Shell

Output = Belajar Aksara Sasak itu mudah
> |
```

Gambar 5. Penggunaan *String Replacement*.

3. METODE PENELITIAN

Dalam penelitian ini, peneliti menyusun suatu alur penelitian yang dapat dilihat pada Gambar 6.



Gambar 6. Alur penelitian.

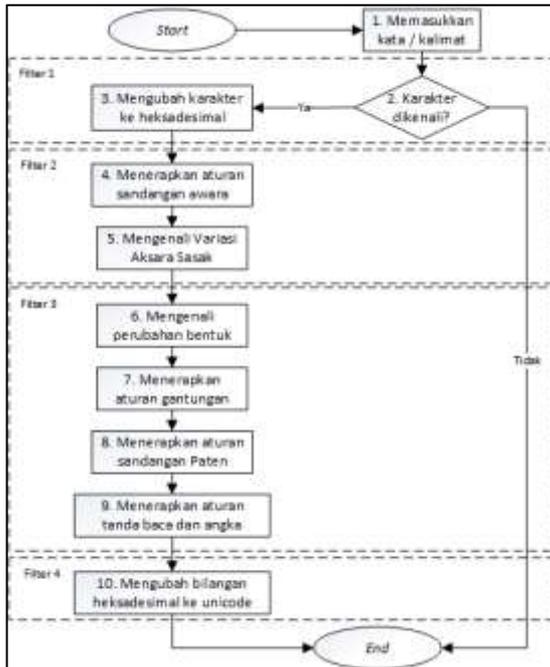
Proses awal penelitian ini adalah dimulai dengan melakukan studi literatur terhadap buku, artikel dan jurnal-jurnal yang berhubungan dengan transliterasi aksara Latin menjadi aksara daerah dalam hal ini adalah aksara Sasak. Kemudian semua data, fakta dan aturan-aturan dalam penulisan aksara Sasak dikumpulkan untuk dapat membangun sebuah sistem

dengan algoritma *rule based*. Dalam penelitian ini, data yang digunakan dalam membangun pengetahuan aturan penulisan aksara Sasak menggunakan 3 sumber data yaitu 250 data papan nama jalan di Kota Mataram, 800 data Kamus Bali - Indonesia beraksara Latin dan Bali, serta 10 lembar lontar Doyan Neda (Dewi Anjani). Kekerabatan aksara Sasak dengan aksara Bali membuat mayoritas aturan penulisan aksara Sasak memiliki kesamaan dengan aturan penulisan aksara Bali. Terdapat perbedaan aturan seperti penggunaan aksara *rekan*, dan penggunaan aksara “*akara*” untuk penulisan huruf “q”. Bentuk aksara Sasak yang memiliki kesamaan dengan aksara Bali membuat penulisan aksara Sasak dapat dilakukan dengan menggunakan aksara Bali. Dalam penulisan aksara Sasak secara digital dapat memanfaatkan *Unicode Balinese* yang berada pada blok *Unicode “1B00-1B7F”*. Data font *Unicode Balinese* dapat dilihat pada Tabel VI.

TABEL VI. DATA *BALINESE UNICODE*.

<i>Balinese Unicode</i>								
	1B0	1B1	1B2	1B3	1B4	1B5	1B6	1B7
0	ᮀ	ᮁ	ᮂ	ᮃ	ᮄ	ᮅ	ᮆ	ᮇ
1	ᮈ	ᮉ	ᮊ	ᮋ	ᮌ	ᮍ	ᮎ	ᮏ
2	ᮐ	ᮑ	ᮒ	ᮓ	ᮔ	ᮕ	ᮖ	ᮗ
3	ᮘ	ᮙ	ᮚ	ᮛ	ᮜ	ᮝ	ᮞ	ᮟ
4	ᮠ	ᮡ	ᮢ	ᮣ	ᮤ	ᮥ	ᮦ	ᮧ
5	ᮨ	ᮩ	᮪	᮫	ᮬ	ᮭ	ᮮ	ᮯ
6	᮰	᮱	᮲	᮳	᮴	᮵	᮶	᮷
7	᮸	᮹	ᮺ	ᮻ	ᮼ	ᮽ	ᮾ	ᮿ
8	ᮿ	ᮾ	ᮽ	ᮼ	ᮽ	ᮾ	ᮿ	ᮾ
9	ᮾ	ᮿ	ᮾ	ᮿ	ᮾ	ᮿ	ᮾ	ᮿ
A	ᮾ	ᮿ	ᮾ	ᮿ	ᮾ	ᮿ	ᮾ	ᮿ
B	ᮾ	ᮿ	ᮾ	ᮿ	ᮾ	ᮿ	ᮾ	ᮿ
C	ᮾ	ᮿ	ᮾ	ᮿ	ᮾ	ᮿ	ᮾ	ᮿ
D	ᮾ	ᮿ	ᮾ	ᮿ	ᮾ	ᮿ	ᮾ	ᮿ
E	ᮾ	ᮿ	ᮾ	ᮿ	ᮾ	ᮿ	ᮾ	ᮿ
F	ᮾ	ᮿ	ᮾ	ᮿ	ᮾ	ᮿ	ᮾ	ᮿ

Dengan beragamnya bentuk aksara dan aturan-aturan dalam penulisan aksara Sasak, kemudian disusun sebuah algoritma *rule based* untuk dapat menyesuaikan aturan-aturan dalam penulisan teks Latin dengan aturan penulisan aksara Sasak.



