



Implementasi Algoritma *Iterative Dichotomiser 3* (ID3) Untuk Penentuan Jumlah Dana Bantuan Perbaikan Rumah Di Bappeda

Tomi Kapri¹, Muhammad Nasir², Eka Puji Agustini³

^{1,2,3} Fakultas Ilmu Komputer niversitas Bina Darma

tomikapri42@gmail.com¹, nasir@binadarma.ac.id², eka_puji@binadarma.ac.id³

Abstrak

Rumah merupakan tempat tinggal yang nyaman baik secara individu maupun keluarga. Namun tentu ada kriteria penilaian yang perlu dilihat dan dipertimbangkan sebagai bentuk penilaian sebagai yang layak huni. Bagi masyarakat miskin, untuk memenuhi kriteria layak huni sangatlah tidak mudah dan menjadi masalah yang perlu diperhatikan. Oleh karena itu, untuk memperkecil kemungkinan agar tidak terjadinya kesalahan dalam untuk penentuan jumlah dana bantuan perbaikan rumah dilakukan pendekatan dengan metode Algoritma *Iterative Dichotomiser 3* (ID3). Dalam penelitian ini akan digunakan Algoritma *Iterative Dichotomiser 3* (ID3) dalam menentukan data yang menerima dan tidak menerima bantuan dengan kriteria jumlah keluarga, dinding, lantai, tingkat kerusakan dan pekerjaan. Pengembangan sistem menggunakan metode *Rational Unified Process* (RUP). Perangkat ini dibuat dengan menggunakan Bahasa Pemrograman Java, *NetBeans* 8.2. Hasil dari penelitian ini berupa perangkat lunak yang mampu membantu menghasilkan keputusan secara otomatis.

Kata Kunci : Rumah layak huni, *Iterative Dichotomiser 3* (ID3), *Rational Unified Process* (RUP), Pemrograman Java, *NetBeans* 8.2

1. PENDAHULUAN

Salah satu masalah besar yang dihadapi oleh negara manapun di dunia ini adalah kemiskinan. Kemiskinan dipandang sebagai ketidakmampuan dari sisi ekonomi untuk memenuhi kebutuhan dasar makanan dan bukan makanan yang diukur dari sisi pengeluaran itu yang dipaparkan oleh Badan Pusat Statistik (BPS). Bagaimana mereka dapat memenuhi kebutuhan pokok lainnya seperti kesehatan, pendidikan, lingkungan bahkan tempat tinggal jika kebutuhan dasar makanan saja tidak dapat dipenuhi. Sebagaimana kebijakan negara Indonesia dengan programnya agar untuk menghapuskan kemiskinan sebagaimana sudah dimandatkan dalam UUD 1945 Pasal 28 H yang berbunyi "Setiap orang berhak hidup sejahtera lahir dan batin, bertempat tinggal, dan mendapatkan lingkungan



hidup yang baik dan sehat serta berhak memperoleh pelayanan kesehatan". Rumah merupakan tempat tinggal yang nyaman baik secara individu maupun keluarga. Namun tentu ada kriteria penilaian yang perlu dilihat dan dipertimbangkan sebagai bentuk penilaian sebagai yang layak huni. Bagi masyarakat miskin, untuk memenuhi kriteria layak huni sangatlah tidak mudah dan menjadi masalah yang perlu diperhatikan.

Berdasarkan latar belakang di atas diperoleh suatu rumusan masalah yaitu "Bagaimana Mengimplementasikan Algoritma ID3 Untuk Penentuan Jumlah Dana Bantuan untuk Perbaikan Rumah?". Untuk lebih terarah dan tidak menyimpang dari pokok masalah yang ada. Maka penulis membatasi pada permasalahan berikut ini: Penelitian ini di BAPPEDA Banyuasin Data penerima bantuan berdasarkan tahun 2015 sampai dengan 2017 dan Perangkat lunak yang dibangun mampu memberikan hasil keputusan berupa nominal bantuan yang diterima oleh masyarakat yang layak menerima bantuan.

