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.

Team project ©2017
Dony Pratidana S. Hum | Bima Agus Setyawan S. IIP

BAB III

METODOLOGI DAN PERANCANGAN

3.1 Metode Penelitian

Penelitian implementasi algoritma kompresi Deflate pada aplikasi pengiriman dan penerimaan *file* dengan Android Beam pada sistem operasi Android melalui beberapa langkah penelitian sebagai berikut.

1. Studi Literatur

   Sebelum melakukan perancangan aplikasi terlebih dahulu dilakukan studi literatur dan kepustakaan dari berbagai sumber guna mendukung proses penelitian. Sumber yang didapat berasal dari jurnal – jurnal penelitian orang lain, artikel yang berkaitan, serta laporan baik itu berupa media cetak ataupun media internet.

2. Perancangan Aplikasi

   Perancangan aplikasi terdiri dari perancangan sistem aplikasi dan *user interface*. Perancangan sistem dapat dijabarkan menggunakan *flowchart*.

3. Pembuatan Aplikasi

   Pembuatan aplikasi dimulai dengan melakukan pembuatan aplikasi serta pengimplementasian algoritma kompresi Deflate kedalam sistem yang diikuti dengan pembuatan *user interface* aplikasi. Pembuatan aplikasi ini dipandu oleh rancangan sistem yang sudah dibuat sebelumnya sehingga pembuatan aplikasi menjadi lebih terstruktur. Pembuatan aplikasi dilakukan dengan menggunakan aplikasi IDE sistem operasi Android yaitu

Implementasi Algoritma ..., Yuddistira Kiki, FTI UMN, 2014

Android Studio versi 0.1.6 dengan bahasa pemrograman Java. JRE yang digunakan adalah versi 1.7.0.

4. Implementasi dan uji Coba

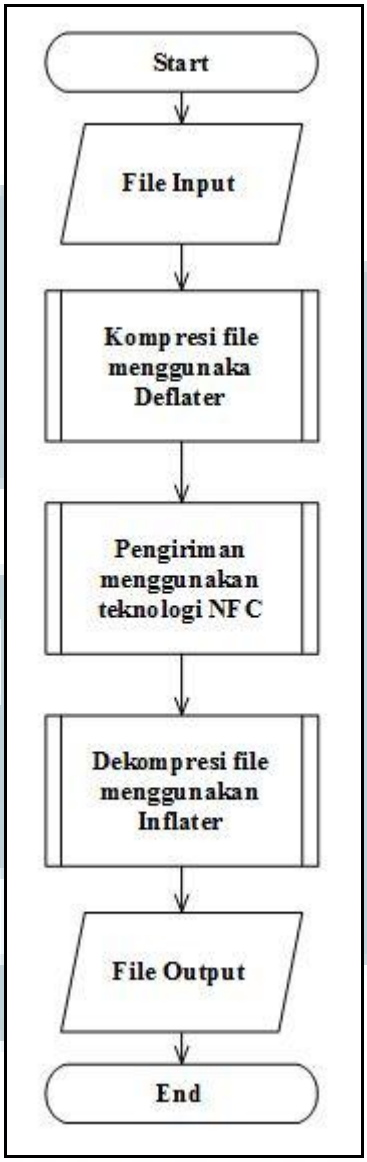
Uji coba dilakukan untuk menghindari adanya sistem aplikasi yang tidak berjalan seharusnya. Dengan uji coba bisa didapat kekurangan apa yang dimiliki sistem aplikasi yang sedang dibangun. Uji coba dilakukan dengan beberapa jenis *file* yang sudah disiapkan yang nantikan akan dicatat waktu proses aplikasi dan ukuran *file output*. Nantinya data tersebut akan dianalisis dan dibandingkan untuk dicapainya suatu kesimpulan guna mencapai tujuan penelitian ini.

5. Penulisan Laporan

Melakukan penulisan laporan dari hasil penelitian yang telah dilakukan dimulai dari tahapan studi literatur, analisis dan perancangan aplikasi, pembuatan aplikasi dan uji coba aplikasi.

3.2 Perancangan Aplikasi

Sistem Aplikasi memiliki tahapan – tahapan penting dalam menjalankan prosesnya seperti pada gambar 3.1.



Gambar 3.1. Urutan proses sistem aplikasi pengiriman dan penerimaan *file*.

