



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk mengubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODOLOGI DAN PERANCANGAN SISTEM

3.1 Metodologi Penelitian

Untuk melakukan penelitian berjudul Rancang Bangun Aplikasi Pencarian Berkas Menggunakan Algoritma Word2Vec. Maka, metode yang dilakukan oleh peneliti adalah penentuan topik, studi literatur, pengumpulan data, perancangan system, pembuatan sistem, pengujian sistem, evaluasi, serta konsultasi dan penulisan naskah penelitian. Adapun rencana pengerjaan digambarkan dalam tahapan pada Tabel 3.1

Tabel 3.1 Rencana tahap penelitian

Kegiatan	Minggu ke-													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Studi Literatur	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Pengumpulan Data			■	■										
Perancangan Sistem					■	■								
Pembuatan Sistem							■	■	■	■				
Pengujian Sistem dan Evaluasi											■	■		
Konsultasi dan penulisan	■	■	■	■	■	■	■	■	■	■	■	■	■	■

Adapun penjabaran secara mendetail mengenai metode yang digunakan adalah sebagai berikut.

1. Pengumpulan Data

Pengumpulan data akan dilakukan dengan cara mencari berkas menggunakan *keyword* yang mempunyai arti yang mirip dengan nama berkas atau *keyword* dengan *typo*.

2. Studi Literatur

Studi Literatur akan dilakukan dengan mempelajari dan mendalami beberapa topik terkait Word2Vec dan algoritma yang digunakan untuk menyelesaikan masalah tersebut, untuk diaplikasikan pada sistem.

3. Perancangan Sistem

Tahap perancangan sistem akan mempunyai alur seperti ini:

- Pengguna memasukkan *keyword* untuk mencari berkas yang ada di dalam komputer pengguna.
- Aplikasi akan mencari berkas pada komputer pengguna menggunakan algoritma Word2Vec.

4. Pembuatan Sistem

Pada tahap pembuatan sistem, akan dimulai dengan membuat program berbasis Windows dengan Bahasa Python menggunakan Jupyter Notebook untuk membuat *back-end* aplikasi.

5. Pengujian Sistem dan Evaluasi

Pengujian dan evaluasi akan dilakukan dengan cara menguji hasil aplikasi pencarian berkas yang dibuat dengan membetulkan *bug* yang ada pada aplikasi. Lalu dicatat hasil dan keakuratan *output* dari aplikasi.

6. Penulisan Laporan

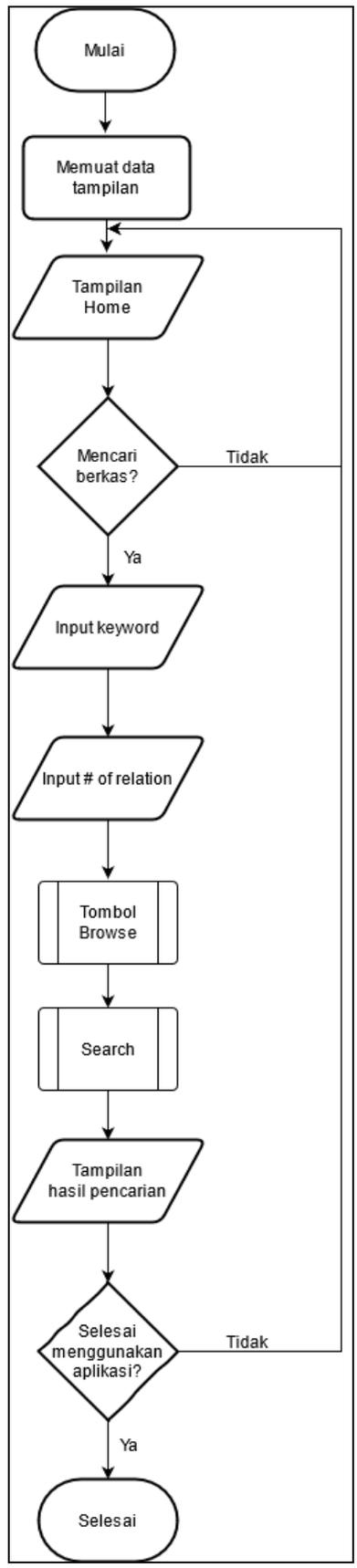
Penulisan laporan mencakup seluruh rangkaian dari awal seperti studi literatur sampai dengan proses evaluasi hingga kesimpulan dari penelitian yang

telah dilakukan. Kesimpulan yang didapat merupakan ringkasan hasil penelitian, pengolahan data dan evaluasi yang telah dilakukan.