2. METODOLOGI PENELITIAN

2.1 Metode Pengumpulan Data

Pengumpulan data merupakan tahap penting dalam penulisan karya ilmiah. Pengumpulan data menurut Sugiyono (2012:137) dapat dilakukan dengan berbagai *setting*, berbagai sumber, dan berbagai cara dalam upaya mengumpulkan data. Adapun metode penelitian yang digunakan dalam penelitian ini ialah:

- a. Metode Wawancara (*interview*), Dalam penelitian ini mengadakan wawancara atau tanya jawab langsung kepada staff kantor BAPPEDA Banyuasin tentang program BSPS mengenai regulasi dan bagaimana data para calon penerima bantuan tersimpan.
- b. Metode Studi Pustaka, Dengan melakukan pencarian dan pengumpulan buku-buku maupun informasi dari jurnal sebagai bahan referensi serta mempelajari dan memahami proses data *mining* dan Metode ID3 untuk melakukan prediksi nantinya.

2.2 Metode Pengembangan Perangkat lunak

Rational Unified Process (RUP) merupakan suatu metode rekayasa perangkat lunak yang dikembangkan dengan mengumpulkan berbagai praktik terbaik yang terdapat dalam industri pengembangan perangkat lunak. *Rational Unified Process (RUP)* adalah salah satu proses rekayasa perangkat lunak yang menyediakan disiplin untuk menetapkan tugas dan tanggung jawab dalam pengembangan organisasi. Tujuan *Rational Unified Process (RUP)* adalah untuk menjamin sebuah produk perangkat lunak yang berkualitas tinggi dalam memenuhi kebutuhan para pengguna akhir dengan jadwal dan anggaran yang dapat diprediksi. *Rational Unified Process (RUP)* bukanlah suatu proses tunggal dengan aturan yang konkrit, melainkan suatu kerangka proses yang dapat diadaptasi dan dimaksudkan untuk disesuaikan oleh organisasi pengembang dan

tim proyek perangkat lunak yang akan memilih elemen proses sesuai dengan kebutuhan mereka. *Rational Unified Process (RUP)* adalah salah satu dari sekian banyak proses yang terdapat di dalam Rational Process Library, yang memberikan simulasi terbaik untuk pengembangan atau kebutuhan proyek (Kruchten, 2001).

3. HASIL DAN PEMBAHASAN

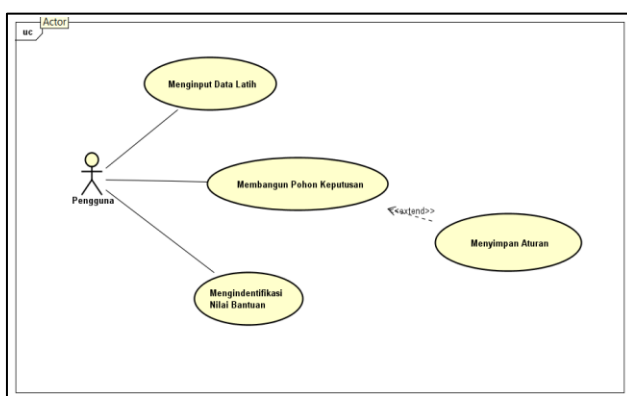
Hasil yang diperoleh dalam penelitian ini adalah sebuah Sistem yang berbasis Desktop. pada tahap pembangunan aplikasi ini peneliti membuat 1 aktor yang terlibat yakni admin, pengguna disini merupakan pemerintah BAPPEDA Banyuasin yang ingin menggunakan aplikasi ini.