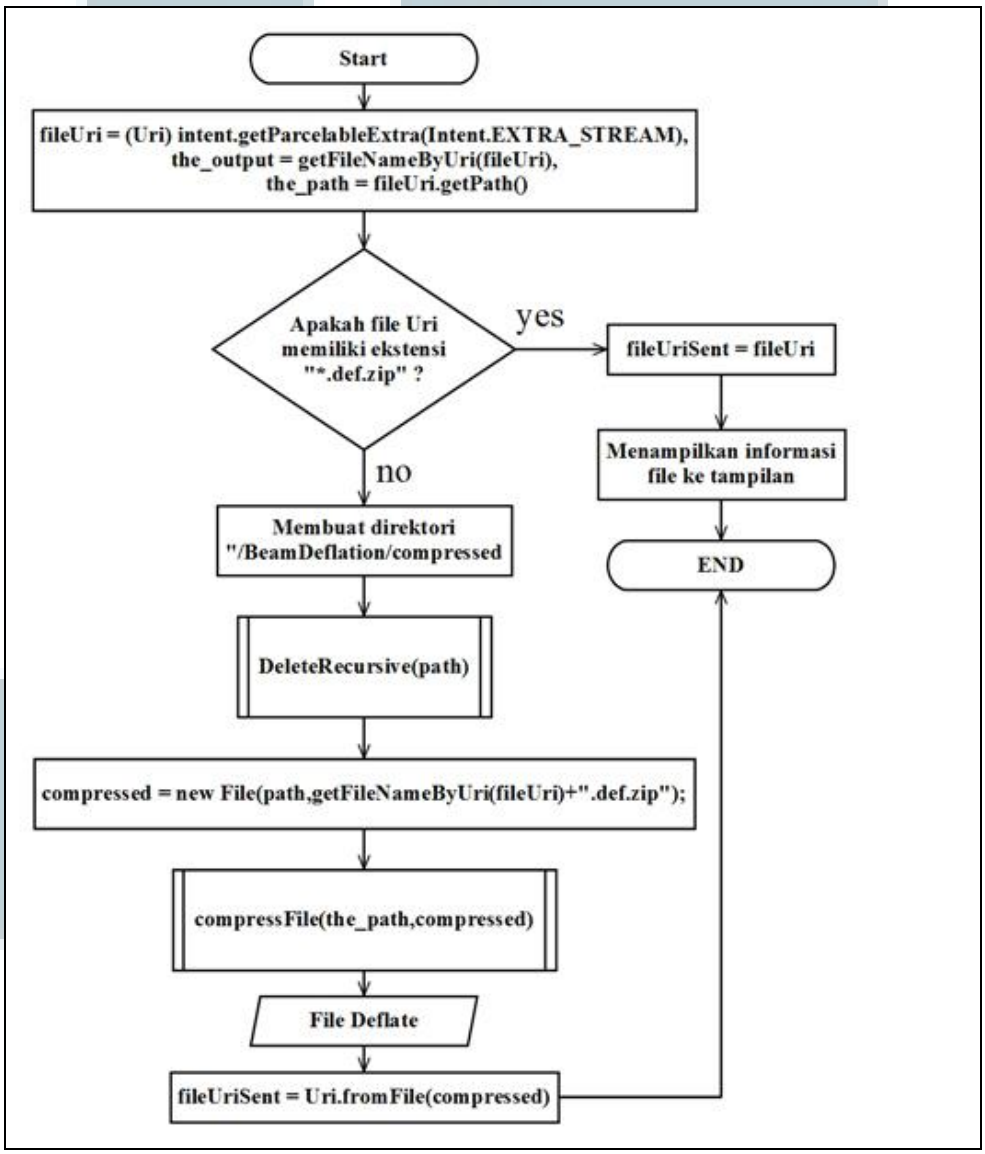
1. *File Input*

Memasukkan *file* ke dalam sistem aplikasi yang ingin diproses dan dikirimkan dengan menggunakan Android Beam.

2. Desain *flowchart* kompresi *file* menggunakan Deflater

Kompresi ini dilakukan dengan bantuan Deflater yang dimiliki oleh sistem operasi Android. *File* diproses untuk mendapatkan data *properties*. Data tersebut

