



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, 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

ANALISIS DAN PERANCANGAN APLIKASI

3.1 Metode Penelitian

Untuk melakukan penelitian ini, peneliti melakukan beberapa metode penelitian seperti:

1. Studi Literatur

Yaitu dengan melakukan tinjauan pustaka untuk mendapatkan informasi berupa data, tulisan, keterangan melalui buku, jurnal atau sumber informasi lainnya yang mendukung landasan teori dalam Tugas Akhir ini. Selain itu juga mencari informasi terkini melalui internet sehingga data teori yang didapat mengikuti perkembangan teknologi.

2. Analisis algoritma

Pada tahap ini akan dilakukan proses analisis konsep *Burrows-Wheeler* yang meliputi algoritma *Burrows-Wheeler Transform* (BWT), *Move-To-Front* (MTF), *Huffman Coding* meliputi karakteristik masing-masing algoritma, sehingga nantinya dapat diimplementasikan dengan baik pada sistem.

3. Perancangan Program

Dari hasil analisis yang dilakukan selanjutnya dibuat rancangan untuk membangun aplikasi. Rancangan yang dibuat meliputi *interface*, proses dan alur program.

4. Implementasi

Implementasi merupakan tahap mengimplementasikan hasil rancangan yang dilakukan pada tahap sebelumnya. Pada tahap ini, setiap pustaka yang digunakan, digabungkan dan disesuaikan dengan kebutuhan perangkat lunak. Selain itu, setiap modul kelas, fungsi, antarmuka, dan sebagainya juga dibuat.

5. Pengujian dan Analisis Program

Pada tahap ini akan dilakukan analisis terhadap fokus permasalahan penelitian, yaitu apakah sistem yang dibuat berhasil atau tidak, serta perbaikan bila terdapat kesalahan.

3.2 Analisis

Dalam penelitian yang dilakukan, algoritma yang digunakan adalah algoritma Burrows-Wheeler, dimana algoritma Burrows-Wheeler terdiri dari tiga tahap metode yang dilakukan secara berurutan yaitu *Burrows-Wheeler Transform* (BWT), *Move-To-Front* (MTF), dan *Huffman Coding*. Untuk membangun aplikasi kompresi dengan berbasis *desktop*, maka aplikasi yang dibangun menggunakan bahasa pemrograman C# serta dengan menggunakan *framework* .NET yaitu *framework* 4.5. Selain itu aplikasi ini menggunakan *library* DropNet yaitu suatu *client library* yang berfungsi untuk menghubungkan aplikasi dengan Dropbox. Aplikasi yang dibangun merupakan aplikasi pada *windows operating system* dan melalui aplikasi Dropbox yang terintegrasi dengan *Windows*. Dimana pada saat *file* yang telah dipilih akan diunggah, lalu melalui tahapan kompresi setelah itu *file* tersebut diunggah, sedangkan pada saat *file* yang telah dipilih akan diunduh

dilakukan dekompresi terlebih dahulu baru akan diunduh. Aplikasi ini nantinya akan diberi nama dengan PressYourBox.

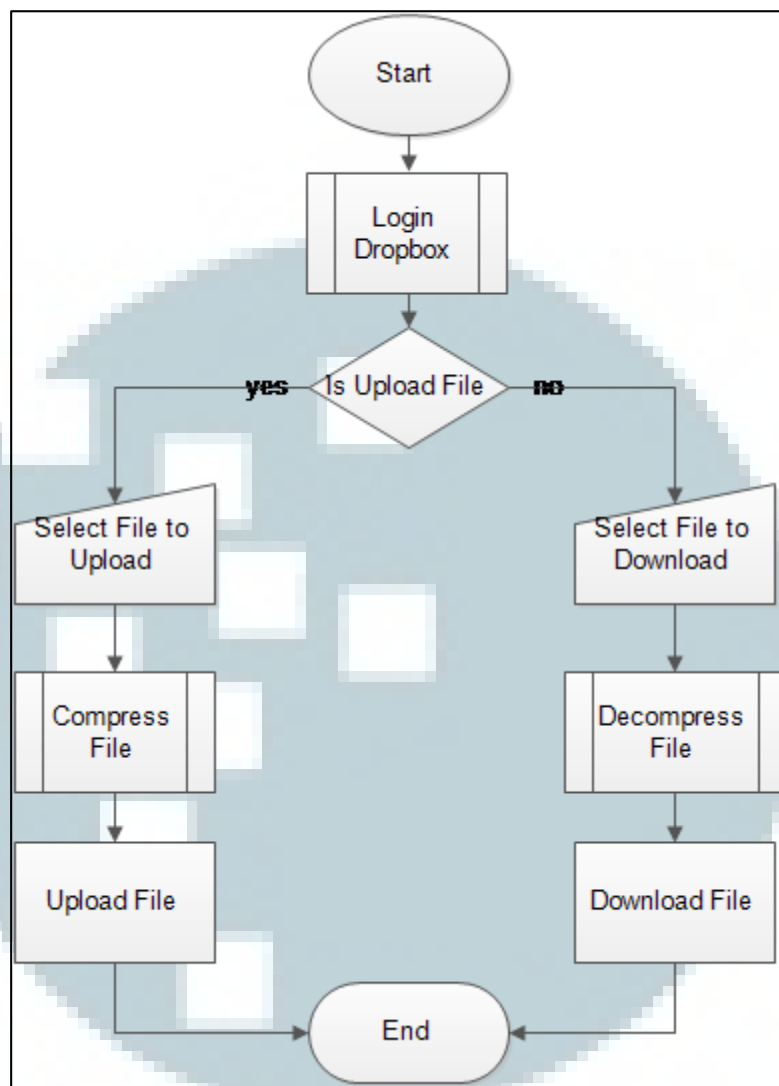
3.3 Perancangan

Pada subbab ini dipaparkan hasil rancangan untuk aplikasi PressYourBox baik itu *flowchart* sistem, fungsionalitas sistem, masukan dan keluaran sistem, dan desain modul.