Gambar 7. Alur sistem dengan algoritma *rule based*.

Proses transliterasi dirancang dengan 4 filter utama untuk dapat mengubah aksara Latin menjadi aksara Sasak. Dimana dalam filter 1 dilakukan pengenalan karakter dan mengubah ke bilangan heksadesimal. Sebuah teks dengan aksara Latin yang dikenali oleh sistem diberikan tanda sesuai dengan Tabel VII.

TABEL VII. PENGENALAN KATAKTER LATIN.

Karakter	Bentuk	Tindakan	Contoh
Konsonan	b, c, d, f, g, h, j, k, l, m, n, p, q, r, s, t, v, w, y, z	Menambahkan tanda "/" di akhir huruf	Huruf "k" menjadi "k/"
Vokal	a, i, u, e, ē, o	Menambahkan tanda "/" di awal huruf	"o" menjadi "/o"
Angka	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	Menambahkan tanda "*" diawal dan diakhir angka	1 menjadi "*1*"
Tanda baca	titik, koma dan spasi	Tetap	Tetap

Proses penambahan karakter seperti tabel VII digunakan untuk membedakan jenis karakter (vokal, konsonan, angka dan tanda baca) serta untuk membentuk banyaknya variasi id unik yang dari setiap karakter sehingga alamat heksadesimal yang dihasilkan menjadi lebih bervariasi dan meminimalisir terjadinya kesamaan alamat saat proses transliterasi.

Setelah pengenalan karakter selesai dilakukan, kemudian diubah menjadi bilangan heksadesimal yang digunakan sebagai alamat selama proses transliterasi dilakukan. Beberapa *sample* dari proses perubahan karakter Latin menjadi bilangan heksadesimal dapat dilihat pada Tabel VIII.

TABEL VIII. SAMPLE PERUBAHAN KARAKTER KE HEKSADESIMAL.

Karakter	Heksadesimal
s/	0073002f
/a	002f0061
g	002a1b59002a
, (koma)	002c

Dalam filter 2, melakukan perubahan alamat berdasarkan suku kata. Mengingat sistem penulisan aksara Sasak yang menggunakan sistem *Abugida*, yaitu sebuah aksara Sasak mewakili 2 huruf (konsonan dan vokal "a") maka pengenalan suku kata penting untuk dilakukan. Beberapa proses perubahan suku kata dengan vokal huruf "a" dalam filter 2 dapat dilihat pada Tabel IX.

TABEL IX. PERUBAHAN AKSARA UTAMA.

Aturan kondisi	Hex sebelum	Operasi Hex	Hex sesudah	Hasil
b//a	0062002f002f0061	diubah	1b29	ba
c//a	0063002f002f0061	diubah	1b18	ca
d//a	0064002f002f0061	diubah	1b24	da
g//a	0067002f002f0061	diubah	1b15	ga
h//a	0068002f002f0061	diubah	1b33	ha
j//a	006a002f002f0061	diubah	1b1a	ja
k//a	006b002f002f0061	diubah	1b13	ka
l//a	006c002f002f0061	diubah	1b2e	la
m//a	006d002f002f0061	diubah	1b2b	ma
n//a	006e002f002f0061	diubah	1b26	na
p//a	0070002f002f0061	diubah	1b27	pa
r//a	0072002f002f0061	diubah	1b2d	ra
s//a	0073002f002f0061	diubah	1b32	sa
t//a	0074002f002f0061	diubah	1b22	ta
w//a	0077002f002f0061	diubah	1b2f	wa
y//a	0079002f002f0061	diubah	1b2c	ya
n/ga	006e002f1b15	diubah	1b17	nga
n/ya	006e002f1b2c	diubah	1b1c	nya
n/g/	006e002f0067002f	006e+0067	00d5002f	ng/
n/y/	006e002f0079002f	006e+0079	00e7002f	ny/

Melalui Tabel IX, terjadi perubahan alamat dari alamat heksadesimal karakter aksara Latin menjadi alamat heksadesimal yang baru (*hex* setelah). Alamat "*hex* setelah" didapatkan dari bilangan heksadesimal aksara utama dari aksara Bali pada Tabel VI.

