

## Pengembangan Aplikasi *Encode* dan *Decode Tree* Menggunakan *Blob Code*

<sup>1</sup>Astria Hijriani, <sup>2</sup>Wamiliana dan <sup>3</sup>Ady Candra

<sup>1</sup>Jurusan Ilmu Komputer Universitas Lampung

<sup>2</sup>Jurusan Matematika Universitas Lampung

<sup>3</sup>Jurusan Ilmu Komputer Universitas Lampung

### Abstrak

Tree Structure is used widely to represent problems in daily life, especially in computer science and technology. One method that is used to represent a tree is Blob Code. In this research, we developed an application to Encode and Decode Tree using Blob Code. This application aims to simplify the process of changing tree into a blob code (encode) and the process of changing Blob code into the form of tree (decode).

**Keywords:** *Encode, Decode, Tree, representation Tree, Blob Code.*

### 1. Pendahuluan

Graf adalah salah satu bagian dari bidang ilmu matematika yang dapat digunakan untuk mencari solusi dari suatu permasalahan. Dalam kehidupan sehari-hari graf digunakan sebagai visualisasi dengan menggambarkan bermacam-macam struktur hingga lebih mudah dimengerti. *Tree* merupakan salah satu struktur dari graf yang sering digunakan untuk menyelesaikan masalah khususnya di bidang ilmu komputer dan teknologi. *Tree* adalah suatu graf yang mempunyai  $n$  titik,  $n-1$  garis dan tidak mempunyai lingkaran (*cycle free*) serta merupakan graf terhubung[1]. Untuk dapat mengaplikasikan konsep *tree* ke dalam pemrograman komputer diperlukan proses *encode* dan *decode*. *Encode* adalah suatu proses untuk melakukan pengkodean ulang terhadap suatu data, sehingga data tersebut apabila dilihat tidak akan sama dengan aslinya, dengan tanpa merusak informasi di dalamnya. *Decode* adalah suatu metode pembacaan suatu data yang telah dikirimkan dalam bentuk sandi, kemudian diterjemahkan kembali ke dalam bentuk aslinya[2].

Pada tahun 2010 Zayadi telah melakukan penelitian tentang pengkodean *encode* dan *decode tree* menggunakan *Blob code* dan *Prufer code*. Pada penelitian tersebut Zayadi belum mengembangkan aplikasi pengkodean *encode* dan *decode*. Pada penelitian ini dilakukan penelitian tentang pengkodean *encode* dan *decode tree* menggunakan *Blob code*.

### 2. Metode

Metode yang digunakan pada penelitian ini adalah *Extreme Programming*. Terdapat 4 tahap pada pengembangan perangkat lunak yang menggunakan XP terdiri dari planning seperti memahami kriteria pengguna dan perencanaan pengembangan, *designing* seperti perancangan *prototype* dan tampilan, *coding* termasuk pengintegrasian, dan yang terakhir adalah testing[3].

#### a. Analysis

Tahap ini merupakan tahap dasar dalam membuat aplikasi perangkat lunak, dimana peneliti menganalisis kebutuhan-kebutuhan apa saja yang diperlukan untuk membangun aplikasi. Pada penelitian ini penulis mencari kebutuhan yang diperlukan seperti perangkat keras (*hardware*) dan

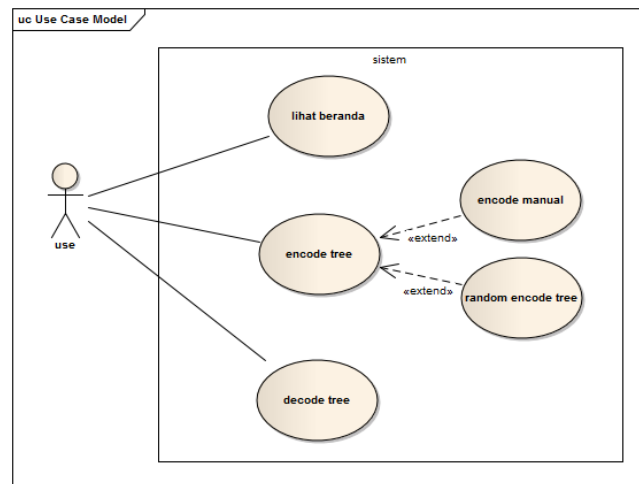
perangkat lunak (*software*) yang mendukung tahapan pengerjaan aplikasi. Untuk mengetahui kebutuhan-kebutuhan yang akan diterapkan di dalam aplikasi, penulis mengumpulkan data berdasarkan metode pengumpulan data wawancara dan studi pustaka.