3.1 Desain

Desain perangkat lunak dilakukan untuk mengetahui rancangan antar muka seperti apa yang ingin dibuat sebelum pada proses *pengodingan*. Proses perancangan menggunakan *Unified Modeling Language (UML)* merupakan “suatu standar bahasa yang banyak digunakan dalam membuat analisis dan desain serta menggambarkan arsitektur dalam pemograman berorientasi objek” (Rosa A.S dan M. Shalahuddin, 2016:137) yang terdiri dari 13 macam diagram namun pada tahap ini penulis menggunakan tiga diagram yaitu *usecase diagram*, *Activity diagram* dan *class diagram*.

3.1.1 Use case

Use Case merupakan gambaran statis akan suatu cara bagaimana seorang aktor dalam sistem dapat dipergunakan menurut perannya. *Use Case Diagram* menunjukkan bagaimana hubungan *use case* sistem antara satu dengan yang lain dan bagaimana pengguna dapat menjalankan tiap-tiap *use case*.



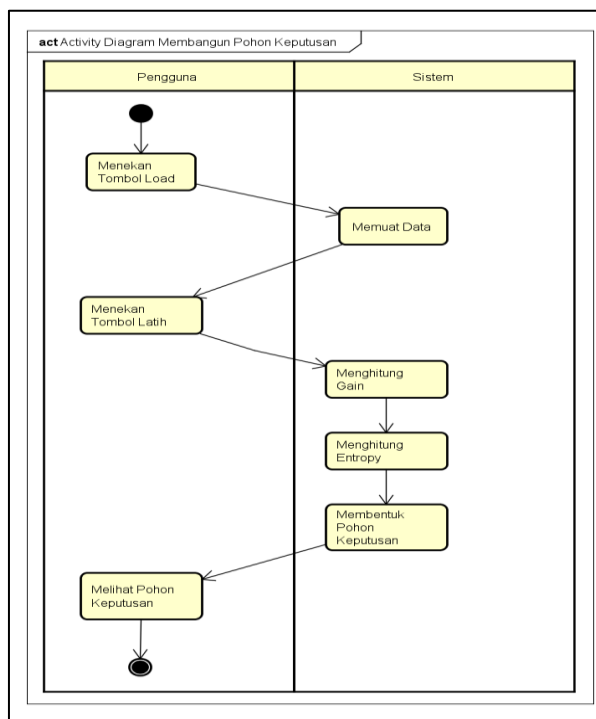
Gambar 1 Use Case

3.1.2 Activity Diagram

Activity Diagram digunakan untuk mendefinisikan rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan, urutan atau pengelompokan tampilan dari *interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan, rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya dan rancangan menu yang ditampilkan pada perangkat lunak. Berikut *activity diagram* yang terdapat pada perangkat lunak:

a. Activity Diagram Membangun Pohon Keputusan

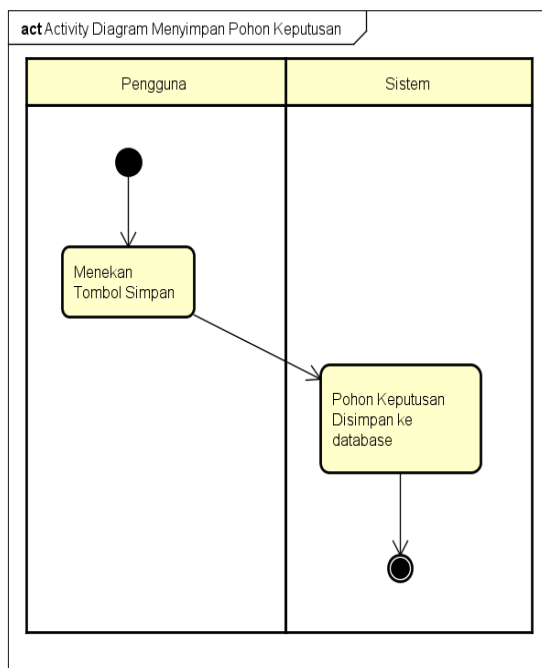
Diagram di bawah ini menggambarkan langkah-langkah bagaimana pohon keputusan dibangun, dimulai dari memilih tombol muat data dan kemudian dilanjutkan dengan memilih bangkit untuk membangun pohon keputusan.