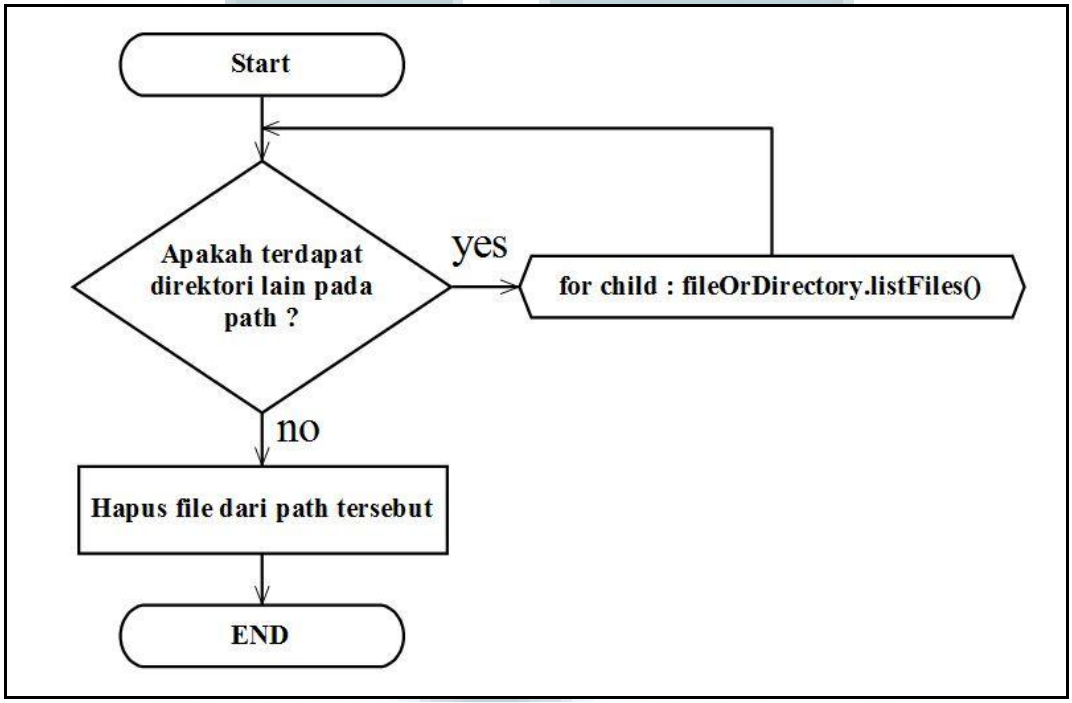
akan digunakan untuk mendapatkan *path file* tersebut. Lalu dilakukan pengecekan apakah *file* yang akan dikirimkan ini merupakan *file* yang masih utuh atau *file* yang sudah hasil kompresi sebelumnya dengan aplikasi ini. Setelah itu dilakukan kompresi dengan menggunakan bantuan Deflater. Hasil kompresi ini akan menghasilkan *output* yang berasal dari penulisan *byte* kembali ke dalam *file baru* yang sudah ditentukan. Proses ini dapat dijabarkan lebih jelas dalam *flowchart* pada gambar 3.2.



Gambar 3.2. *Flowchart* kompresi.

3. Desain *flowchart* subproses DeleteRecursive

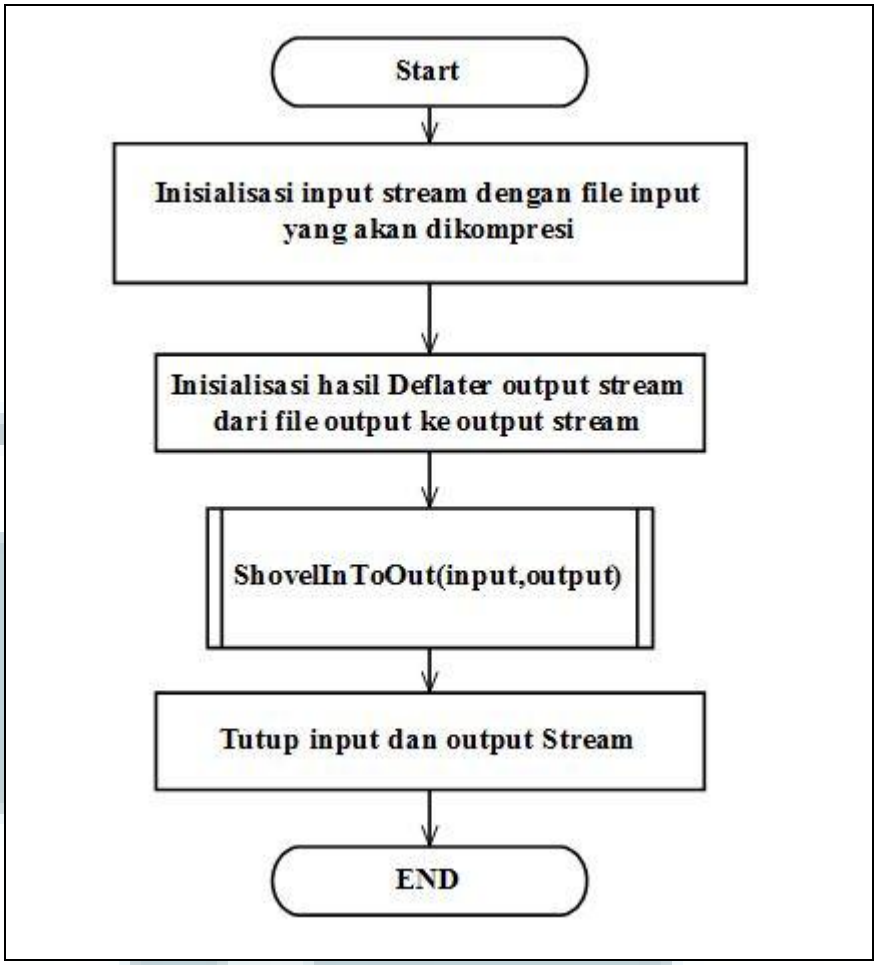
Gambar 3.3 berikut merupakan proses subproses dari fungsi DeleteRecursive. Fungsi ini akan menghapus semua file yang terdapat pada alamat direktori yang sudah ditentukan.