### ***b. Design***

Setelah dilakukan analisis terhadap kebutuhan-kebutuhan yang diperlukan dalam pembuatan aplikasi ini, tahap selanjutnya yaitu tahap desain. Dalam tahap ini semua kebutuhan yang telah dianalisis dirancang menjadi *usecase diagram*, *activity diagram*, dan *sequence diagram*.

#### ***Usecase***

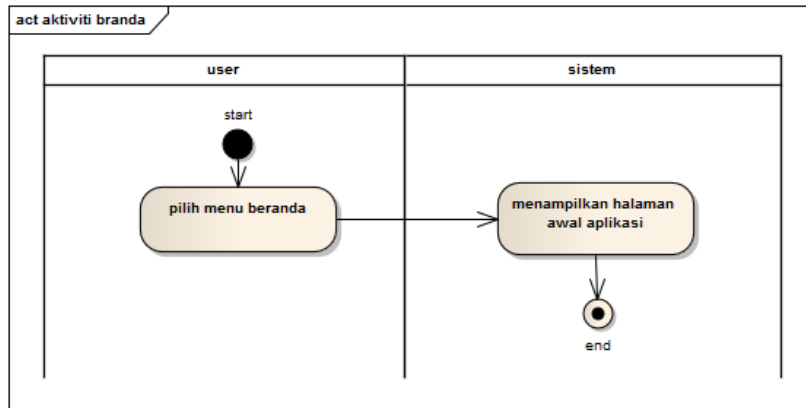
Sistem ini digunakan oleh seorang actor yaitu user. *User* dapat menginputkan *decode tree* dan juga dapat menginputkan *encode tree* sesuai *usecase* pada Gambar 1



Gambar 1 : *Usecase*

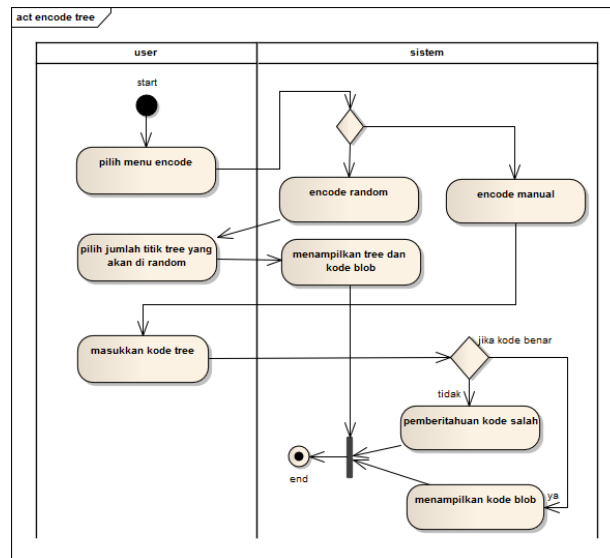
#### ***Activity Diagram***

*Activity Diagram* dibuat berdasarkan dengan *usecase* diagram. Aplikasi ini memiliki 3 buah aktivitas yaitu *lihat beranda*, *encode tree* dan *decode tree*. Aktivitas *lihat beranda* merupakan proses untuk dapat melihat halaman awal aplikasi. Pada halaman beranda terdapat petunjuk penggunaan aplikasi, proses ini dapat dilihat pada Gambar 2.



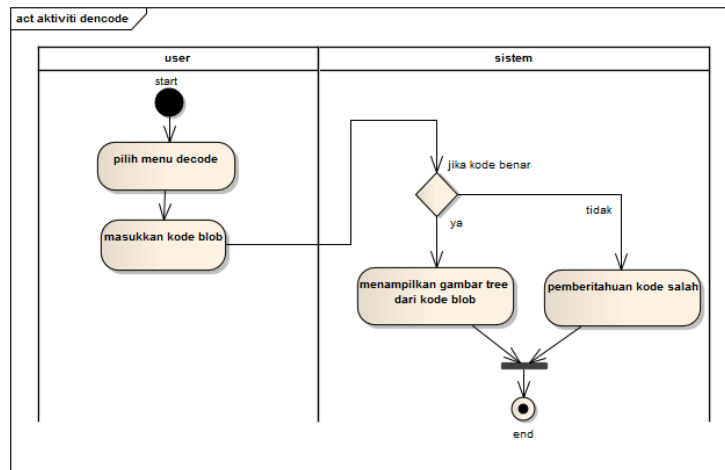
Gambar 2 : Activity diagram lihat beranda

Aktivitas *encode tree* terdapat 2 proses yaitu *encode random* dan *encode manual*. *Encode random* merupakan proses memilih jumlah titik *tree* yang akan diacak kemudian diubah menjadi sebuah *tree* dan mendapatkan kode Blob dari *tree* yang telah dirandom tersebut, sedangkan *encode manual* merupakan proses menginput kode *tree* yang kemudian akan diubah menjadi bentuk *tree* dan didapat kode Blobnya, proses ini dapat dilihat pada Gambar 3.



