

LAPORAN SEMENTARA

LEXICAL ANALYZER DAN PARSER

Disusun untuk memenuhi tugas besar mata kuliah Teori Bahasa dan Automata

Oleh Kelompok 8:

- | | |
|-------------------------------|--------------|
| 1. Naufal Abdurrahman Burhani | (1301204008) |
| 2. Harvian Khusnan Hafidz | (1301204303) |
| 3. Khalillullah Al Faath | (1301204376) |



Fakultas Informatika

Telkom University

2022

Daftar Isi

Daftar Isi	2
1. Tinjauan Pustaka.....	3
1.1. Bahasa Rusia dan Aksara Kiril.....	3
1.2. <i>Lexical Analysis</i>	4
1.2.1. Token	4
1.2.2. Deterministic Finite Automata (DFA)	4
1.2.3. Context-free Grammar	4
2. Ketentuan Program	4
2.1. Context-free Grammar	4
2.2. Finite Automata.....	5
2.2.1. FA Subject	5
2.2.2. FA Verb	6
2.2.3. FA Object.....	6
2.2.4. FA Gabungan	6
3. Implementasi Program	7
3.1. Variabel-variabel dalam pembangunan web	7
3.2. Enumerasi list alfabet dan daftar state.....	8
3.3. Daftar Transition Table	9
3.4. Lexical Analyzer	10
4. Hasil Program	10
4.1. Tampilan web	10
4.2. Output program	11
4.2.1. Contoh input valid.....	11
4.2.2. Contoh input invalid	12
5. Resources dan tutorial	13
5.1. Resources	13
5.2. Tutorial	13
6. Bibliography	15

1. Tinjauan Pustaka

1.1. Bahasa Rusia dan Aksara Kiril

Bahasa Rusia adalah bahasa yang termasuk dalam rumpun bahasa Slavia Timur, yang merupakan bahasa yang umumnya digunakan di negara-negara bekas Uni Soviet. Bahasa Rusia juga termasuk ke dalam bahasa resmi PBB (Esipova & Gradirovski, 2008).

Bahasa Rusia umumnya memakai aksara Kiril (bahasa Inggris: cyrillics, rusia: кириллица). Saat ini, alfabet Kiril yang dipakai di Rusia telah dimodifikasi sedemikian rupa dan terdiri atas sebelas huruf vokal, dua puluh konsonan, serta dua penanda keras dan lunak sehingga seluruhnya ada 33 huruf.

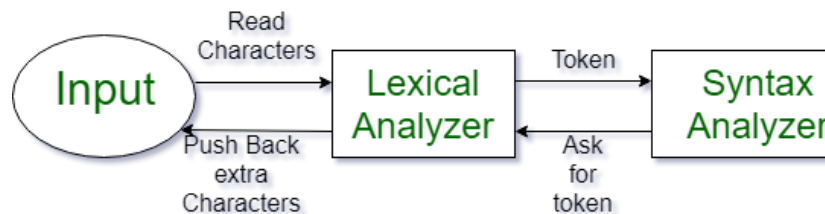
Berikut adalah aksara dalam bahasa Rusia.

Huruf	Nama huruf	Alih Aksara	Alih Aksara nama huruf
А а	а	a	<i>a</i>
Б б	бэ	b	<i>be</i>
В в	вэ	v	<i>ve</i>
Г г	гэ	g	<i>ge</i>
Д д	дэ	d	<i>de</i>
Е е	йэ	ye	<i>ye</i>
Ё ё	йо	yo	<i>yo</i>
Ж ж	жэ	zh	<i>zhe</i>
З з	зэ	z	<i>ze</i>
И и	и	i	<i>i</i>
Й й	и краткое	y	<i>i kratkoye</i>
К к	ка	k	<i>ka</i>
Л л	эл	l	<i>el</i>
М м	эм	m	<i>em</i>
Н н	эн	n	<i>en</i>
О о	о	o	<i>o</i>
П п	пэ	p	<i>pe</i>
Р р	эр	r	<i>er</i>
С с	эс	s	<i>es</i>
Т т	тэ	t	<i>te</i>
У у	у	u	<i>u</i>
Ф ф	эф	f	<i>ef</i>
Х х	ха	h	<i>ha</i>
Ц ц	цэ	ts	<i>tse</i>
Ч ч	че	ch	<i>che</i>
Ш ш	ша	sh	<i>sha</i>
Щ щ	ща	sch	<i>scha</i>