Gambar 2 Activity Diagram Login Admin

b. Activity Diagram Menyimpan Aturan

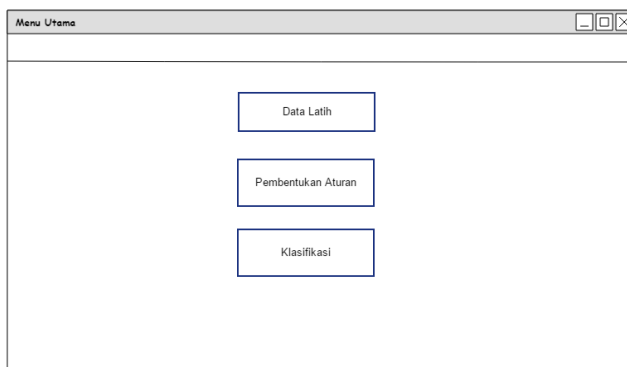
Diagram activity pengguna ini diperuntukan oleh pengguna dalam melihat informasi yang terdapat pada aplikasi informasi mengenai pembuat aplikasi, mengenai informasi pelayanan public dan informasi peta.



Gambar 3 Activity Diagram Menyimpan Aturan

3.2 Desain Interface

ada proses ini *design interface* dibuat atau dirancang untuk mengetahui bagaimana interaksi antara *user* dengan *system*. Penulis menggunakan *justinmind* sebagai tampilan *interface* yang memungkinkan masyarakat lebih terbayang dengan apa yang sebenarnya diinginkan sehingga dalam tahap evaluasi akan menjadi lebih mudah. Seperti gambar berikut ini :



Gambar 4 Design Beranda

3.3 Pembuatan kode program

Pada tahap ini, hasil dari perancangan pada tahap sebelumnya diterjemahkan ke dalam bahasa pemrograman atau disebut juga langkah pembuatan kode. Pada Pengembangan Sistem Penentuan Jumlah Dana Bantuan Perbaikan Rumah Di ini BAPPEDA dibuat menggunakan bahasa pemrograman *java*.

3.4 Pengujian

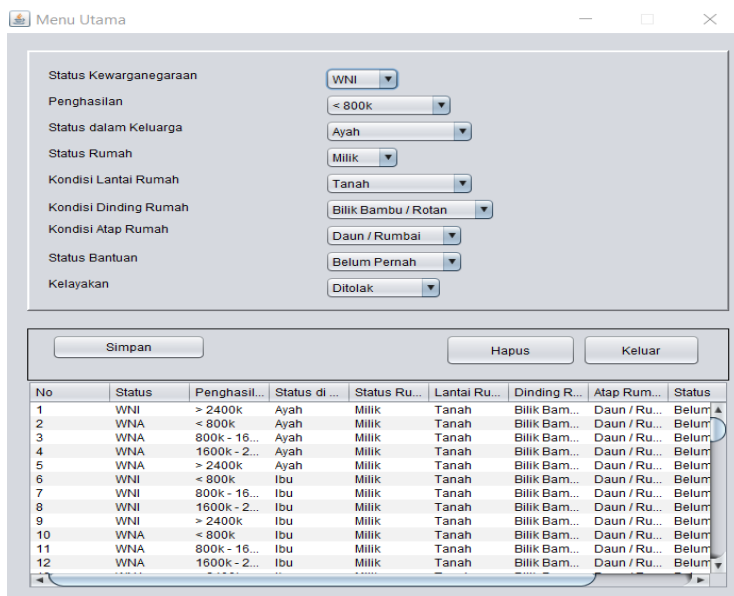
Setelah proses pembuatan perangkat lunak telah selesai, proses selanjutnya di uji menggunakan metode pengujian *black box testing* untuk mengetahui *system* yang telah dibuat sesuai dengan hasil rancangan sebelumnya dan meminimalisi kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