Gambar 3 : Activity encode tree

Aktivitas *decode tree* merupakan proses memasukkan kode Blob menjadi bentuk *tree*, proses ini dapat dilihat pada Gambar 4.

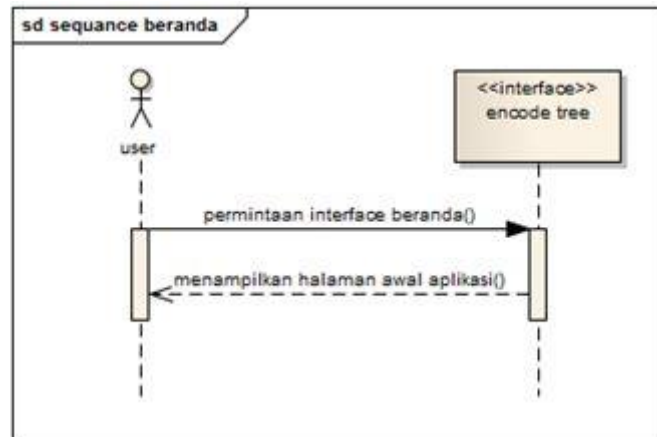


Gambar 4 : Activity diagram encode tree

### Sequence Diagram

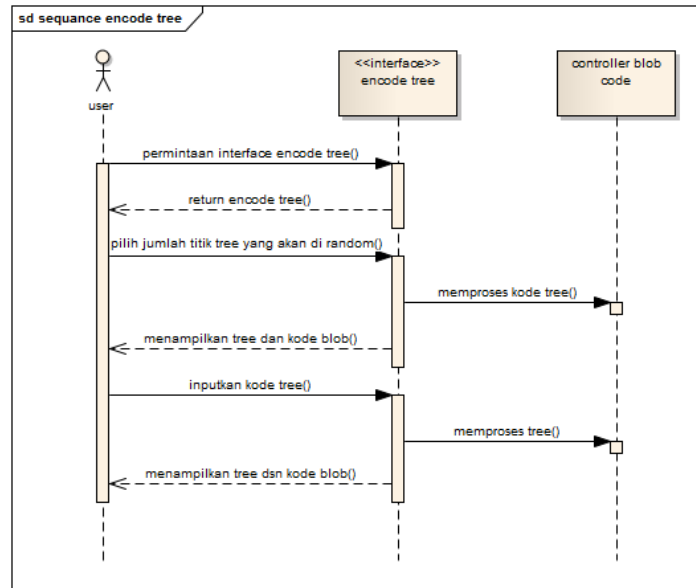
Langkah selanjutnya dari tahap desain adalah pembuatan *sequence diagram*. Jumlah *sequence diagram* yang dibuat harus sesuai dengan *activity diagram*.

Pada Gambar 5 dijelaskan tentang *sequence diagram* melihat beranda, dimana user meminta tampilan awal aplikasi.



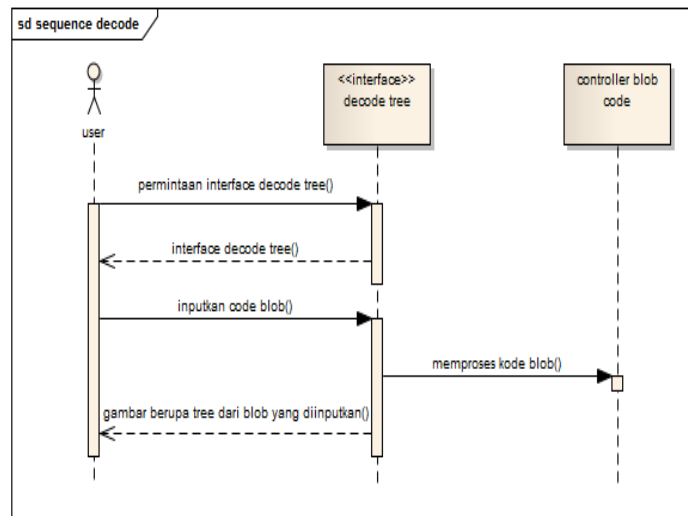
Gambar 5 : Sequence diagram lihat beranda

Pada Gambar 6 dijelaskan tentang *sequence diagram encode tree*, dimana terdapat 2 pilihan proses yaitu *encode random* dan *encode manual*. Proses *encode manual* user meminta *interface encode tree* kemudian menginputkan *tree* sehingga dihasilkan kode Blob, sedangkan pada proses *encode random* user meminta *interface encode tree* kemudian memilih jumlah titik *tree* yang akan dirandom menjadi sebuah *tree* sehingga dihasilkan kode Blob.