3.3.1 Flowchart Sistem

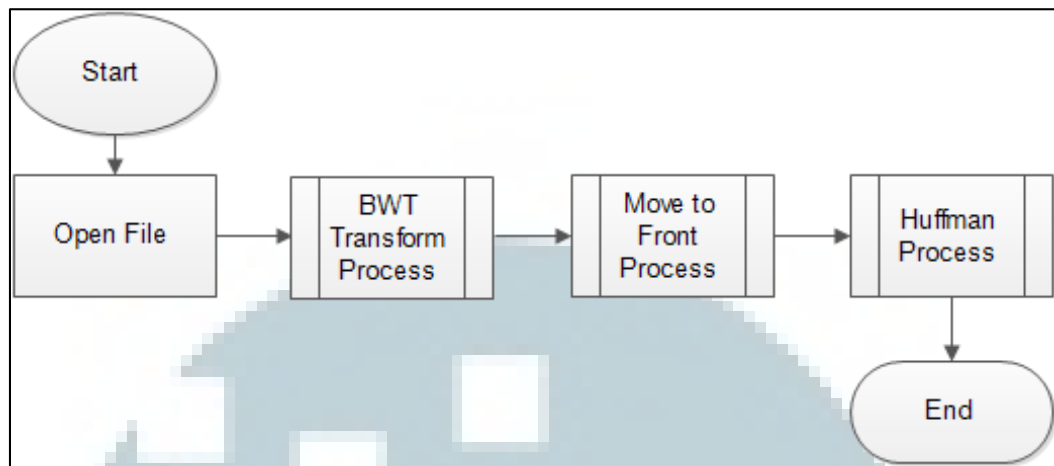
Proses kerja aplikasi yang dibuat memiliki *flowchart* seperti gambar di bawah ini. Ketika aplikasi dijalankan terdapat *form* yang meminta *email* serta *password* dropbox yang digunakan pengguna sebagai proses untuk *login* ke dalam dropbox. Setelah itu pengguna dapat *allow* aplikasi PressToBox untuk terkoneksi dengan dropbox pengguna. Setelah itu akan muncul *form* baru yang berisi dua *tab* yaitu *tab* kompresi dan *tab* dekompresi, serta secara *default* akan berada pada *tab* kompresi. Selanjutnya pengguna dapat memilih *file* yang akan diunggah atau diunduh.

UMN



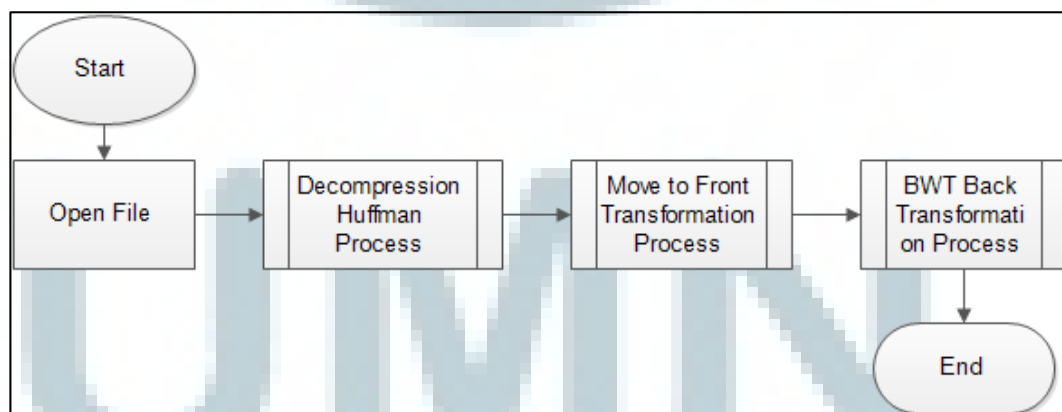
Gambar 3.1 *Flowchart* sistem

Pada saat *file* diunggah, maka aplikasi akan melakukan kompresi *file* terlebih dahulu menggunakan modul *Burrows-Wheeler Transform* (BWT), *Move-To-Front* (MTF), dan *Huffman Coding* secara berurutan, kemudian diunggah ke dropbox.



Gambar 3.2 Flowchart kompresi file

Apabila pengguna mengunduh *file* yang telah dikompresi, maka setelah *file* tersebut berhasil diunduh, *file* tersebut akan didekompresi menggunakan modul *Huffman Coding*, *Move-To-Front* (MTF), dan *Burrows-Wheeler Backtransform* (BWT) secara berurutan, setelah itu diletakkan di *folder* PressYourBox. Setelah semua proses telah dilakukan, maka akan memunculkan suatu *messagebox* untuk menunjukkan *file* yang telah diproses dengan waktu yang dibutuhkan.