Ъ ъ	твёрдый знак	*	<i>tvyordiy znak</i>
Ы ы	ы	y	<i>y</i>
Ь ь	мягкий знак	^*	<i>myagkiy znak</i>
Э э	э	e	<i>e</i>
Ю ю	йу	yu	<i>yu</i>
Я я	йа	ya	<i>ya</i>

1.2. Lexical Analysis

Lexical Analysis adalah tahap pertama dalam compiler yang dengannya program input tingkat tinggi diubah menjadi deretan token. *Lexical Analysis* ini dapat diimplementasikan dengan menggunakan Deterministic Finite Automata (DFA).



Outputnya adalah deretan token yang kemudian akan dikirimkan ke parser untuk dilakukan analisis sintaks.

1.2.1. Token

Token adalah kumpulan karakter yang dapat diperlakukan sebagai sebuah unit dalam grammar suatu bahasa (GeeksforGeeks, 2021).

1.2.2. Deterministic Finite Automata (DFA)

DFA (Deterministic Finite Automata) adalah Finite-state Machine atau mesin keadaan terbatas yang menerima atau menolak string dari simbol dan hanya menghasilkan perhitungan unik dari otomata untuk setiap string yang di masukan (Belajar CPP, 2018).

1.2.3. Context-free Grammar

Context-free Grammar (CFG) adalah sebuah grammar yang aturan produksinya berbentuk $A \rightarrow \alpha$, di mana A adalah sebuah symbol non-terminal dan α adalah kumpulan symbol terminal dan non-terminal dengan catatan α bisa merupakan string kosong (John E. Hopcroft, et.al., 2001).

2. Ketentuan Program

2.1. Context-free Grammar

Notasi yang digunakan untuk context-free grammar adalah Backus-Naur form. Berikut ini adalah context-free grammar yang digunakan dengan notasi Backus-Naur form.

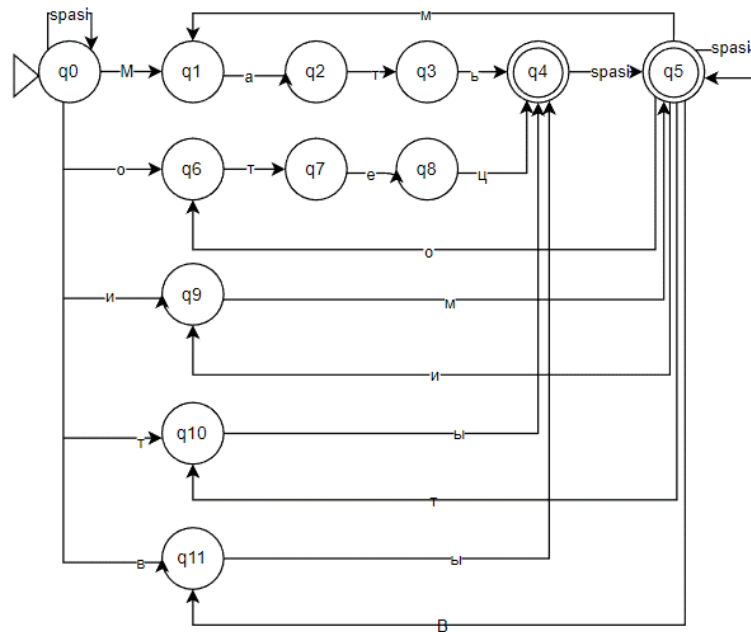
$\langle S \rangle ::= \langle SB \rangle \langle VB \rangle \langle OB \rangle$
 $\langle SB \rangle ::= \text{"мать"} \mid \text{"отец"} \mid \text{"им"} \mid \text{"ты"} \mid \text{"вы"}$
 $\langle VB \rangle ::= \text{"готовит"} \mid \text{"уяить"} \mid \text{"тянуть"} \mid \text{"рисовать"} \mid \text{"читать"}$
 $\langle OB \rangle ::= \text{"книга"} \mid \text{"комар"} \mid \text{"ром"} \mid \text{"ученик"}$