3.5 Construction

Tahap *Construction* di dalam metode pengembangan *Rational Unified Process (RUP)* merupakan tahap di mana dilakukan realisasi terhadap desain yang telah diciptakan pada tahapan sebelumnya. Mulai dari menentukan lingkungan pengembangan, mempelajari cara kerja Algoritma ID3, menentukan aktor dan merancang *usecase*, memodelkan *activity diagram* dan *class diagram* serta merancang antar muka pengguna.

a. Training Data Latih dengan ID3

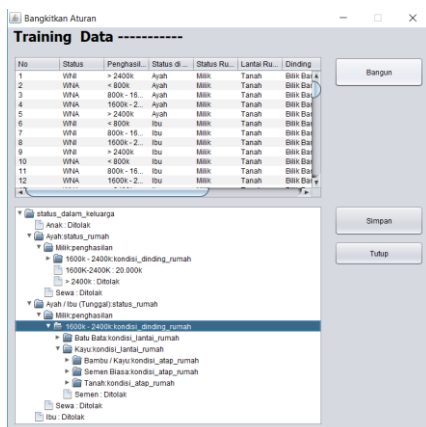
Ketika data latih sudah siap, data latih tersebut dipergunakan untuk membangun aturan. Aturan diperoleh melalui perhitungan Algoritma ID3, dengan mencari nilai gain terbaik untuk menentukan cabang baru dan nilai entropi 0 sebagai hasil keputusan. Berikut gambar tampilan pengguna ketika persiapan membangun aturan.



Gambar 5 Data Persiapan Pelatihan

b. Menu Membangun Pohon Keputusan

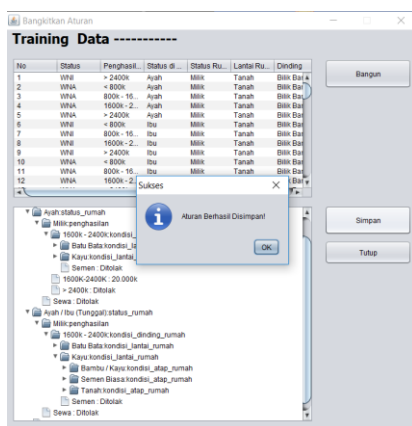
Dengan menekan tombol *build*, pohon aturan dibangun dan setiap simpul berakhir dengan sebuah keputusan. Maka dari itu ini disebut dengan pohon keputusan.



Gambar 6 Tampilan Pembangunan Pohon Keputusan

c. Menu Pohon Keputusan Yang Telah Berhasil Disimpan

Pohon keputusan yang berhasil disimpan tersebut berubah menjadi 2 tabel antara lain anteseden dan konsekuen. Konsekuen menyimpan data keputusan sedangkan anteseden menyimpan data sebab atau faktor.

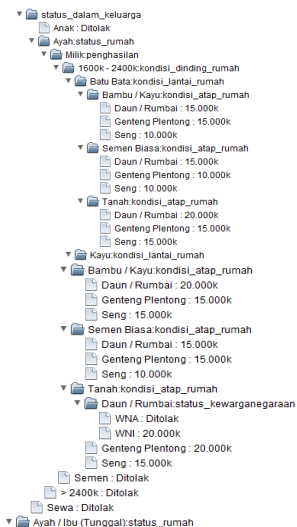


Gambar 7 Pohon Keputusan Berhasil disimpan

d. Menu Pohon Keputusan

Dengan melihat pohon keputusan tersebut, dapat disimpulkan bahwa terdapat banyak sekali simpul/*node* yang tiap ujungnya berakhir di daun yang datanya

bernilai Ditolak, 10.000k, 15.000k dan 20.000k. Semua daun dalam pohon keputusan merupakan sebuah konsekuen atau keputusan dari kumpulan antesedent.