4.2.1. Pengujian akurasi

Pengujian terhadap kinerja aplikasi dilakukan dengan menghitung nilai akurasi dari kesesuaian hasil transliterasi dengan data uji yang digunakan. Pengujian akurasi dilakukan terhadap 3 sumber data yang dapat dilihat pada Tabel XIV.

TABEL XIV. HASIL PENGUJIAN AKURASI.

Sumber Data	Kasus	Error	Persentase Berhasil
Papan nama jalan beraksara Sasak di Kota Mataram	250	7	97,20 %
Kamus Bali - Indonesia beraksara Latin dan Bali	800	29	96,375 %
10 lembar Lontar Doyan Neda Dewi Anjani	600	208	65,33 %
Total	1200	244	85,21 %

Berdasarkan hasil pengujian transliterasi dengan 1650 data uji, terdapat sebanyak 244 data uji yang tidak sesuai dengan hasil transliterasi aplikasi "Transliterasi Aksara Sasak" yang dibangun. Sehingga diperoleh akurasi dengan perhitungan dengan persamaan (1).

$$\text{Akurasi} = \frac{(1650-244)}{1650} \times 100\% = 85,21\% \quad (1)$$

Akurasi yang diperoleh dari aplikasi "Transliterasi Aksara Sasak" sebesar 85,21% dari 1650 data uji. Akurasi tidak dapat mencapai 100% dikarenakan beberapa hal sebagai berikut:

- Kesalahan penulisan aksara pada beberapa nama jalan.
- Penulisan huruf "e" dan "ē" yang dituliskan sama pada huruf Latin padahal memiliki aksara yang berbeda.
- Perubahan bentuk vokal yang beragam variasi. Misalkan terdapat huruf "ia" yang memiliki beberapa perubahan seperti "ia", "ya", "hya" dan "iya".
- Ketidaksesuaian aksara Latin dari Lontar Doyan Neda (Dewi Anjani).

4.2.2. Pengujian *blackbox*

Pengujian dengan *blackbox testing* melakukan pengujian terhadap fungsi-fungsi pada fitur aplikasi tanpa harus memperhatikan *source* atau struktur program. Dengan *blackbox testing*, peneliti dapat mengetahui kekurangan dan *error (bug)* yang terjadi pada fitur-fitur yang ada dalam aplikasi sebelum dirilis. Pengujian *blackbox* dilakukan oleh 10 orang

mahasiswa Program Studi Teknik Informatika Universitas Mataram. Hasil pengujian *blackbox* dapat dilihat pada Tabel XV.

TABEL XV. HASIL PENGUJIAN BLACKBOX.