Gambar 3.3 Flowchart dekompresi file

3.3.2 Fungsionalitas Sistem

Sistem pada aplikasi ini memiliki fungsionalitas sebagai berikut:

1. Menerima *input file stream* yang akan diubah dalam bentuk *array of byte*.
2. Pada modul Burrows-Wheeler Transform akan dilakukan pengurutan *input* karakter yang identik berurutan.
3. Pada modul *Move to Front* akan mengganti karakter yang telah diurutkan pada proses Burrows-Wheeler Transform menjadi karakter *null*.
4. Melakukan penyusunan *bit* dalam proses Huffman berdasarkan hasil proses *Move to Front*.

3.3.3 Masukan dan Keluaran Sistem

Masukan pada sistem aplikasi ini adalah sebagai berikut:

1. Masukkan pada sistem aplikasi secara umum adalah *file* yang akan dikompresi ataupun didekompresi.
2. Sistem akan menerima alamat *file* yang akan dikompresi ataupun didekompresi
3. Masukkan pada modul Burrows-Wheeler Transform adalah *array of byte* dari *file* yang akan dikompresi dan *array of byte* dari *output* dari modul *Move to Front* pada saat dekompresi.
4. Masukkan pada modul *Move to Front* adalah *array of byte* dari *output* Burrows-Wheeler Transform pada saat kompresi dan *array of byte* dari *output* Huffman Code pada saat dekompresi.

5. Masukkan pada modul *Huffman Code* adalah *array of byte* dari *output Move to Front* pada saat kompresi dan *array of byte* dari *file* yang akan didekompresi.

Sedangkan untuk keluaran pada sistem aplikasi adalah sebagai berikut:

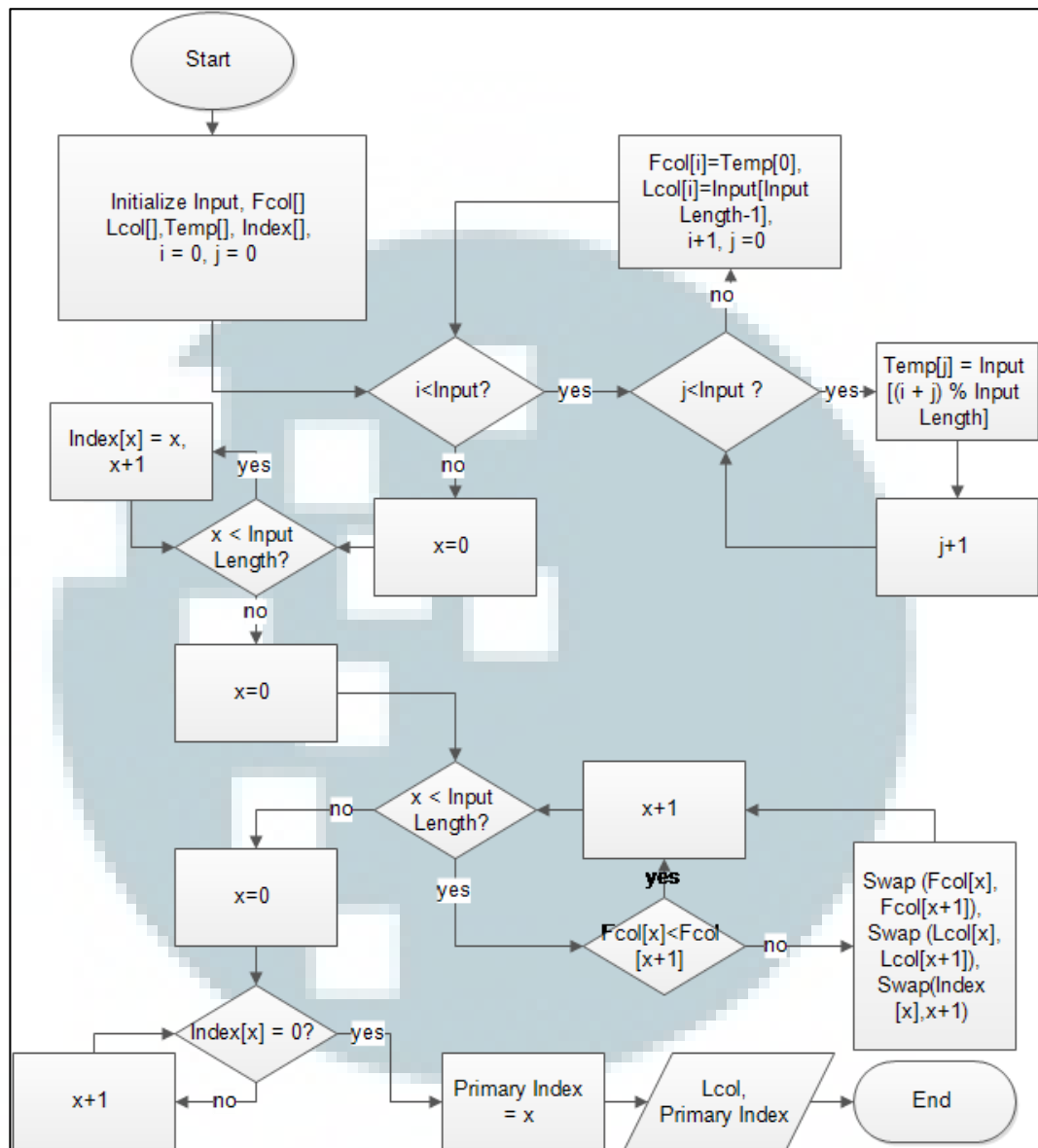
1. Keluaran dari sistem aplikasi secara umum adalah *file* yang telah melalui proses kompresi ataupun dekompresi.
2. Keluaran dari modul Burrows-Wheeler adalah *array of byte* pada saat kompresi dan *array of byte* suatu *file* hasil proses dekompresi dengan ekstensi *file* awal pada saat dekompresi.
3. Keluaran dari modul *Move to Front* adalah *array of byte* pada saat kompresi ataupun dekompresi.
4. Keluaran dari modul *Huffman Code* adalah *array of byte* suatu *file* hasil proses kompresi dengan ekstensi *file* .pyb pada saat kompresi dan *array of byte* pada saat dekompresi.