3.2 Perancangan Aplikasi

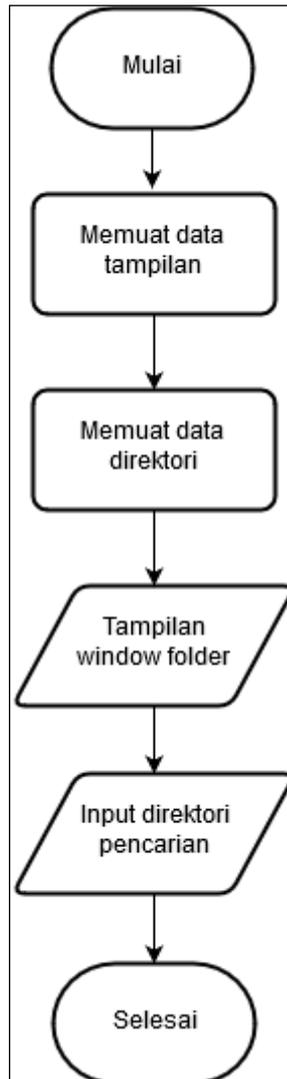
3.2.1 Flowchart

Flowchart adalah diagram yang menjelaskan alur kerja program. Alur dari sistem yang dibuat akan dijelaskan dengan flowchart di bawah ini.