2.2. Finite Automata

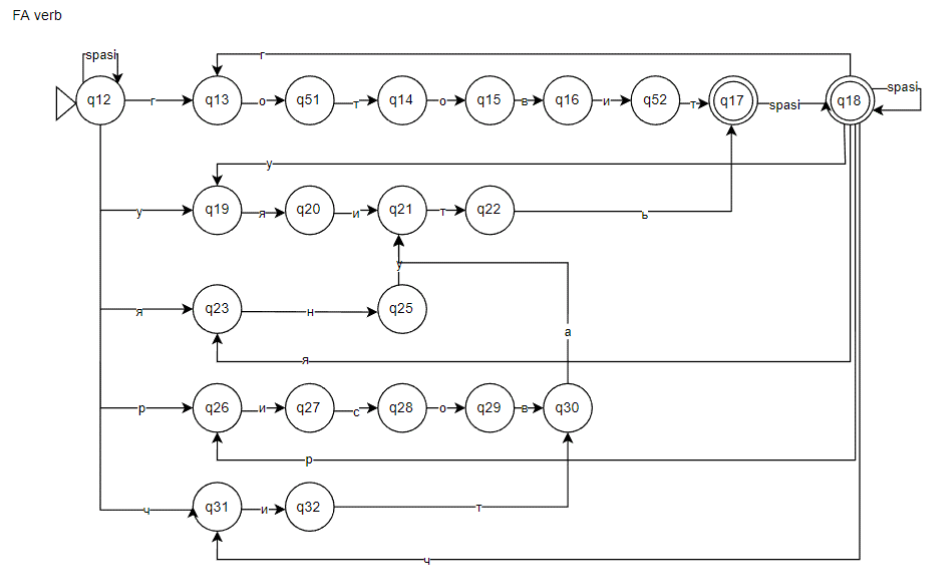
Setelah membuat aturan untuk grammar, langkah selanjutnya yang akan dilakukan adalah membangun finite automata sebagai rules state-state serta value yang akan digunakan pada program lexical analyzer yang akan dibuat. Adapun bentuk model finite automata yang dibuat adalah berikut ini.