Gambar 3.3. *Flowchart* subproses DeleteRecursive.

4. Desain *flowchart* subproses compressFile

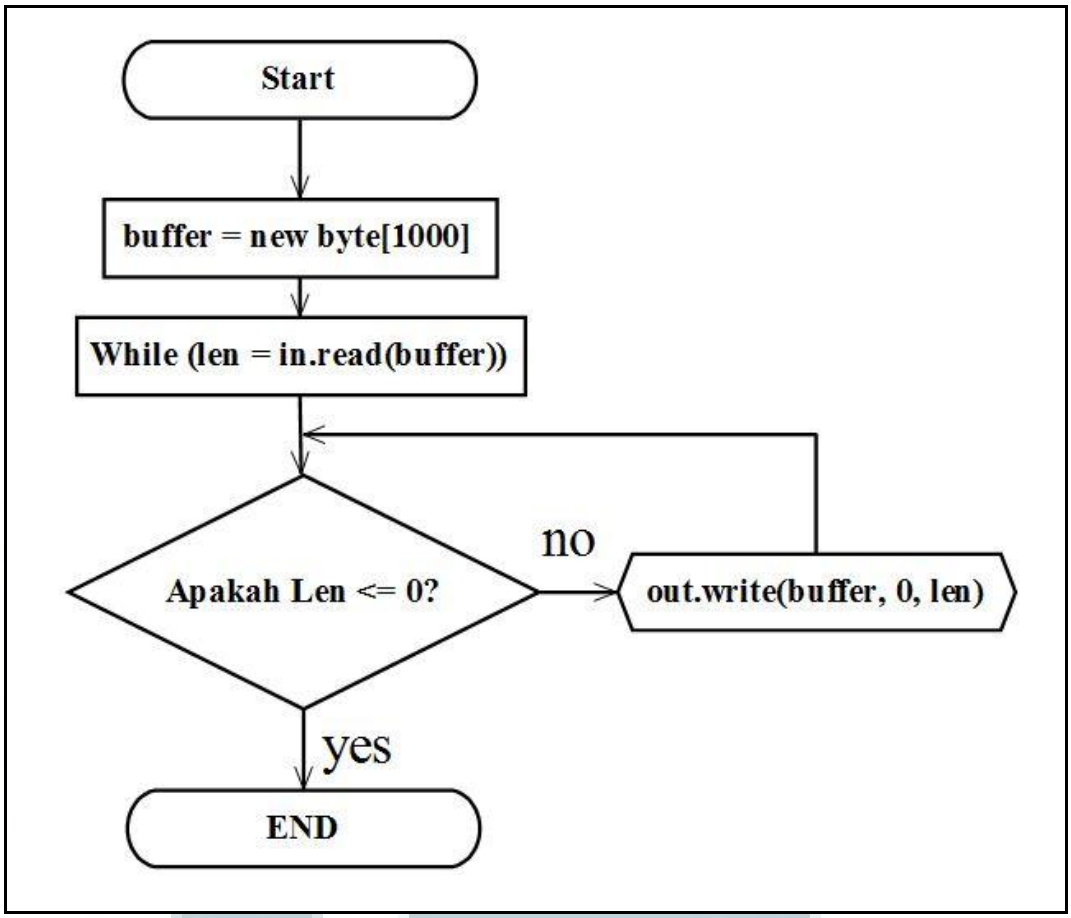
Gambar 3.4 menampilkan *flowchart* subproses compressFile. Fungsi subproses compressFile adalah melakukan kompresi file asli menjadi file kompresi.



Gambar 3.4. Desain *flowchart* subproses compressFile.