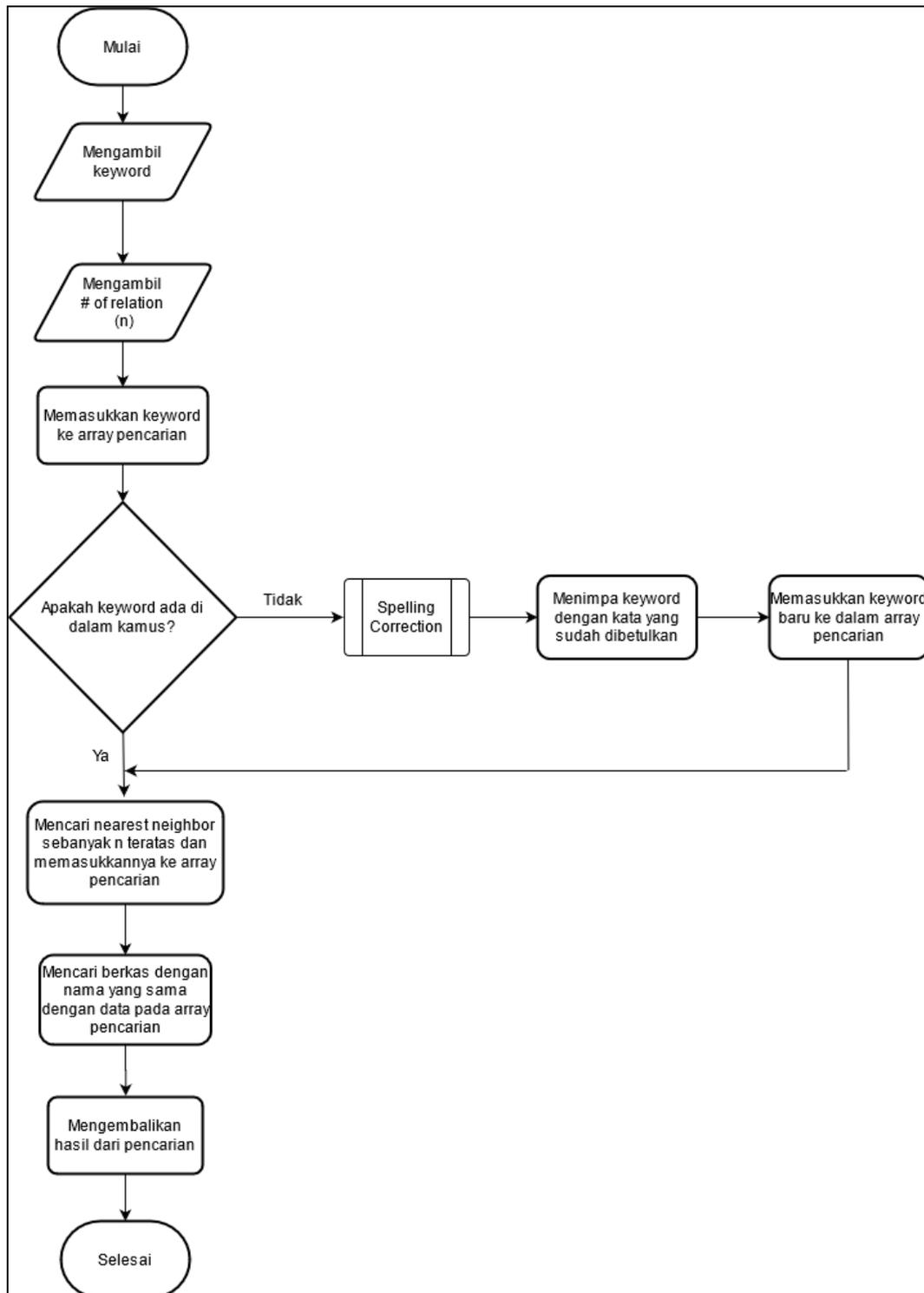
Gambar 3.1 Flowchart Aplikasi

Gambar 3.1 menunjukkan alur yang terjadi pada laman menu utama atau *home* pada aplikasi serta pada aplikasi secara keseluruhan. Sebelum laman ini ditampilkan, aplikasi akan memuat *model* sebagai kamus untuk mencari kata yang memiliki arti mirip dengan *keyword* yang akan dimasukkan oleh pengguna dengan cara menghitung *semantic similarity* dari *keyword* dengan kata yang ada di dalam model. Pencarian dilakukan berdasarkan *keyword* yang dimasukkan oleh pengguna. Pengguna juga diminta memasukkan *# of relation* untuk mencari berapa banyak *nearest neighbor* yang ingin dipakai. Lalu pemilihan *folder* tempat pencarian dengan menekan tombol *browse*. *Keyword*, *# of relation*, dan direktori *folder* akan dipakai oleh proses *search* (pencarian). Hasil dari proses pencarian akan ditampilkan agar pengguna dapat mengakses berkas yang telah selesai dicari.



Gambar 3.2 Flowchart tombol *browse*

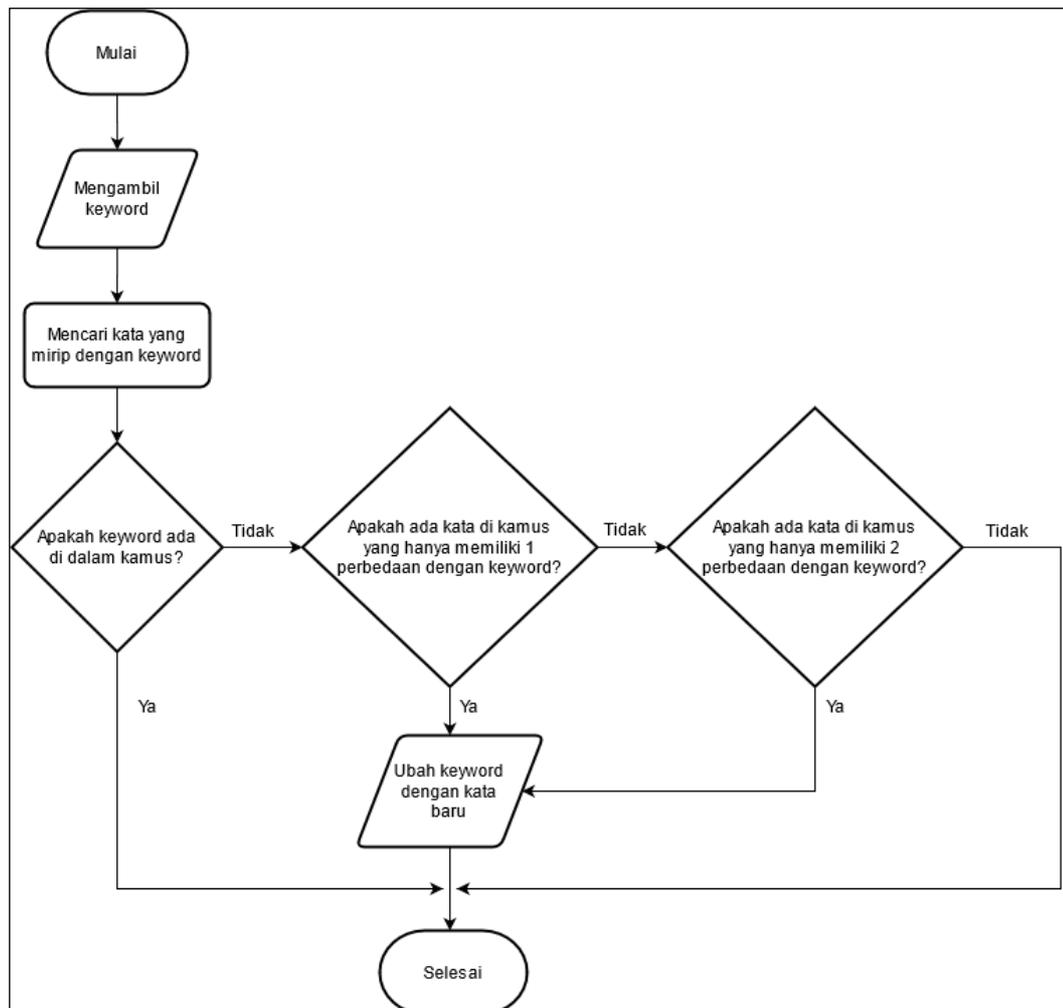
Gambar 3.2 menggambarkan alur dari laman *browse*. Pada proses ini aplikasi akan mengambil *folder* yang ada dalam komputer pengguna. Lalu pengguna akan memilih di *folder* mana pencarian berkas akan dilakukan.