3.3.4 Desain Modul

Aplikasi ini terdapat tiga modul, yaitu modul *Burrows-Wheeler Transform* (BWT), modul *Move-To-Front* (MTF), dan modul *Huffman*. Modul ini dilakukan secara berurutan untuk melakukan kompresi maupun dekompresi. Berikut ini penjelasan rancangan dari masing-masing modul yang disertai dengan diagram alir.

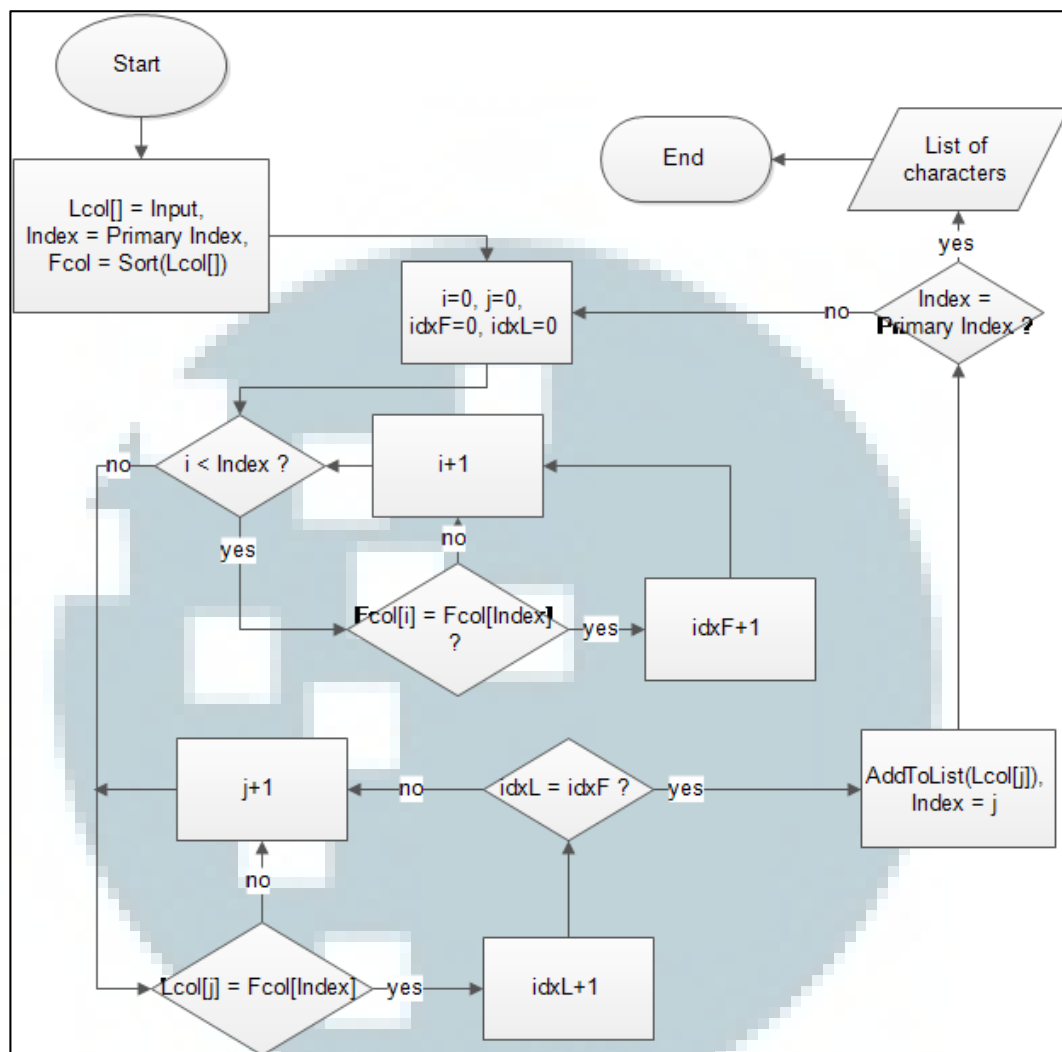
1. Modul Burrows-Wheeler Transform

Pada modul ini merupakan penerapan dari algoritma Burrows-Wheeler Transform. Pada proses kompresi dengan modul ini hal yang pertama dilakukan adalah membaca *byte* yang terdapat pada *file* yang ingin dikompresi, kemudian hasil pembacaan *byte* tersebut akan disimpan dalam *array of byte*. Selanjutnya, fungsi BWT_Transform akan dipanggil untuk membentuk suatu *array of byte* untuk menampung kolom matriks yang terakhir. Matriks diasumsikan ada, namun tidak tertulis dalam program untuk menghemat pemakaian memori komputer. Sehingga yang ditampung adalah kolom terakhir matriks yang ditulis dalam bentuk *array of byte*. Setelah itu dibentuk lagi *array of byte* dengan ukuran panjang yang sama dengan panjang *array of byte* dari *file* yang dibaca dan diisi dengan angka indeks dari angka nol sampai panjang ukuran *array of byte*. *Array of byte* ini disebut indeks. Hal ini digunakan untuk mengetahui letak karakter pertama dari *array of byte file* yang dibaca setelah dilakukan pengurutan secara *lexicographical*. Setelah itu dilakukan pengurutan karakter pada kolom pertama matriks yang isinya sama dengan *array of byte* dari *file* yang dibaca, namun *array of byte* dari kolom akhir matriks dan *array of byte* indeks diurutkan sesuai dengan posisi kolom pertama matriks atau *array of byte* dari *file* yang dibaca.