No	Fungsi Aplikasi	Hasil Pengujian	
		Berhasil	Gagal
1.	Menampilkan halaman "Splash Screen"	10 resp.	-
2.	Menampilkan halaman "Transliterasi Aksara Sasak"	10 resp.	-
3.	Memasukkan teks Latin	10 resp.	-
4.	Menampilkan aksara Sasak	9 resp.	1 resp.
5.	Menghapus teks Latin	10 resp.	-
6.	Memasukkan karakter "ē"	10 resp.	-
7.	Memasukkan karakter petik	10 resp.	-
8.	Memasukkan karakter penegas "a "	10 resp.	-
9.	Mengatur ukuran aksara Sasak	10 resp.	-
10.	Mengatur <i>align left</i>	10 resp.	-
11.	Mengatur <i>align center</i>	10 resp.	-
12.	Mengatur <i>align right</i>	10 resp.	-
13.	Melakukan <i>copy</i> aksara to <i>clipboard</i>	10 resp.	-
14.	Melakukan <i>scrolling</i> pada <i>display text</i> aksara	10 resp.	-
15.	Menampilkan halaman "Aksara Sasak"	10 resp.	-
16.	Menampilkan <i>detail item</i> aksara Sasak	10 resp.	-
17.	Menampilkan halaman "Petunjuk Aplikasi"	10 resp.	-
18.	Menampilkan <i>detail item</i> petunjuk aplikasi	10 resp.	-
19.	Menampilkan halaman "Tentang Aplikasi"	10 resp.	-

Melalui pengujian *blackbox* pada Tabel, terdapat 1 fungsi dari aplikasi yang tidak berjalan yaitu tidak dapat menampilkan aksara Sasak hasil transliterasi. Dari observasi yang dilakukan, aksara Sasak tidak dapat ditampilkan dikarenakan *smartphone* responden tidak mendukung *Unicode Balinese* yang digunakan dalam pengembangan aplikasi. Sehingga yang ditampilkan untuk pengguna adalah simbol kotak.

4.2.3. Pengujian MOS (*Mean Opinion Score*)

Pengujian dengan MOS (*Mean Opinion Score*) dilakukan dengan memberikan kuesioner kepada sejumlah responden. Pengujian MOS dilakukan untuk mengetahui penilaian dari subjektivitas responden

terhadap kualitas suatu layanan aplikasi. Pengujian MOS menggunakan beberapa pertanyaan dalam bentuk kuesioner yang diberikan kepada responden. Berikut ini daftar pertanyaan yang diajukan kepada responden:

- a. Aplikasi “Transliterasi Aksara Sasak” dapat berjalan dengan baik. (Tidak terjadi *error / force close*, dll.)
- b. Aplikasi “Transliterasi Aksara Sasak” dapat membantu dalam transliterasi aksara Latin ke aksara Sasak.
- c. Aplikasi “Transliterasi Aksara Sasak” dapat menampilkan dan memberikan hasil transliterasi yang sesuai dengan aturan penulisan Aksara Sasak.
- d. Aplikasi “Transliterasi Aksara Sasak” memiliki tampilan yang menarik dan mudah digunakan (*User Friendly*).
- e. Aplikasi “Transliterasi Aksara Sasak” dapat membantu dalam pelestarian Aksara Sasak.

Dari beberapa pertanyaan yang diajukan, kemudian responden diminta untuk menjawab pertanyaan dengan beberapa parameter jawaban sesuai dengan Tabel.

TABEL XVI. PARAMETER PENILAIAN MOS.

MOS	Keterangan	Bobot Nilai	Kelompok
SS	Sangat Setuju	5	<i>Good</i>
S	Setuju	4	<i>Good</i>
KS	Kurang Setuju	3	<i>Neutral</i>
TS	Tidak Setuju	2	<i>Bad</i>
STS	Sangat Tidak Setuju	1	<i>Bad</i>

Berdasarkan jawaban yang diberikan oleh responden, kemudian dilakukan perhitungan MOS dari setiap pertanyaan yang diberikan kepada responden dengan persamaan (2).

$$Mean P_i = \frac{\sum P_i}{n} \tag{2}$$

Dimana *Mean p_i* merupakan skor rata-rata dari setiap pertanyaan, *p_i* merupakan jumlah skor dari responden terhadap setiap pertanyaan, dan *n* merupakan jumlah responden yang melakukan pengujian MOS. Kemudian dilakukan perhitungan MOS dengan menggunakan persamaan (3).

$$MOS = \frac{\sum_{n=1}^k mean P_i}{k} \tag{3}$$

Dimana MOS dihasilkan dari total skor rata-rata dari setiap pertanyaan yang dibagi dengan *k* yaitu jumlah pertanyaan yang diajukan.

Pengujian MOS aplikasi “Transliterasi Aksara Sasak” dilakukan dengan memanfaatkan kuesioner *google form* dengan penyebaran melalui media sosial. Pengujian MOS mendapatkan 40 orang masyarakat Suku Sasak sebagai responden dengan hasil yang dapat disajikan pada Tabel.

TABEL XVII. HASIL PENGUJIAN MOS.

Pertanyaan	SS (5)	S (4)	KS (3)	TS (2)	STS (1)	Total	Mean P _i
Pertanyaan 1	20	18	1	1	-	40	4,42
Pertanyaan 2	21	16	3	-	-	40	4,45
Pertanyaan 3	18	19	3	-	-	40	4,37
Pertanyaan 4	18	21	1	-	-	40	4,42
Pertanyaan 5	25	14	1	-	-	40	4,60
MOS (<i>Mean Opinion Score</i>)							4,45

Melalui Tabel , diketahui bahwa hasil pengujian MOS yang dilakukan terhadap 40 orang responden menunjukkan aplikasi “Transliterasi Aksara Sasak” yang dibangun telah berjalan dengan baik dengan mendapatkan nilai 4,45 dari skala 5 (hasil uji MOS ≥ 4.00).