5. Desain *flowchart* subproses ShovelInToOut.

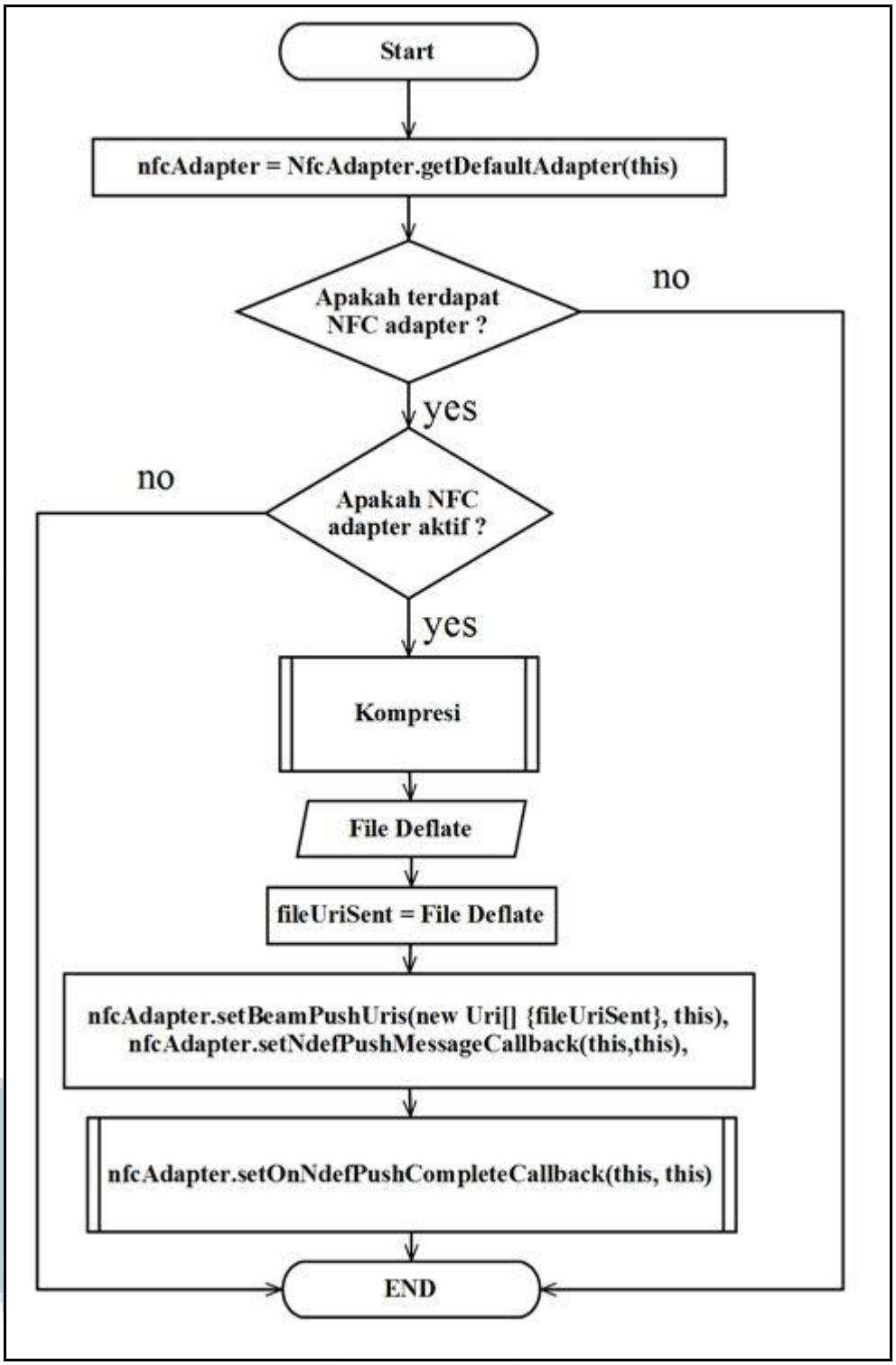
Gambar 3.5 merupakan desain *flowchart* subproses ShovelInToOut yang berguna memindahkan byte dari input stream ke output stream.



Gambar 3.5. Desain *flowchart* subproses ShovelInToOut.

6. Pengiriman menggunakan Android Beam.

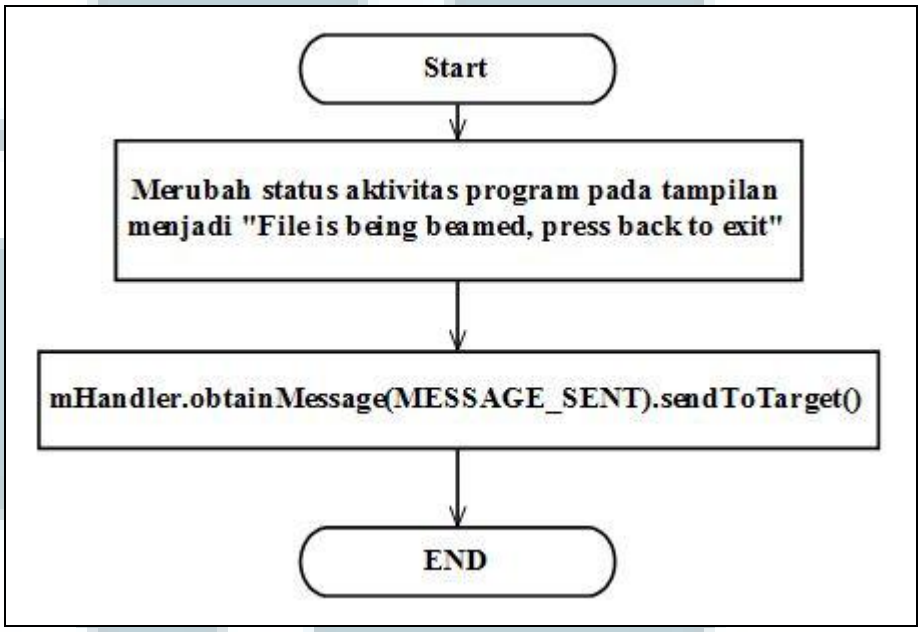
Setelah *file* hasil kompresi didapat maka dilakukanlah prosedur pengiriman menggunakan Android Beam. Pengiriman menggunakan NFC pada Android dibantu menggunakan fitur Android Beam. Dengan menggunakan Android Beam aplikasi pengiriman dan penerimaan *file* dapat melakukan pengiriman NFC *Data Exchange Format* (NDEF) guna melakukan koneksi pengiriman ke perangkat tujuan. Untuk lebih jelas maka dibuatlah *flowchart* pengiriman dengan menggunakan Android Beam dapat dilihat pada gambar 3.6.