2.2.1. FA Subject

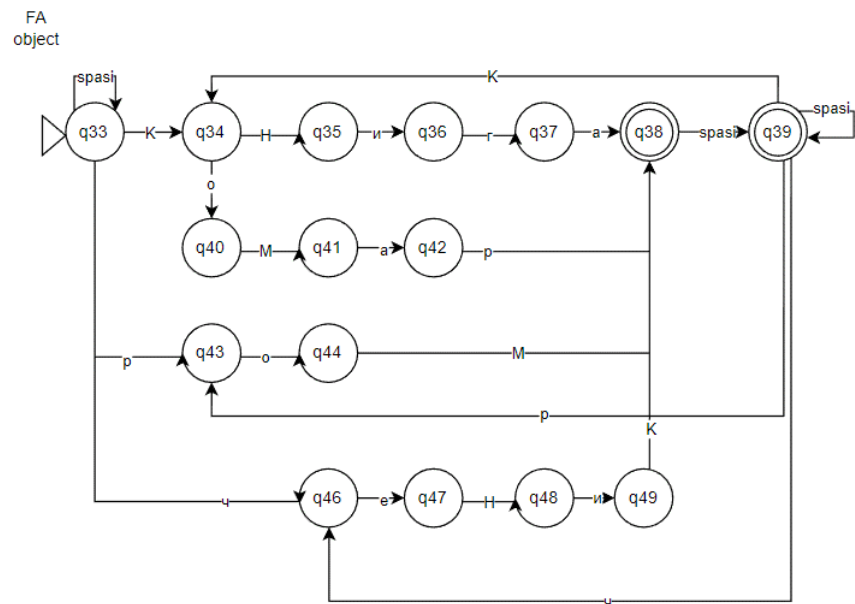
FA
subject



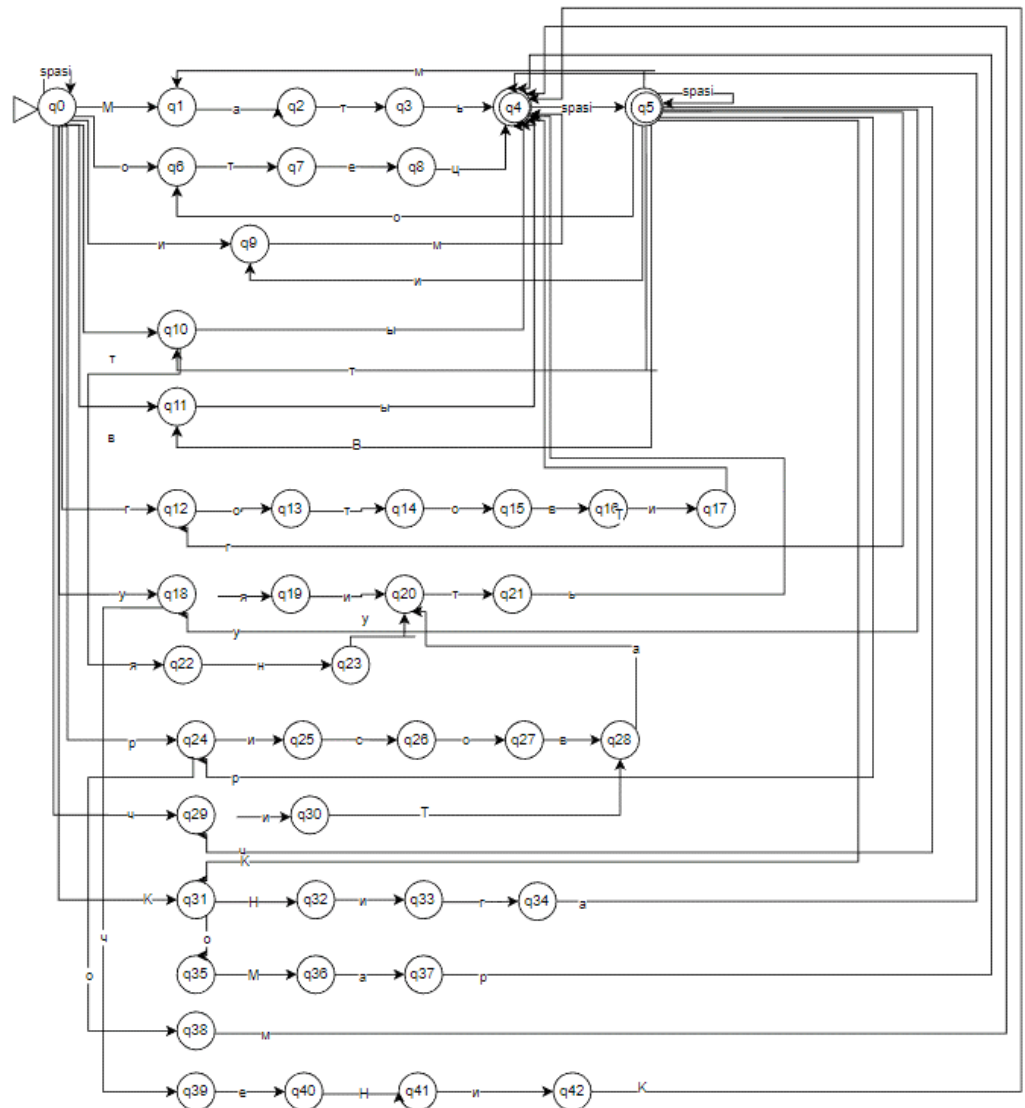
2.2.2. FA Verb



2.2.3. FA Object



2.2.4. FA Gabungan



3. Implementasi Program

Program yang dibuat adalah program berbasis web dengan menggunakan bahasa Javascript. Untuk pembangunan websitenya menggunakan HTML dan CSS.