4.3. Tampilan Aplikasi

Penelitian menghasilkan sebuah aplikasi *android* yang terdiri dari 5 halaman utama, yaitu halaman *splash screen*, halaman transliterasi, halaman aksara Sasak, halaman petunjuk dan halaman tentang aplikasi. Adapun tampilan aplikasi “Transliterasi Aksara Sasak” yang telah dibangun yaitu :



Gambar 8. Halaman *splash screen*.

Halaman *splash screen* pada Gambar 8 memuat logo dan nama aplikasi yang berfungsi sebagai tampilan awal ketika memasuki aplikasi. Ketika pengguna pertama kali membuka aplikasi “Transliterasi Aksara Sasak”, maka halaman *splash screen* ditampilkan selama 3 detik dan kemudian diarahkan ke halaman “Transliterasi” pada Gambar 9.



Gambar 9. Halaman transliterasi.

Halaman transliterasi aksara merupakan halaman dimana pengguna dapat melakukan transliterasi aksara Latin menjadi aksara Sasak. Pengguna dapat memasukkan teks beraksara Latin pada kolom editor dan kemudian sistem menampilkan hasil transliterasi yang pada bagian *display text* seperti pada Gambar 9.



Gambar 10. Halaman aksara Sasak.

Halaman aksara Sasak merupakan halaman yang memuat aksara yang digunakan dalam penulisan aksara Sasak. Pada halaman aksara Sasak, terdapat item-item yang dapat dipilih pengguna untuk menampilkan aksara Sasak.



Gambar 11. Halaman petunjuk.

Pada halaman petunjuk, terdapat item-item yang dapat dipilih pengguna untuk menampilkan petunjuk penggunaan dan informasi-informasi yang dapat dilihat oleh pengguna. Ketika pengguna memilih salah satu item, maka akan tampil detail petunjuk dari item yang dipilih.



Gambar 12. Halaman tentang aplikasi.

Halaman tentang aplikasi memuat deskripsi dan informasi mengenai aplikasi transliterasi aksara Sasak.

5. KESIMPULAN DAN SARAN

Berdasarkan pengujian akurasi yang telah dilakukan, aplikasi transliterasi aksara Latin menjadi aksara Sasak dengan algoritma *rule based* memperoleh akurasi 97,20% dari 250 data papan nama jalan di Kota Mataram, 96,375% dari 800 data Kamus Bali - Indonesia beraksara Latin dan Bali serta 65,33% dari 600 kasus data lontar Doyan Neda (Dewi Anjani). Secara keseluruhan akurasi yang diperoleh aplikasi sebesar 85,21% dari 1650 data uji. Pengujian fungsionalitas aplikasi dengan metode *blackbox* menunjukkan bahwa fungsi-fungsi pada aplikasi "Transliterasi Aksara Sasak" telah berjalan dengan baik, namun hasil transliterasi aksara Sasak tidak dapat ditampilkan untuk smartphone yang belum support dengan *Unicode Balinese*. Pengujian MOS (*Mean Opinion Score*) menunjukkan bahwa aplikasi telah berjalan dengan baik dan layak untuk digunakan masyarakat dengan memperoleh nilai sebesar 4,45 dari skala 5 terhadap 40 orang responden. Sehingga melalui pengujian akurasi, *blackbox* dan MOS (*mean opinion score*) yang telah dilakukan, algoritma *rule based* telah berhasil diimplementasikan dalam rancang bangun aplikasi transliterasi aksara Latin menjadi aksara Sasak.

Adapun saran untuk pengembangan berikutnya agar aplikasi transliterasi dapat melakukan transliterasi dari aksara Sasak menjadi aksara Latin, aplikasi transliterasi kedepannya dapat melakukan transliterasi melalui pengambilan gambar atau foto serta memperbanyak data uji dari lontar atau tulisan-tulisan beraksara Sasak untuk menambah pengetahuan mengenai aturan penulisan aksara Sasak.

DAFTAR PUSTAKA

- [1] R. Maulana, *Aksara-Aksara di Nusantara Seri Ensiklopedia*, I. Yogyakarta: Penerbit Samudra Biru, 2020.