Gambar 3.6. Desain *flowchart* pengiriman dengan Android Beam.

7. Desain *flowchart* subproses NdefPushComplete.

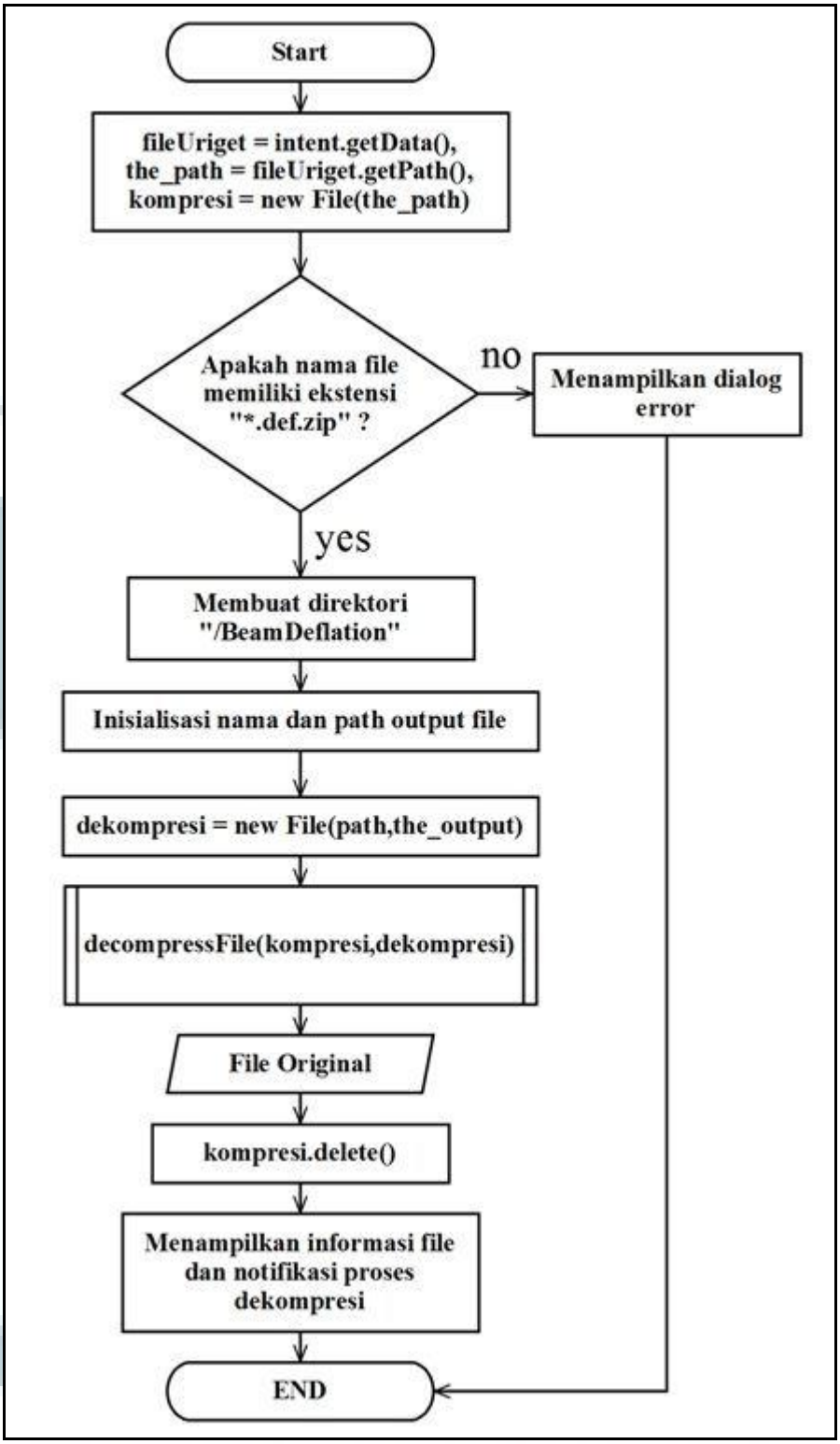
Gambar 3.7 merupakan desain *flowchart* NdefPushComplete untuk memberitahu *handler* pada NFC bahwa proses pengiriman pesan Ndef telah berhasil yang dilanjutkan proses menampilkan status program pada tampilan.



