

BAB II

LANDASAN TEORI

2.1 *Finite State Automata*

Finite state automata (FSA) adalah mesin abstrak berupa sistem model matematika dengan masukan dan keluaran diskrit yang dapat mengenali bahasa paling sederhana (bahasa reguler) dan dapat diimplementasikan secara nyata (Simaremare, 2013).

Finite State Automata (FSA) dapat menerima *input* dan mengeluarkan *output* yang memiliki *state* yang berhingga banyaknya dan dapat berpindah dari satu *state* ke *state* lainnya berdasarkan *input* dan fungsi transisi. *Finite State Automata* tidak memiliki tempat penyimpanan/*memory*, hanya bisa mengingat *state* terkini.

Finite State Automata dinyatakan oleh pasangan 5 *tuple*, yaitu:

$$M = (Q, \Sigma, \delta, S, F)$$

Q = himpunan *state*

Σ = himpunan simbol *input*

δ = fungsi transisi $\delta : Q \times \Sigma$

S = *state* awal / *initial state*, $S \in Q$

F = *state* akhir, $F \subseteq Q$

2.2 *Regular Expression*

Regular Expression atau *RegEx*, adalah deretan karakter spesial yang mendefinisikan sebuah pola dalam pencarian teks (Anggie, 2015).

Pola pada *Regular Expression* biasanya digunakan dalam algoritma pencarian *string* dan validasi *input user*.

2.3 Aksara Sunda

Aksara Sunda merupakan sistem penulisan yang berkembang di Jawa Barat untuk menulis tulisan berbahasa Sunda. Menurut catatan sejarah, aksara ini telah dipakai oleh orang Sunda dari abad ke-14 sampai abad ke-18. (Ridwan, 2018).

Berhubung dengan aksara Sunda itu dapat dibedakan atas beberapa variasi sesuai dengan bahan tulisannya (batu, logam, kulit, daun, pisau pangot, tinta, pahat, palu), masa pemakaiannya, serta perkembangan penguasaan teknik dan kecerdasan manusianya, maka perlu ditentukan satu variasi yang dapat dijadikan aksara yang baku. Berdasarkan kelengkapan aksara dan sistem pengaksaraannya serta kepraktisan untuk menuliskannya dewasa ini, aksara Sunda variasi yang ditulis pada naskah yang seyogyanya ditetapkan sebagai Aksara Sunda yang akan dipakai sekarang atau lebih dikenal aksara Sunda Baku. (Idin, 2008)

Aksara Sunda Baku merupakan sistem penulisan hasil penyesuaian aksara Sunda kuno yang digunakan untuk menuliskan bahasa Sunda kontemporer. Saat ini aksara Sunda baku juga lazim disebut dengan istilah aksara Sunda. (Dadan, 2008).

Aksara Sunda merupakan Aksara Silabis dimana karakter konsonan dengan tanda vokalisasi menyatu dengan aksara dasar.

Aksara Sunda terdiri dari 32 aksara, yaitu :

1. Aksara Swara (aksara vokal mandiri): a, é, i, o, u, e, dan eu
2. Aksara Ngalagéna (konsonan berbunyi a): ka-ga-nga, ca-ja-nya, ta-da-na, pa-ba-ma, ya-ra-la, wa-sa-ha, fa-va-qa-xa-za.
3. Rarangkéén untuk mengubah, menghapus, atau menambah bunyi pada aksara dasar.
4. Angka Aksara Sunda. Angka diapit oleh simbol garis lurus.

2.4 Komponen yang digunakan dalam pengembangan sistem

2.4.1 Visual Studio Code

















Visual Studio Code atau *VSCode* adalah perangkat lunak teks editor *open source* yang dibuat oleh *Microsoft* untuk mengedit *source code* berbagai jenis bahasa pemrograman.

2.5 Alat Bantu Pengembangan Sistem

2.5.1 Flowchart

Flowchart atau bagan alir adalah bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian masalah (Al-Bahra, 2005). *Flowchart* juga dapat diartikan sebagai bagan yang menunjukkan alir di dalam program/prosedur secara logika (Jogiyanto, 2005).

Berikut simbol-simbol pada *flowchart* beserta fungsinya:

	Flow Direction symbol Yaitu simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga connecting line.		Simbol Manual Input Simbol untuk pemasukan data secara manual on-line keyboard
	Terminator Symbol Yaitu simbol untuk permulaan (start) atau akhir (stop) dari suatu kegiatan		Simbol Preparation Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan di dalam storage.
	Connector Symbol Yaitu simbol untuk keluar - masuk atau penyambungan proses dalam lembar / halaman yang sama.		Simbol Predefine Proses Simbol untuk pelaksanaan suatu bagian (sub-program)/prosedure
	Connector Symbol Yaitu simbol untuk keluar - masuk atau penyambungan proses pada lembar / halaman yang berbeda.		Simbol Display Simbol yang menyatakan peralatan output yang digunakan yaitu layar, plotter, printer dan sebagainya.
	Processing Symbol Simbol yang menunjukkan pengolahan yang dilakukan oleh komputer		Simbol disk and On-line Storage Simbol yang menyatakan input yang berasal dari disk atau disimpan ke disk.
	Simbol Manual Operation Simbol yang menunjukkan pengolahan yang tidak dilakukan oleh computer		Simbol magnetik tape Unit Simbol yang menyatakan input berasal dari pita magnetik atau output disimpan ke pita magnetik.
	Simbol Decision Simbol pemilihan proses berdasarkan kondisi yang ada.		Simbol Punch Card Simbol yang menyatakan bahwa input berasal dari kartu atau output ditulis ke kartu
	Simbol Input-Output Simbol yang menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya		Simbol Dokumen Simbol yang menyatakan input berasal dari dokumen dalam bentuk kertas atau output dicetak ke kertas.