Gambar 3.4 *Flowchart fungsi Burrows-Wheeler Transform*

UMN

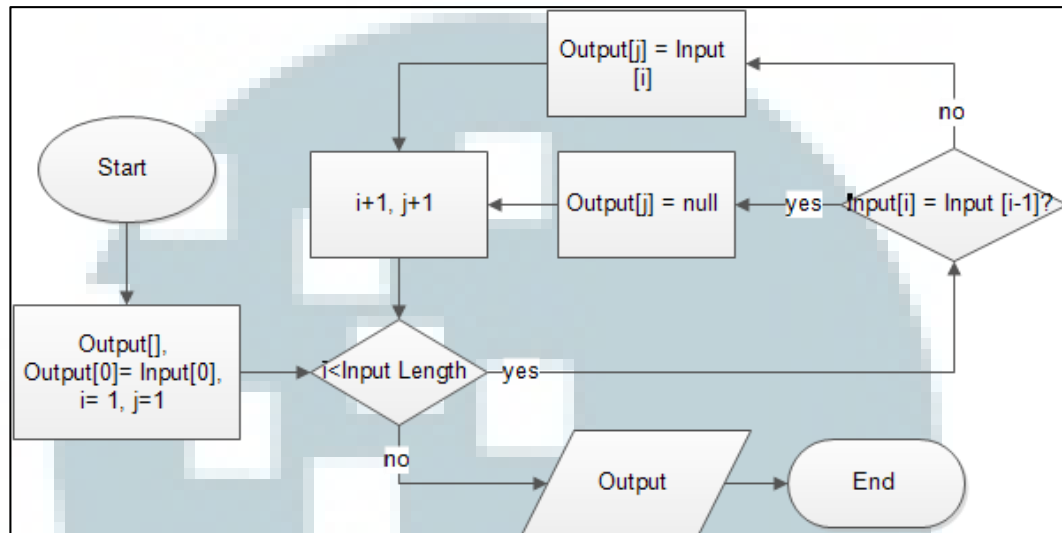


Gambar 3.5 Flowchart fungsi Burrows-Wheeler *BackTransformation*

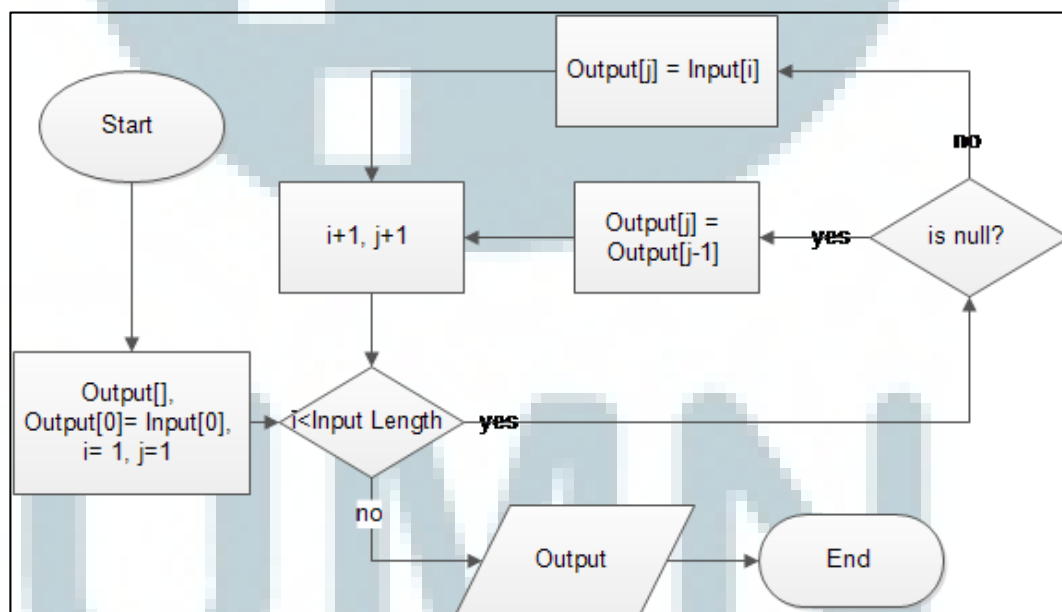
2. Modul Move to Front

Modul *Move to Front* merupakan modul yang menerapkan algoritma *Move to Front*. Pada fungsi yang berada dalam modul ini berguna sebagai proses untuk mengubah karakter yang berurutan menjadi karakter *null*. Modul ini menerima masukan *array of byte*, lalu dari indeks pertama *array* tersebut sampai akhir akan dilakukan pengecekan karakter yang identik berurutan. Setiap menemukan karakter yang identik berurutan, maka karakter yang pertama ditampung dan karakter selanjutnya diubah menjadi karakter *null* sampai menemukan karakter

yang berbeda. Pada saat proses dekompresi, proses yang dilakukan hampir sama, namun setiap menemukan karakter *null* setelah karakter selain *null* maka karakter *null* tersebut akan diubah menjadi karakter sebelum karakter *null* tersebut.