Gambar 3.7. Desain *flowchart* subproses NdefPushComplete.

8. Dekompresi *file* menggunakan Inflater

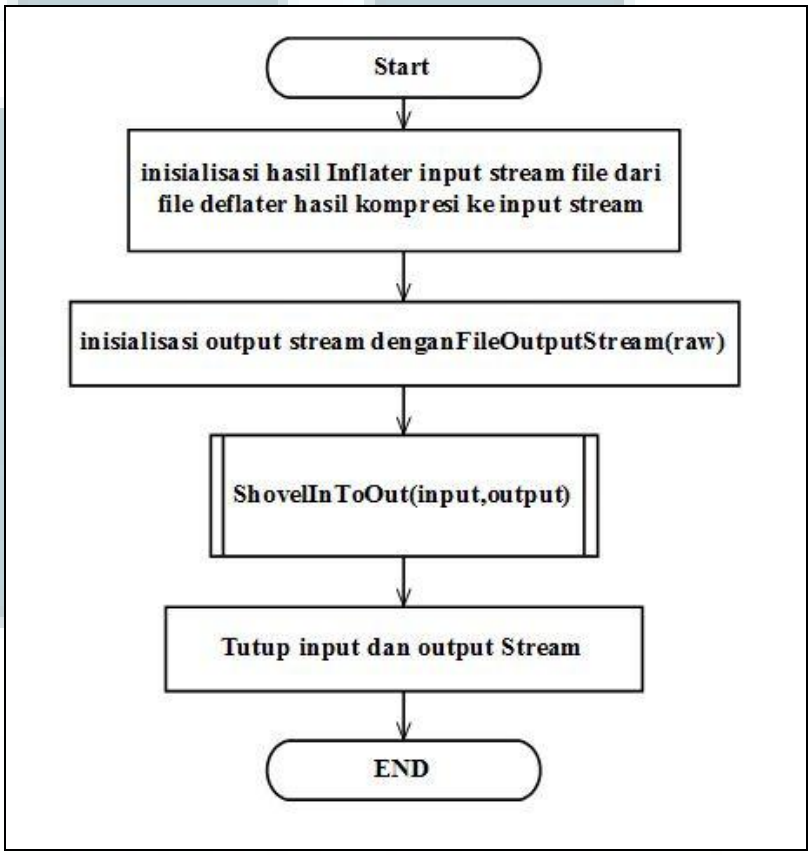
Sama seperti kompresi, dekompresi dibantu menggunakan inflater. *File* yang didapat di perangkat tujuan akan diproses untuk mendapatkan *path*-nya. Setelah itu dilakukan pengecekan apakah nama *file* yang didapat merupakan hasil kompresi di perangkat sumbernya. Jika ya maka dilanjutkan proses dekompresi *file*. Proses ini mirip dengan proses kompresi dimana terjadi pengisian *byte* hasil dekompresi ke *file* baru. *Flowchart* pada gambar 3.8 menjelaskan proses dekompresi lebih jelas.



Gambar 3.8. *Flowchart* dekompresi.

9. Desain *flowchart* subproses decompressFile

Gambar 3.9 menampilkan *flowchart* subproses decompressFile. Fungsi subproses decompressFile adalah melakukan dekompresi file deflate hasil kompresi menjadi file asli atau original-nya.