Gambar 6 : Sequence diagram encode tree

Pada Gambar 7 dijelaskan tentang sequence diagram *decode tree*, dimana *user* meminta *interface decode tree* kemudian menginputkan kode Blob sehingga dihasilkan gambar berupa *tree*.



Gambar 7 : Sequence diagram decode tree

### 3. Implementasi dan Hasil

Berikut ini merupakan tahapan implementasi dari Pengembangan Aplikasi *Encode Dan Decode Tree* Menggunakan *Blob Code*.

### a. Halaman Utama Sistem

Halaman utama yang ditampilkan pada aplikasi ini dapat dilihat pada Gambar 8. Pada halaman utama Aplikasi *Encode Dan Decode Tree* Menggunakan *Blob Code* terdapat petunjuk penggunaan aplikasi yang dapat membantu seorang user untuk menggunakan aplikasi ini.



Gambar 8 : Halaman Utama Aplikasi

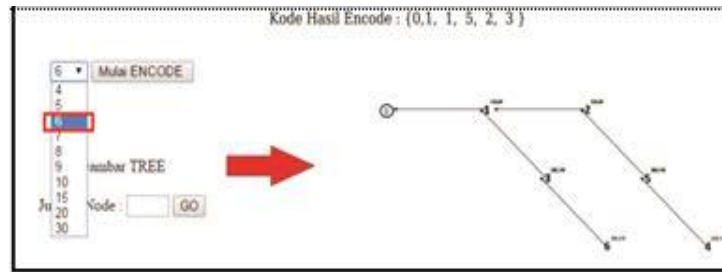
### b. Halaman *Encode*

Halaman *Encode* yang ditampilkan pada aplikasi ini dapat dilihat pada Gambar 9.



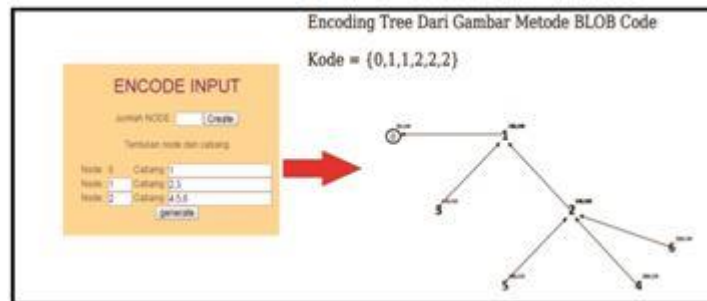
Gambar 9 : Halaman *Encode*

Pada halaman *encode* terdapat 2 pilihan yaitu *encode* random dan *encode* manual. Pada *encode* random langkah pertama yang harus dilakukan adalah memilih jumlah titik yang akan dilakukan random, kemudian klik tombol mulai *encode* lalu akan muncul *tree* yang jumlahnya sama dengan titik yang telah dipilih dan hasil kode Blob-nya, dapat dilihat pada Gambar 10.



Gambar 10 : Encode random

Encode manual yang pertama dilakukan adalah memasukkan jumlah titik utama yaitu titik yang terdapat panah yang mengarah pada titik tersebut lalu klik tombol *create*. Kemudian akan muncul *form input* baru yaitu *form node* memasukkan titik *tree* utama sedangkan pada *form* cabang masukkan cabang dari titik utama tersebut, dapat dilihat pada Gambar 11.