Gambar 2.1 Simbol-Simbol pada *Flowchart* beserta Fungsinya

(Rizki Ardi Saputra, 2017)

2.5.2 UML (*Unified Modelling Language*)

Unified Modelling Language (UML) adalah sebuah “bahasa” yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. *UML* menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan *UML* dapat dibuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena *UML* juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka lebih cocok untuk penulisan piranti lunak dalam bahasa berorientasi objek seperti *C++*, *Java*, atau *VB.NET* (Sulistyorini, 2009).

Berikut ini adalah beberapa jenis diagram *UML* :

a. *Use case Diagram*

Diagram ini bersifat statis. Diagram ini memperlihatkan himpunan *Use Case* dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan serta diharapkan pengguna (Sulistyorini, 2009).

b. *Sequence Diagram* (Diagram urutan)

Diagram ini bersifat dinamis. *Sequence Diagram* merupakan diagram interaksi yang menekankan pada pengiriman pesan (*message*) dalam suatu waktu tertentu (Sulistyorini, 2009).

c. *Class Diagram*

Class diagram adalah sebuah *class* yang menggambarkan struktur dan penjelasan *class*, paket, dan objek serta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class diagram* juga menjelaskan hubungan antar *class* dalam sebuah sistem yang sedang dibuat dan bagaimana caranya agar mereka saling berkolaborasi untuk mencapai sebuah tujuan. (Kusumawaty, 2012).

2.6 Penelitian Terkait

Pada bagian penelitian terkait, ada beberapa jurnal yang diangkat sebagai referensi dalam memperkaya bahan kajian pada penelitian dan sebagai acuan untuk penelitian selanjutnya. Berikut merupakan penelitian-penelitian berupa beberapa jurnal yang berkaitan dengan penelitian yang akan dilakukan, diantaranya akan dijelaskan pada tabel 2.1 sebagai berikut:

Tabel 2.1 Rangkuman Penelitian Terkait

No	Nama Penulis	Judul	Hasil Penelitian
1.	C Slamet (2018)	Latin to Sundanese script conversion using Finite State automata algorithm	Menggunakan algoritma FSA untuk pengenalan silabis Aksara Sunda dari tulisan latin. Hasilnya FSA sangat bagus dalam mengenali dan menkonversi Tulisan Latin ke Aksara Sunda.
2.	Lelah (2018)	Implementasi Finite State Automata Pada Aplikasi Pemberlajaran Aksara Sunda	Menggunakan algoritma FSA untuk pengenalan silabis Aksara Sunda dari tulisan latin dengan pola suku kata hingga FSA tingkatan 3.
3	Hendro Triokta Brianto (2014)	Konversi Alfabet Latin ke Aksara Sunda Kaganga	Menggunakan algoritma Brute Force untuk mengubah tulisan latin menjadi Aksara Sunda
4	Abdul Haris, M.Kom (2015)	Aplikasi Konversi Aksara Sunda ke Bahasa Indonesia berbasis Web Menggunakan PHP MYSQL	Menggunakan kamus untuk Mengkonversi Aksara Latin menjadi Aksara Sunda
5	Dimas Nur Iman Tabroni (2016)	Aplikasi Konversi Aksara Latin ke Aksara Jawa Menggunakan Finite State Automata dengan Visual Basic	Menggunakan algoritma FSA untuk pengenalan silabis Aksara Jawa dari tulisan latin. Hasilnya FSA sangat bagus dalam mengenali dan menkonversi Tulisan Latin ke Aksara Jawa. Codepoint karakter belum menggunakan sistem Unicode
6	Mohamad Arifin Wabula (2015)	Implementasi Finite State Automata pada Aplikasi Translator Latin - Aksara Jawa	Menggunakan algoritma FSA untuk pengenalan silabis Aksara Sunda dari tulisan latin dengan pola suku kata hingga FSA tingkatan 3.
7	Jenie Sundari	Melestarikan Aksara Sunda Dengan	Perancangan dan implementasi Media Pembelajaran Aksara Sunda

No	Nama Penulis	Judul	Hasil Penelitian
	(2016)	Aplikasi Multimedia	dengan multimedia digital
8	Anton Sahbana (2016)	Aplikasi Pembelajaran Aksara Sunda	Media pembelajaran Aksara Sunda berbasis Web menggunakan Bahasa Pemrograman PHP
9	Tyara Umi Yuhanis Sarrahdiba (2014)	Perancangan Media Pembelajaran Aksara Sunda untuk Siswa Sekolah Dasar (Studi Kasus SDN Sukasenang)	Perancangan Media Pembelajaran Aksara Sunda yang disajikan di dalam sebuah Buku
10	Arini Habibah (2020)	Aplikasi pembelajaran aksara sunda berbasis Android	Perancangan dan implementasi Media Pembelajaran Aksara Sunda dengan multimedia di platform Android