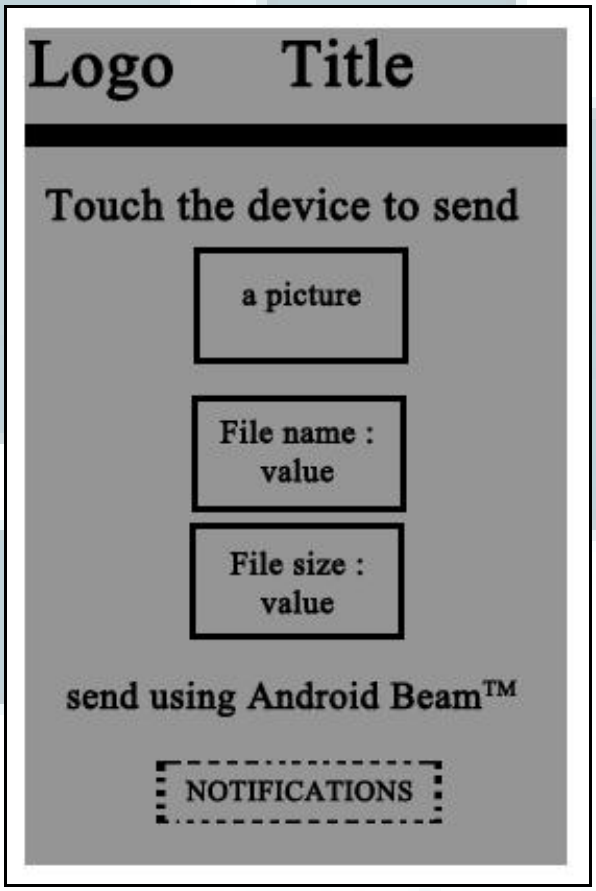
Gambar 3.9. Desain *flowchart* subproses decompressFile.

10. *File Output*

File hasil dekompresi dan merupakan merupakan *file* yang sudah selesai terproses. *File* ini merupakan *file* yang sama dengan *file input*.

3.3 Sketsa Antarmuka

Tampilan antarmuka aplikasi haruslah memiliki elemen – elemen informasi yang dibutuhkan untuk pengguna. Tampilan antarmuka aplikasi pengiriman dan penerimaan *file* sebagai berikut.

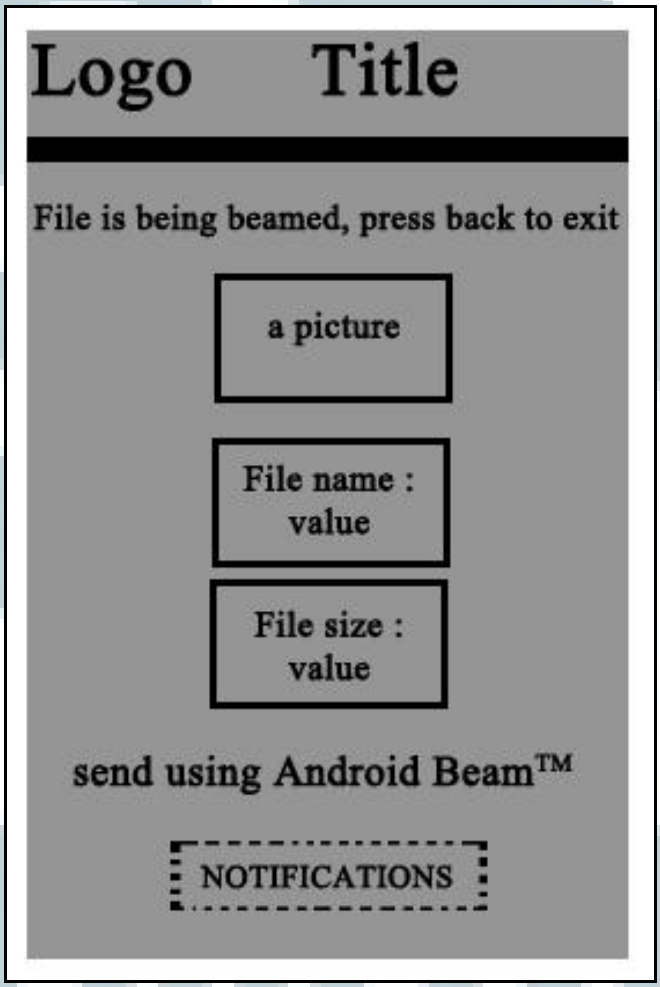


Gambar 3.10. Desain rancangan tampilan antarmuka aplikasi pengiriman dan penerimaan *file*.

Dari Gambar 3.10 dapat dilihat tampilan aplikasi pengiriman dan penerimaan *file* sangat sederhana. Tampilan antarmuka lebih berisi label – label teks yang menampilkan informasi tentang *file* yang digunakan aplikasi dan teks bantuan dalam melakukan prosedur pengiriman menggunakan Android Beam.

Karena itu hampir tidak ada tombol pada tampilan antarmuka aplikasi pengiriman dan penerimaan *file*.

Ketika proses kompresi atau dekompresi telah selesai dijalankan, sistem akan menampilkan notifikasi pada tampilan aplikasi dalam jenjang waktu tertentu kemudian menghilang.



Gambar 3.11. Desain tampilan aplikasi pengiriman dan penerimaan *file* setelah pengiriman.

Pada gambar 3.11 dapat dilihat status aplikasi sedang melakukan pengiriman file sehingga label status berubah. Notifikasi akan keluar ketika file sudah selesai dikirim dan akan menghilang pada jenjang waktu tertentu.