3.1. Variabel-variabel dalam pembangunan web

```
var inputKalimat = document.getElementById('input_kalimat')
;
var hasil = document.getElementById('result');
var clear = document.getElementById('btn-clear');
var loading = document.getElementById('loading');
var form = document.getElementById('form')
```

3.2. Enumerasi list alfabet dan daftar state

Karena aksara yang digunakan bukan aksara latin maka perlu mengambil Unicodenya. Untuk aksara Kiril dalam Unicode terdiri dari decimal 1024 sampai 1279. Untuk handle huruf latin maka ditambahkan ke dalam list dengan Unicode dari 65—122.

Dilakukan juga pembentukan state yang terdiri dari “q” yang diikuti angka dari 0—24.

```
var alfabet = [];  
for (let i = 1024; i <= 1279; i++){  
  let text = String.fromCharCode(i);  
  alfabet.push(text);  
}  
  
for (let i = 65; i <= 122; i++){  
  let text = String.fromCharCode(i);  
  alfabet.push(text);  
}  
  
// const alfabet = /^[u0400-\u04FF]+$/;  
// console.log('Пруθim:', alfabet.test('Пруθim'));  
// console.log('Hello:', alfabet.test('Hello'));  
  
var stateList = [];  
  
let N = 42;  
for (let i = 0; i <= N; i++){  
  var strAngka = i.toString();  
  var str = "q";  
  var namaState = str.concat(strAngka);  
  stateList.push(namaState)  
}
```


3.3. Daftar Transition Table

Menginisialisasi untuk setiap alfabet sebagai error. Setelah itu dibuat transition table sesuai dengan FA gabungan.

```
for(var state in stateList) {
    for(daftarAlfabet in alfabet) {
        transitionTable[[stateList[state], daftarAlfabet]] = 'ERROR'
    }
    transitionTable[[stateList[state], '#']] = 'ERROR'
    transitionTable[[stateList[state], ' ']] = 'ERROR'
}
```

```
// initial state
transitionTable[["q0", " "]] = "q0";

// final state
transitionTable[["q4", " "]] = "q5";
transitionTable[["q4", "#"]] = "ACCEPT";
transitionTable[["q5", " "]] = "q5";
transitionTable[["q5", "#"]] = "ACCEPT";

//Subject мать
transitionTable[["q0", "м"]] = "q1"
transitionTable[["q1", "а"]] = "q2"
transitionTable[["q2", "т"]] = "q3"
transitionTable[["q3", "б"]] = "q4"

transitionTable[["q5", "м"]] = "q1"

//subject омега
transitionTable[["q0", "о"]] = "q6"
transitionTable[["q6", "т"]] = "q7"
transitionTable[["q7", "е"]] = "q8"
transitionTable[["q8", "ц"]] = "q4"

transitionTable[["q5", "ц"]] = "q6"
```

3.4. Lexical Analyzer

Dibuat lexical analyzer mengikut pada tata cara analisis leksikal pada poin 1.

```
form.onSubmit = (event) => {  
  
    event.preventDefault()  
  
    loading.style = 'display: inline-block'  
  
    // lexical analysis  
    var indexChar = 0;  
    var state = 'q0';  
    var currentToken = '';  
    var validation = '';  
    var inputChar = inputKalimat.value + '#';  
    console.log(inputChar);  
    while (state !== 'ACCEPT') {  
        var currentChar = inputChar.charAt(indexChar)  
        currentToken += currentChar  
        state = transitionTable[[state, currentChar]]  
        if(state === 'q4') {  
            console.log("valid gais")  
            validation += "valid "  
            currentToken = ''  
        }  
        if(state === 'ERROR') {  
            console.log("ERROR")  
            validation += "ERROR "  
            break;  
        }  
        indexChar++;  
    }  
}
```