Gambar 11: Encode input

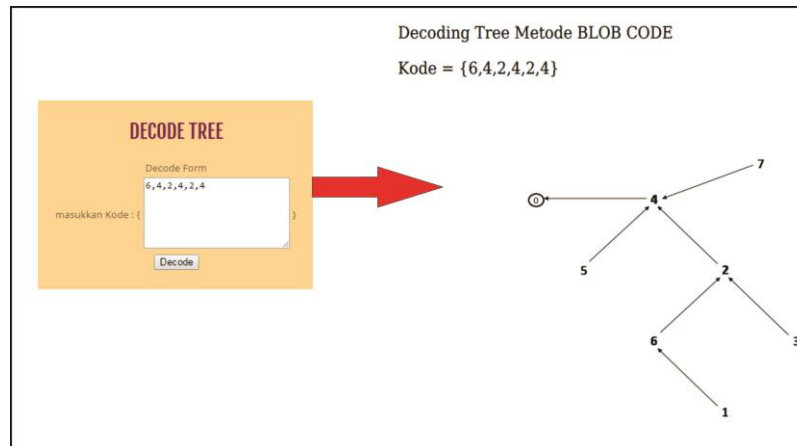
### c. Halaman Decode

Halaman *Decode* yang ditampilkan pada aplikasi ini dapat dilihat pada Gambar 12. Pada halaman *decode* terdapat *form input* yang digunakan untuk menginputkan kode Blob yang kemudian diubah menjadi bentuk *tree*.



Gambar 12 : Halaman decode

Pada *form input* masukkan kode Blob, setiap kode dipisahkan dengan tanda koma. Kemudian klik tombol *decode* lalu akan muncul *tree* dari kode Blob tersebut, dapat dilihat pada Gambar 13.



Gambar 13 : *Input decode*

#### d. *Testing*

Pada tahap ini dilakukan pengujian terhadap aplikasi yang telah dikembangkan. Pengujian ini bertujuan mencari kesalahan-kesalahan aplikasi yang dikembangkan. Sehingga dihasilkan suatu aplikasi yang memberikan kemudahan kepada *user* dalam menggunakan aplikasi tanpa menemukan suatu kesalahan. Berikut adalah hasil pengujian dalam bentuk tabel sesuai dengan yang ditunjukkan Tabel 3.1.



Tabel 3.1 Hasil Pengujian Dalam Tabel

No	Evaluasi	Ya	Tidak	Keterangan
1	Apakah ada pemberitahuan kepada pengguna ketika memasukkan kode yang salah	√		Aplikasi akan memberikan peringatan bila kode yang diinput salah
2	Apakah kode Blob yang telah dimasukkan dapat diubah menjadi bentuk tree	√		Aplikasi akan menampilkan bentuk tree dari kode Blob yang telah di masukkan
3	Apakah aplikasi dapat merandom tree dan mendapatkan kode Blob dari tree tersebut	√		Aplikasi dapat merandom tree dan mendapatkan kode Blob dari tree tersebut
4	Apakah aplikasi dapat merubah kode tree yang telah diinputkan menjadi bentuk tree dan mendapatkan kode Blob.	√		Aplikasi dapat merubah kode tree yang telah diinput menjadi bentuk tree dan mendapatkan kode Blob.
5	Apakah ada pemberitahuan bila input berupa huruf	√		Aplikasi akan memberikan peringatan bila kode yang telah diinput bukan berupa angka
6	Apakah ada pemberitahuan bila input lebih dari 50 angka	√		Aplikasi akan memberikan peringatan bila kode yang telah diinput lebih dari 50 angka
7	Apakah hasil dari <i>encode</i> dapat di <i>copy paste</i> pada <i>form input decode</i>	√		Aplikasi dapat <i>copy paste</i> hasil <i>encode</i> ke <i>form input decode</i>
8	Apakah titik tree tidak saling tumpang tindih		√	Titik tree masih ada yang saling tumpang tindih
9	Apakah menu petunjuk dapat ditampilkan dengan baik	√		Aplikasi dapat menampilkan cara penggunaan
10	Apakah belum memasukkan kode ke dalam <i>form input</i> akan bisa diakses	√		Aplikasi akan member peringatan <i>form</i> belum diisi

#### 4. Kesimpulan

Kesimpulan yang dapat diambil dari penelitian ini adalah sebagai berikut.

1. Aplikasi *encode* dan *decode tree* menggunakan *Blob code* telah berhasil dibangun sesuai dengan kebutuhan.
2. Fungsi *encode* telah berhasil mengubah sebuah *tree* menjadi kode *Blob*.
3. Fungsi *decode* telah berhasil mengubah kode *Blob* menjadi *tree*.
4. Karena keterbatasan pemrograman PHP, disarankan penelitian berikutnya dapat menggunakan OOP yang lain seperti java, C++, dan lain-lain.

#### 5. Referensi

- [1] Wibisono, Samuel. 2010. *Matematika Diskrit*. Graha Ilmu: Yogyakarta.
- [2] Zayadi, Wahyu Emir. 2010. Teknik Encode Dan Decode Tree Menggunakan Blob Code Dan Prufer Code. Skripsi. Universitas Lampung: Lampung.
- [3] Pressman, R., S. 2010. *Software Engineering A Practitioner's Approach 7<sup>th</sup> Edition*. McGraw-Hill: New York.