- [2] Unicode Consortium, "Balinese - Unicode Character Table." [Online]. Available: <https://unicode-table.com/en/blocks/balinese/>. [Accessed: 15-Dec-2020].
- [3] H. Ismi, Asrin, and A. Widodo, "Analisis Penggunaan Aksara Sasak dalam Keseharian Masyarakat Lombok Barat di Era Globalisasi," *Al Ma'arif J. Pendidik. Sos. dan Budaya*, vol. 2, no. 2, pp. 226–232, 2020.
- [4] A. Aranta *et al.*, "Learning media for the transliteration of Latin letters into Bima script based on android applications," *J. Educ. Learn.*, vol. 15, no. 2, pp. 275–282, 2021.
- [5] A. Aranta, F. Bimantoro, and I. P. T. Putrawan, "Penerapan Algoritma Rule Base dengan Pendekatan Hexadesimal pada Transliterasi Aksara Bima Menjadi Huruf Latin," *J. Teknol. Informasi, Komputer, dan Apl. (JTIKA)*, vol. 2, no. 1, pp. 130–141, Mar. 2020.
- [6] M. Sudarma, I. N. S. Kumara, and I. P. A. E. D. Udayana, "Transliteration Balinese Latin Text Becomes Aksara Bali Using Rule Base And Levenshtein Distance Approach," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 2, no. 3, pp. 401–408, 2016.
- [7] H. Gian Putra, M. Ulum, and Haryanto, "Transliterasi Huruf Latin Bahasa Indonesia ke Aksara Jawa Menggunakan Metode Snakecut," *Fortei7*, vol. 2, no. 1, pp. 96–101, 2019.
- [8] E. Protalinski, "Android passes 2.5 billion monthly active devices," 2019. [Online]. Available: <https://venturebeat.com/2019/05/07/android-passes-2-5-billion-monthly-active-devices/>. [Accessed: 15-Feb-2021].
- [9] E. Utami, J. Istiyanto, S. Hartati, M. Marsono, and A. Ashari, "Penerapan Rule Based dalam Membangun Transliterasi Jawatex," *Bimipa*, vol. 23, no. 1, pp. 78–94, 2013.
- [10] B. W. Yohanes, T. Robert, and S. Nugroho, "Sistem Penerjemah Bahasa Jawa-Aksara Jawa Berbasis Finite State Automata," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 6, no. 2, pp. 127–132, Jun. 2017.
- [11] A. Aranta, Ig. A. Gunadi, and G. Indrawan, "Utilization Of Hexadecimal Numbers In Optimization Of Balinese Transliteration String Replacement Method," in *2018 International Conference on Computer Engineering, Network and Intelligent Multimedia (CENIM)*, 2018, no. October 2019, pp. 131–136.
- [12] I. P. A. E. D. Udayana, M. Sudarma, and I. N. S. Kumara, "Balinese Latin Text Becomes Aksara Bali Using Rule Base Method," *Int. J. Res. IT, Manag. Eng.*, vol. 07, no. 05, pp. 1–7, 2017.
- [13] G. Indrawan and I. K. Pramarta, "The Development of Learning Mobile Application of Latin-to-Balinese Script Transliteration," *J. Pekommas*, vol. 4, no. 2, p. 123, Oct. 2019.
- [14] *Bahan Ajar Kurikulum Muatan Lokal Bagi (SMA, SMK, SMALB) Kelas XII*. Mataram: Dinas Pendidikan dan Kebudayaan Provinsi Nusa Tenggara Barat, 2018.
- [15] Jamaluddin, "Sejarah Tradisi Tulis dalam Masyarakat Sasak Lombok," *Ulumuna*, vol. IX, no. 2, pp. 369–384, 2005.
- [16] M. Tajuddin, "Lontar Doyan Neda (Dewi Anjani)," 2021. [Online]. Available: <https://tajuddin.web.id/detail-naskah/14?page=2>. [Accessed: 15-Jan-2021].
- [17] S. Hudaa, "Transliterasi, Serapan, dan Padanan Kata: Upaya Pemutakhiran Istilah dalam Bahasa Indonesia," *SeBaSa*, vol. 2, no. 1, p. 1, May 2019.
- [18] A. T. Haryanto and T. L. A. Sucipto, *Sistem komputer SMK/MAK Kelas X Semester I*, I. Jakarta: Kementerian Pendidikan dan Kebudayaan Republik Indonesia, 2013.