Gambar 8 Pohon Keputusan

d. Menu Klasifikasi menggunakan Pohon Keputusan

Hasil dari klasifikasi di atas adalah ditolak, mengapa ditolak padahal penghasilan calon penerima bantuan tersebut di bawah 800 ribu dan seorang WNI sejati? Karena berdasarkan aturan 1 Jika status_dalam_keluarga='Anak' maka Ditolak. Classifier menghasilkan kesimpulan "ditolak" berdasarkan aturan 1 dari pohon keputusan di mana hanya perlu satu anteseden status_dalam_keluarga = "Anak" maka otomatis program akan mengklasifikasi calon penerima bantuan sebagai yang "ditolak".

Gambar 9 Menu Kasus Ditolak

3. KESIMPULAN

Akhirnya sudah sampai pada akhir dari laporan. Setelah mengikuti langkah-langkah pengembangan perangkat lunak dengan *Rational Unified Process (RUP)* dan dengan melihat hasil dari implementasi algoritma ID3 mengantarkan pada suatu kesimpulan sebagai berikut:

1. Semua ragam jumlah bantuan yang akan diterima semuanya berhasil terbentuk dalam pohon keputusan.
2. Algoritma ID3 berhasil membangun pohon keputusan dari data latihan dengan melihat nilai *gain* dan entropi.
3. Hasil klasifikasi atau penentuan nilai bantuan sangat tergantung dengan hasil *rule* atau aturan yang tersimpan

DAFTAR PUSTAKA

- [1] Bardin, D. E. (2017). Perancangan dan Implementasi Sistem Pendukung Keputusan Penerimaan Asisten Dosen FTI UKSW dengan Metode Iterative Dichotomiser 3 (ID3). *Satya Wacana*, 1-21.
- [2] HIMAWAN, D. (2014). Aplikasi Data Mining Menggunakan Algoritma ID3 Untuk Mengklasifikasi Kelulusan Mahasiswa Pada Universitas Dian Nuswantoro Semarang. *FASILKOM UDINUS*, 5-11.
- [3] Kristanto, O. (2014). PENERAPAN ALGORITMA KLASIFIKASI DATA MINING ID3 UNTUK MENENTUKAN PENJURUSAN SISWA SMAN 6 SEMARANG. *FASILKOMUDINUS*, 5-11.
- [4] Kurniawan, D. (2014). Membangun Rule Dengan Algoritma Iterative Dichotomiser 3 (Id3) Untuk Penjurusan Siswa SMA. *Jurnal Komputasi*, 42-47.
- [5] Mutmainah, P. A. (2015). DECISION TREE MENGGUNAKAN ALGORITMA ID3 UNTUK MELAKUKAN DETEKSI PENYAKIT KANKER PAYUDARA. *Nakula*, 5-11.
- [6] Rumagit, C. M. (2016). Penerapan Metode ID3 terhadap Perancangan Sistem Penentuan Penerima Bantuan Sosial Pemugaran RLTH Kota Salatiga. *TEKNOSI*, 101-116.
- [7] SETIAWAN, A. (2016). Implementasi Metode ID3 untuk memprediksi kegiatan mahasiswa sepulang kuliah dan kululusan mahasiswa. *Academia*, 1-91.

- [8] Tyasti, A. E. (2015). ALGORITMA ITERATIVE DICHOTOMISER 3 (ID3) UNTUK MENGIDENTIFIKASI DATA REKAM MEDIS. *JURNAL GAUSSIAN*, 237 - 246.
- [9] Wahyudin. (2009). Metode Iterative Dichotomizer 3 (ID3) Untuk Penerimaan Mahasiswa Baru. *JURNAL PENDIDIKAN TEKNOLOGI INFORMASI DAN KOMUNIKASI (PTIK)* , 1-15.
- [10] Wibowo, B. A. (2011). Perancangan Dan Implementasi Sistem Pendukung Keputusan Untuk Jalan Menggunakan Metode ID3 (Studi Kasus BAPPEDA Kota Salatiga). *Satya Wacana*, 1-17.