Gambar 3.6 Flowchart fungsi Move to Front Encode

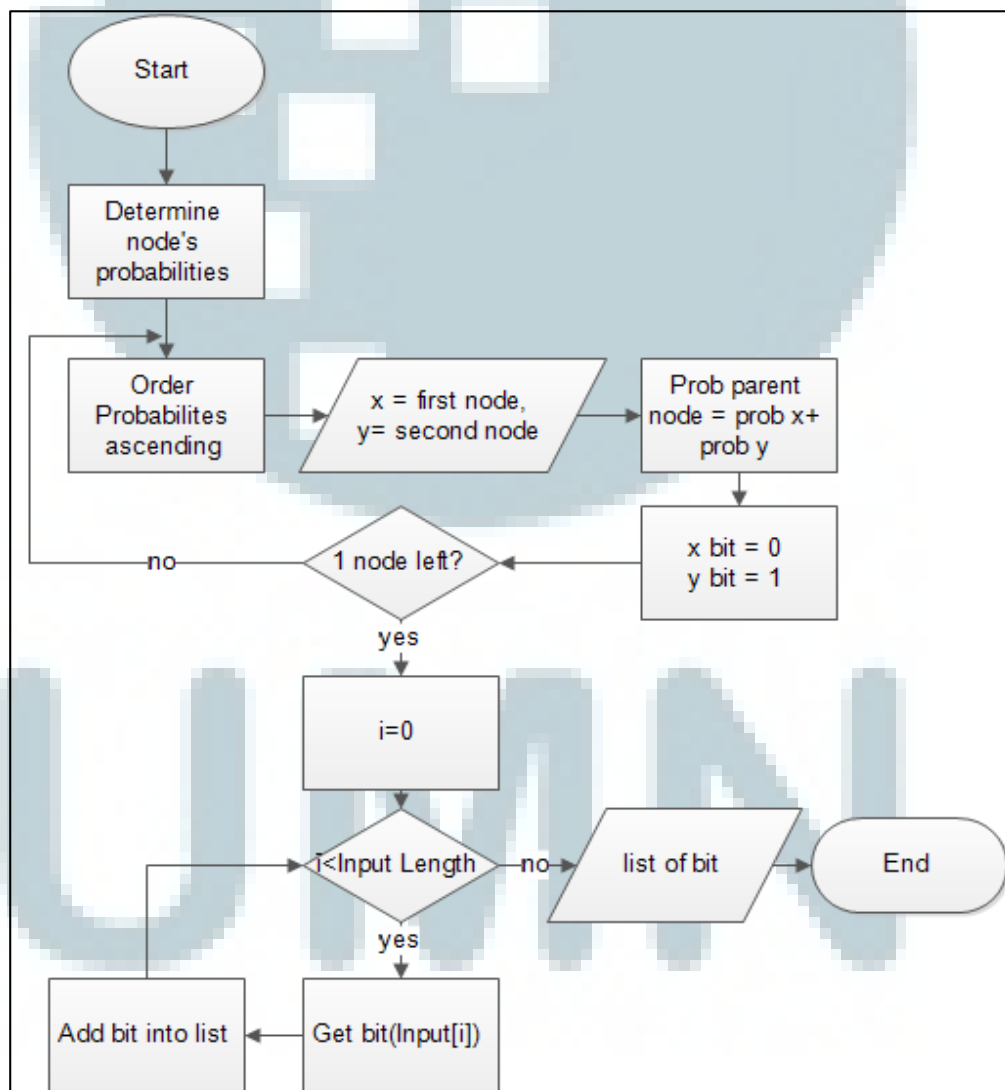


Gambar 3.7 Flowchart fungsi Move to Front Decode

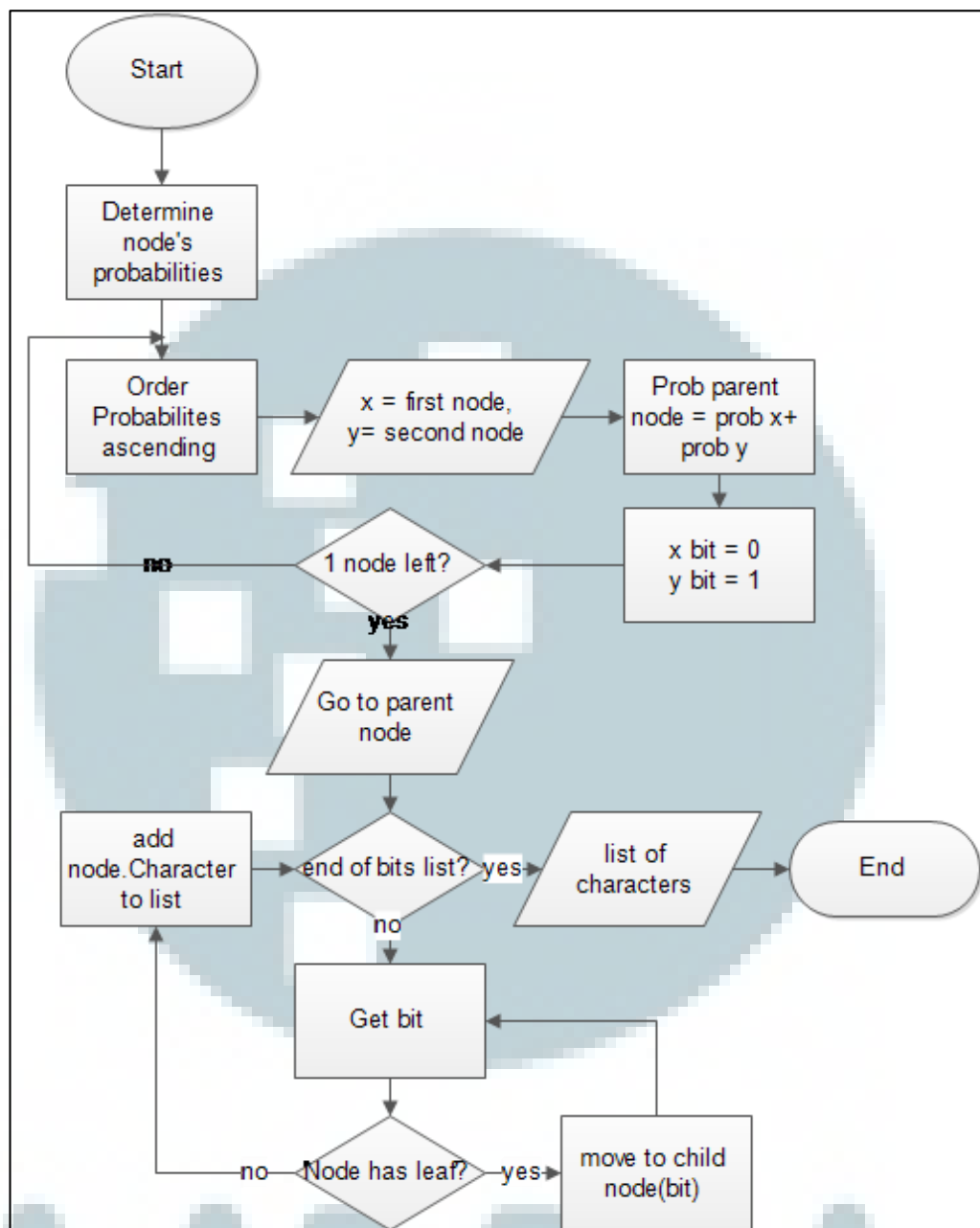
3. Modul Huffman Code

Modul Huffman Code, modul yang menerapkan algoritma Huffman Code. Kegunaan dari modul ini adalah untuk mengurangi panjang *bit*. Dengan

menggunakan aturan Huffman, proses kompresi pertama kali akan membuat tabel frekuensi sesuai frekuensi pada setiap karakter dan membuat pohon *tree nodes* dimana letak tiap *node* ditentukan dari frekuensi yang lebih sering muncul. Setelah itu dilakukan penentuan panjang *bit* sesuai dengan urutan karakter masukan. Pada proses dekompresi, hal yang pertama kali dilakukan adalah membaca tabel