Gambar 3.3 Flowchart Pencarian Berkas

Gambar 3.3 menunjukkan alur pencarian berkas menggunakan *library fastText*. Proses ini akan mengambil *input* dan *# of relation (n)* yang telah dimasukkan oleh pengguna. Lalu aplikasi akan memasukkan *keyword* ke dalam

array pencarian. Lalu aplikasi akan memeriksa apakah *keyword* yang dimasukkan ada di dalam kamus. Jika tidak, aplikasi akan menjalankan *sub-process spelling correction*. Hasil dari *spell correction* akan menimpa *keyword*, lalu dimasukkan ke dalam array pencarian. Setelah proses *spell correction* selesai, proses *searching* akan berlanjut. Semua *nearest neighbor* yang memiliki poin *semantic similarity* n teratas akan dimasukkan ke dalam *array* pencarian, lalu pencarian berkas akan dilakukan dengan cara membandingkan nama berkas yang ada di dalam direktori yang dipilih dengan seluruh kata yang ada di dalam *array* pencarian. Proses ini akan mengembalikan hasil dari pencarian.

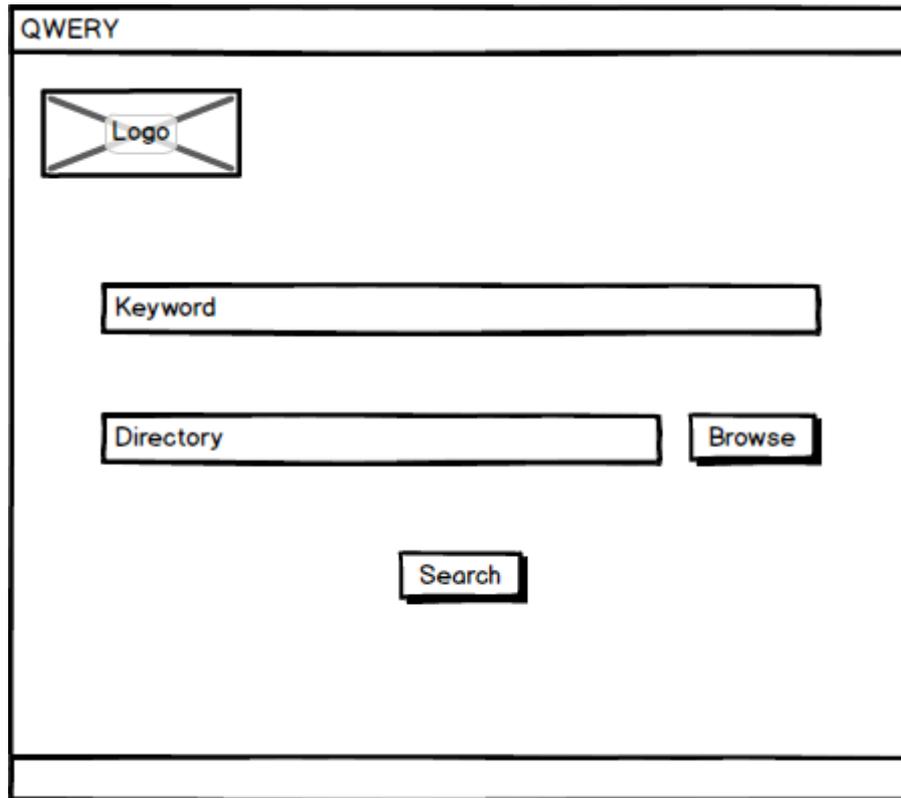


Gambar 3.4 Flowchart *Spelling Correction*

Gambar 3.4 menunjukkan alur proses *spelling correction*. *Keyword* yang masuk ke proses ini kemungkinan memiliki *typo*. Pertama-tama program akan mencari kata yang mirip dengan *keyword*, jika *keyword* ada di dalam kamus, *keyword* tidak akan diganti. Jika tidak, program akan mencari kata dalam kamus yang mempunyai perbedaan 1 karakter dengan *keyword*. Jika program menemukan kata tersebut di dalam kamus, *keyword* akan diganti dengan kata tersebut. Tetapi jika program tidak menemukan kata dengan perbedaan 1 karakter, program akan mencari kata dengan perbedaan 2 karakter, jika menemukannya, program akan mengganti *keyword* dengan kata tersebut. Jika perbedaan 1 karakter dan 2 karakter tidak ditemukan, maka program akan berasumsi bahwa kata tersebut adalah kata yang jarang dipakai atau *rare word*. *Rare word* akan dibiarkan menjadi *keyword*.