4. Hasil Program

4.1. Tampilan web

Kelompok Russia	
Kata-Kata inputan Subjek yang dapat diterima :	
<div><div>Lexical Analyzer</div><div>русский язык Bahasa Rusia</div><div>Input kata atau kalimat dalam aksara russia</div><div>Input kalimat</div><div>ex: ученик</div><div>Analyze</div><div>Hasil</div><div></div><div>Clear</div></div>	

4.2. Output program

4.2.1. Contoh input valid

Dilakukan pengecekan dengan input sebagai berikut:

- Мать: mengeluarkan output “valid” karena ada pada grammar.
- Уяить: mengeluarkan output “valid” karena ada pada grammar.
- Комар: mengeluarkan output “valid” karena ada pada grammar.
- Комар готовит: mengeluarkan output “valid valid” karena kedua kata ada pada grammar

Kelompok Russia

"им (mereka)", "ты(кamu)", "вы(kalian)" Kata-Kata inputan Verb y

Lexical Analyzer
русский язык
Bahasa Rusia

Input kata atau kalimat dalam aksara russia

Input kalimat

мать

Analyze

Hasil

valid

Clear

Kelompok Russia

ка)", "ты(кamu)", "вы(kalian)" Kata-Kata inputan Verb yang dapat

Lexical Analyzer
русский язык
Bahasa Rusia

Input kata atau kalimat dalam aksara russia

Input kalimat

читать

Analyze

Hasil

valid

Clear

Kelompok Russia

ku)", "комар (nyamuk)", "ром (rum)", "ученик (murid)"

Lexical Analyzer
русский язык
Bahasa Rusia

Input kata atau kalimat dalam aksara russia

Input kalimat

комар

Analyze

Hasil

valid

Clear

Kelompok Russia

ng dapat diterima : "книга (buku)", "комар (nyamuk)", "ром (rum)"

Lexical Analyzer
русский язык
Bahasa Rusia

Input kata atau kalimat dalam aksara russia

Input kalimat

комар готовит

Analyze

Hasil

valid valid

Clear

4.2.2. Contoh input invalid

Dilakukan pengecekan input sebagai berikut.

- saya: mengeluarkan output “error” karena tidak ada pada grammar
- makan: mengeluarkan output “error” karena tidak ada pada grammar
- roti: mengeluarkan output “error” karena tidak ada pada grammar
- saya makan roti: mengeluarkan output “error” karena ketiganya tidak ada pada grammar

Kelompok Russia

(kan)", "тянуть (menarik)", "рисовать (menggambar)", "читать (men-
Lexical Analyzer
русский язык
Bahasa Rusia

Input kata atau kalimat dalam aksara russia

Input kalimat

Analyze

Hasil

ERROR

Clear

Kelompok Russia

"отец (ayah)", "им (mereka)", "ты(kamu)", "вы(kalian)" Kata-Kata

Lexical Analyzer

русский язык
Bahasa Rusia

Input kata atau kalimat dalam aksara russia

Input kalimat

Analyze

Hasil

ERROR

Clear

Kelompok Russia

dapat diterima : "книга (buku)", "комар (nyamuk)", "рум (rum)",

Lexical Analyzer

русский язык
Bahasa Rusia

Input kata atau kalimat dalam aksara russia

Input kalimat

Analyze

Hasil

ERROR

Clear

Kelompok Russia

"мать (ibu)", "отец (ayah)", "им (mereka)", "ты(kamu)", "вы(kalian)"