frekuensi yang disimpan pada *file* dan membaca panjang *bit* hasil kompresi, lalu membuat pohon Huffman berdasarkan tabel frekuensi. Panjang *bit* tersebut diterjemahkan sesuai dengan aturan Huffman.



Gambar 3.8 Flowchart fungsi Huffman Code

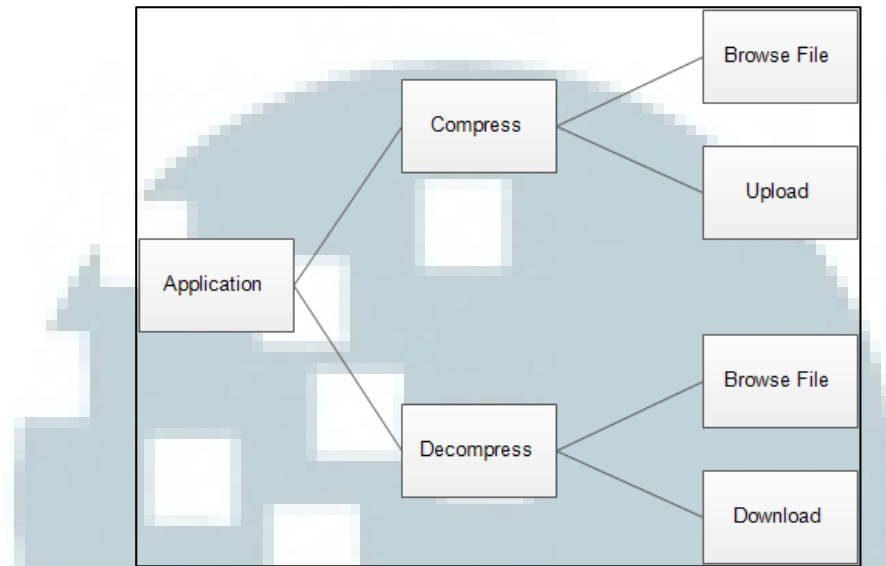


Gambar 3.9 Flowchart fungsi *Decompression Huffman Code*

3.3.5 Hirarki Menu Aplikasi

Pada gambar di bawah ini diperlihatkan rancangan hirarki menu dari aplikasi yang dibangun. Saat aplikasi dijalankan dan berhasil *login*, maka pengguna dapat memilih untuk melakukan kompresi beserta *upload file* yang

dipilih dengan cara memilih *compress tabpage* atau melakukan *download file* beserta dekompresi *file* dengan cara memilih *decompress tabpage*.