3.2.3 Perancangan Antarmuka

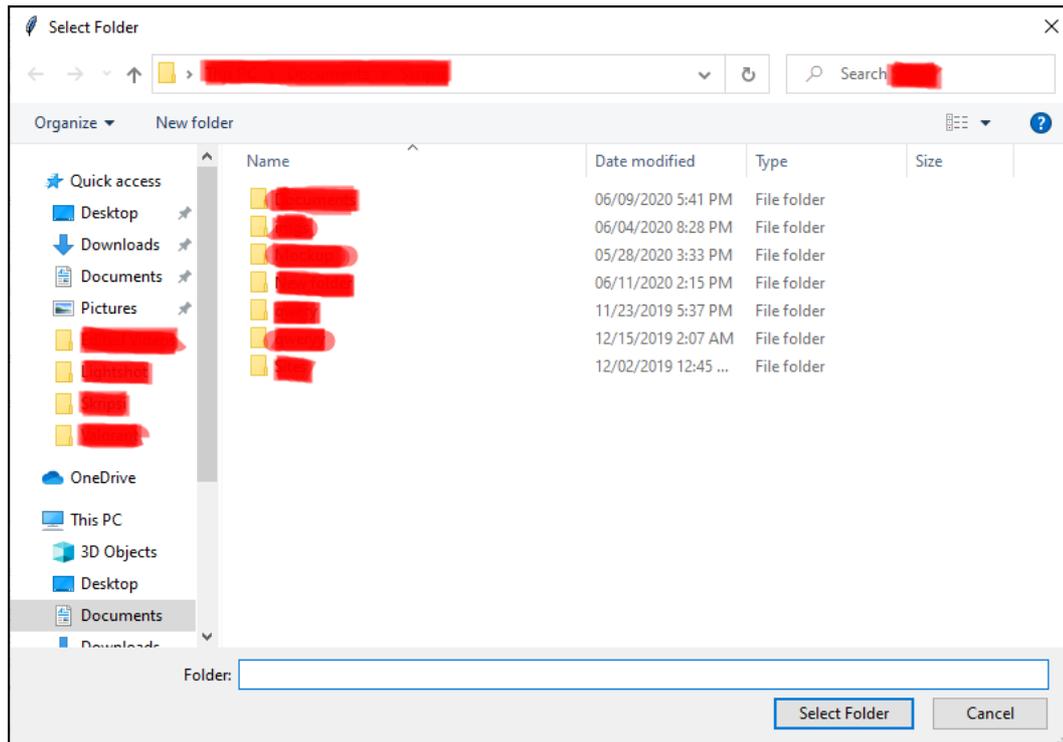
- a. Halaman Utama



Gambar 3.5 Rancangan Antarmuka Halaman Utama

Gambar 3.5 menunjukkan rancangan laman utama pada aplikasi. Pada laman ini terdapat *text box keyword* sebagai media agar pengguna dapat menuliskan kata yang ingin dicari dalam berkas pada komputer. *Text box directory* merupakan *disabled text box* di mana pengguna tidak dapat mengetik apapun pada *text box* tersebut dikarenakan pemilihan direktori dilakukan secara otomatis pada tombol *browse*. Setelah kedua *text box* terisi, pencarian dapat dilakukan dengan menekan tombol *search*.

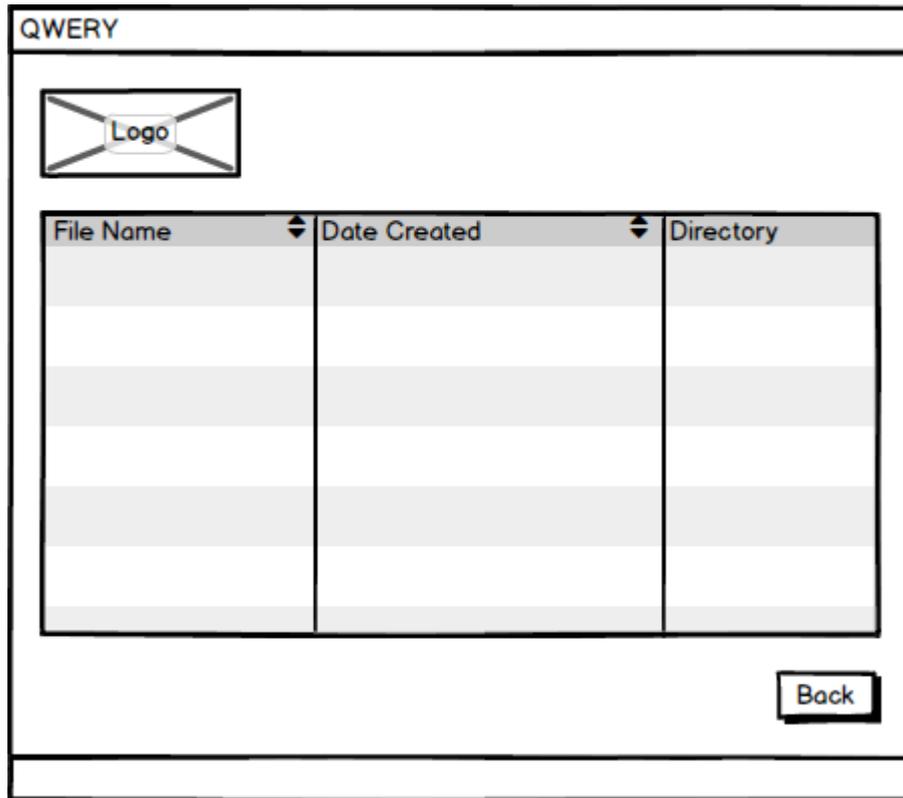
- b. Browse Folder



Gambar 3.6 Rancangan Antarmuka *Browse Folder*

Gambar 3.6 menunjukkan laman yang akan muncul jika tombol *browse* pada halaman utama ditekan. Halaman ini merupakan halaman *browse folder* bawaan Windows. Halaman ini akan mengarahkan pengguna untuk memilih direktori *folder* yang menjadi ruang pencarian.

c. Halaman Hasil Pencarian



Gambar 3.7 Rancangan Antarmuka Hasil Pencarian

Gambar 3.7 menunjukkan rancangan antarmuka laman hasil pencarian. Laman ini akan muncul setelah proses pencarian selesai. Laman ini akan menunjukkan hasil dari pencarian *keyword* yang dimasukkan oleh pengguna beserta dengan kata yang memiliki arti mirip dengan nama berkas yang ada di dalam komputer pengguna.