Lexical Analyzer

русский язык
Bahasa Rusia

Input kata atau kalimat dalam aksara russia

Input kalimat

Analyze

Hasil

ERROR

Clear

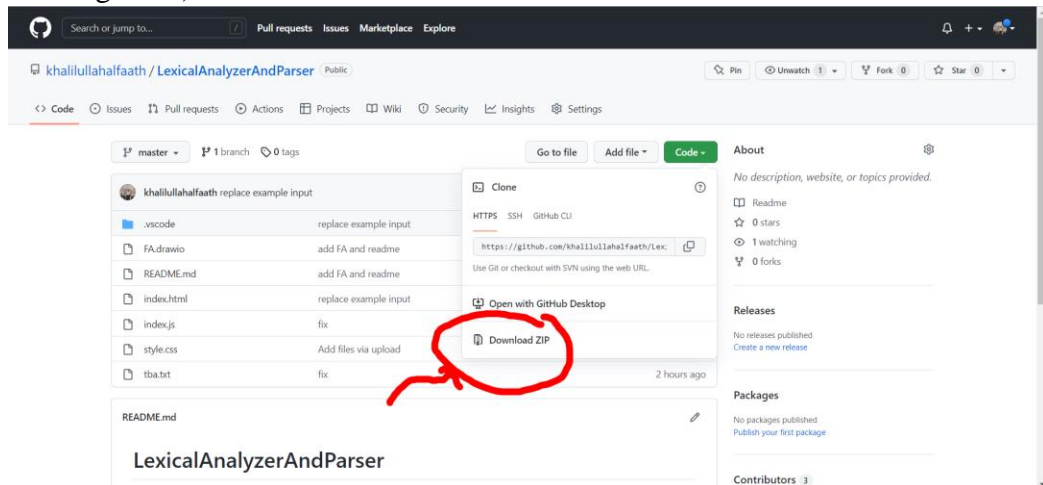
5. Resources dan tutorial

5.1. Resources

Link Github: <https://github.com/khalilullahalfaath/LexicalAnalyzerAndParser>

5.2. Tutorial

1. Download file dari link github pada resources (Jika sudah ada file zip, lompat ke Langkah 3)



2. Ekstrak file .zip dengan winrar atau apapun itu
3. Buka folder hasil ekstrak tadi dan buka file index.html

Name	Date modified	Type	Size
.vscode	08/06/2022 17:06	File folder	
FA.drawio	08/06/2022 01:23	DRAWIO File	359 KB
index	08/06/2022 01:23	Chrome HTML Do...	3 KB
index	08/06/2022 01:23	JavaScript Source ...	6 KB
README	08/06/2022 01:23	Markdown Source ...	1 KB
style	08/06/2022 01:23	CSS Source File	1 KB
tba	08/06/2022 01:23	Text Document	1 KB

4. Maka tampilan web akan muncul

The screenshot shows a web browser window with the title 'Lexical Analyzer'. The address bar shows the file path 'C:/Users/HP/Downloads/LexicalAnalyzerAndPars...'. The browser's taskbar at the bottom shows 'Khalilullah Al Faath', 'Dashboard', and 'APA Citation Gener...'. The main content area of the browser displays a web application titled 'Kelompok Russia'. Below this title is a subtitle in Russian: ',"отец (ayah)", "им (mereka)", "ты(kamu)", "вы(kalian)" Kata-Kata'. The application itself is titled 'Lexical Analyzer' and specifies 'русский язык' (Russian language) and 'Bahasa Rusia'. It prompts the user to 'Input kata atau kalimat dalam aksara russia' (Input word or sentence in Russian script). There is an input field labeled 'Input kalimat' with the example text 'ex: мать'. Below the input field is a blue 'Analyze' button. Underneath the button is the label 'Hasil' (Result) followed by a large, empty light gray rectangular box for the output. At the bottom of the application area is a red 'Clear' button.

5. Masukkan input sesuai yang diinginkan, contoh input yang diterima ada di teks berjalan pada web
6. Contoh inputan valid dan tidak valid ada pada laporan bab 4 (Output program)

6. Bibliography

Esipova, S., & Gradirovski, N. (2008, August 1). *Russian Language Enjoying a Boost in Post-Soviet States*. Retrieved from Gallup.com: <https://news.gallup.com/poll/109228/russian-language-enjoying-boost-postsoviet-states.aspx>