Gambar 3.10 Hirarki Menu Aplikasi PressYourBox

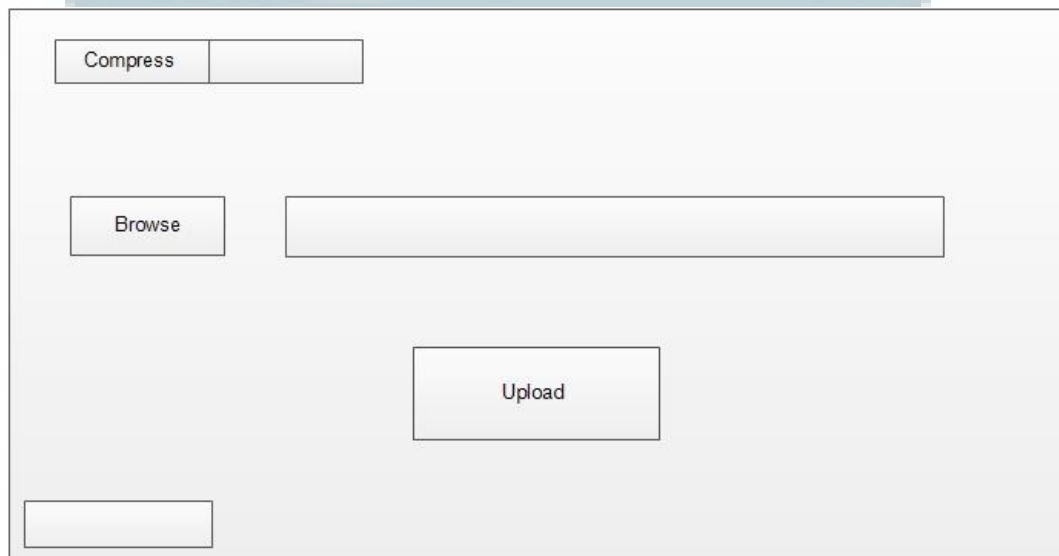
3.3.6 Tampilan Antarmuka

Berikut akan dijelaskan rancangan tampilan antarmuka pada aplikasi yang dibangun. Pada tampilan awal akan menampilkan menu *login* ke dropbox dengan tampilan *form email* dan kata sandi serta sebuah tombol. Rancangan antarmuka *login* diperlihatkan pada gambar 3.11.



A login form interface with a light gray background. It contains two input fields: the top one is labeled "Email" and the bottom one is labeled "Kata sandi". To the right of the "Kata sandi" field is a button labeled "Masuk".

Gambar 3.11 Tampilan *Login*



A "Compress" tab page interface with a light gray background. At the top left is a button labeled "Compress" next to a small input field. Below this is a "Browse" button next to a larger input field. In the center is a button labeled "Upload". At the bottom left is a small input field.

Gambar 3.12 Tampilan *Compress tabpage*



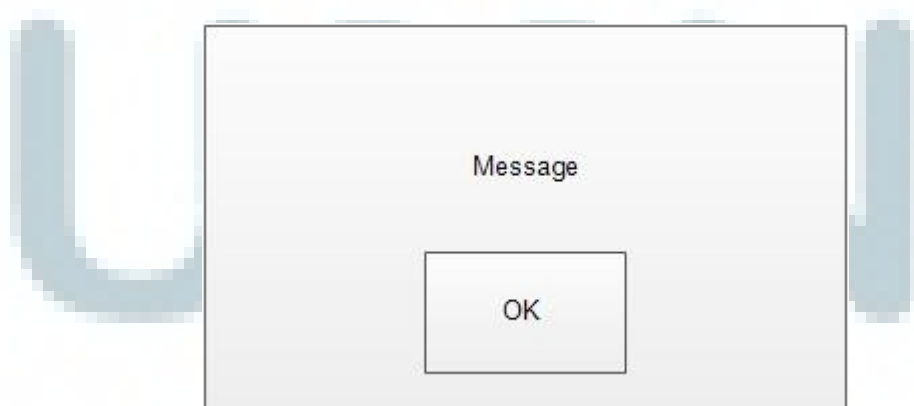
Gambar 3.13 Tampilan *Decompress tabpage*

Gambar 3.12 dan gambar 3.13 merupakan tampilan *tabpage* dalam satu *form* setelah berhasil melakukan *login* ke dropbox. *Tabpage* terdiri dari dua *tabpage*, yaitu *compress* dan *decompress*. Pada *compress tabpage* terdapat tombol untuk memilih *file* yang akan dikompresi lalu akan diunggah jika tombol *download* dalam *tabpage* tersebut ditekan. Pada *decompress tabpage* terdapat tombol *browse* yang berfungsi untuk memilih *file* dengan ekstensi *file .pyb*, setelah itu tombol *download* berfungsi untuk dekompresi *file* yang telah dipilih. Pada saat berlangsungnya proses kompresi ataupun dekompresi terdapat *progress bar* sebagai informasi sejauh mana proses berlangsung.



Gambar 3.14 Tampilan pemilihan *file*

Gambar 3.14 merupakan tampilan pencarian *file* yang akan dikompresi atau didekompresi. Pada saat kompresi, maka akan membuka *path folder* di komputer dan terdapat *filter* untuk membedakan jenis *file* teks, gambar, audio dan video. Sedangkan saat dekompresi akan membuka *path folder* dropbox dan *file* yang dapat dipilih hanya .pyb.



Gambar 3.15 Tampilan *Message box*

Gambar 3.15 merupakan tampilan yang muncul setelah proses kompresi maupun dekompresi selesai. Tampilan berfungsi untuk memberikan informasi nama *file* yang telah diproses dan lama waktu yang dibutuhkan untuk kompresi maupun dekompresi *file*, tanpa disertai waktu *upload* maupun *download file*.

