

Sistem Pendukung Keputusan Pemberian *Reward* Kepada Anggota Terbaik Dengan Menggunakan Metode *Analytical Hierarchy Process* (Studi Kasus: Koperasi Simpan Pinjam Utama Karya Jepara)

Bagus Setiaji¹ dan Djalal Er Riyanto²

^{1,2}Jurusan Ilmu Komputer/ Informatika Universitas Diponegoro, Semarang

¹bagus.setiaji29@gmail.com, ²erriyanto@undip.ac.id

Abstract

Koperasi Simpan Pinjam Utama Karya Jepara is a business corporation in the intermediate level for savings and loans to its members. To attract and retain members, the cooperative held the reward, the cooperative still uses a simple method, namely by drawing without the use of assessment criteria. Thus, there is a potential of election reward receiver to the member who gives a minimum contribution for the cooperative. In writing this final project will be developed a decision support system application of reward to the best member by using Analytical Hierarchy Process (AHP) with prototype process model. This application is developed by AHP method because this method has the ability to determine the candidate receiving the reward by using the assessment criteria. This application is developed with prototype process model because the process model acts as a mechanism to identify some specifications for software requirements. With the application of receiving the reward that is capable of processing four assessment criteria, namely the average balance of the year, the largest deposit in one year, the list of the date, and the intensity of saving transaction within one year will be the data for candidate of receiving the reward. The Cooperative obtains the output from an application like a supporting data as reference to determinate a candidate of reward receiver. So the cooperative does not give the reward to the member who gives a minimum contribution to the cooperative in another time.

Keyword : *Reward, Cooperative, Contribution, Application, Criteria.*

Abstrak

Koperasi Simpan Pinjam Utama Karya Jepara merupakan badan usaha tingkat menengah yang bergerak di bidang simpan dan pinjam untuk para anggotanya. Untuk menarik dan mempertahankan anggota, koperasi tersebut mengadakan kegiatan pemberian *reward* sebagai program tahunan. Dalam menentukan calon penerima *reward*, koperasi tersebut masih menggunakan cara sederhana yaitu dengan cara pengundian tanpa menggunakan kriteria penilaian. Dengan demikian berpotensi terpilihnya penerima *reward* untuk anggota yang memberikan kontribusi minimum terhadap koperasi. Dalam pembuatan tugas akhir ini akan dikembangkan suatu aplikasi sistem pendukung keputusan pemberian *reward* kepada anggota terbaik menggunakan metode *Analytical Hierarchy Process* (AHP) dengan model proses prototipe. Aplikasi dikembangkan dengan metode AHP karena metode tersebut mempunyai kemampuan untuk menentukan calon penerima *reward* dengan menggunakan kriteria penilaian. Aplikasi dikembangkan dengan model proses prototipe karena model proses tersebut bertindak sebagai mekanisme untuk mengidentifikasi spesifikasi-spesifikasi kebutuhan perangkat lunak. Dengan adanya aplikasi pemberian *reward* yang mampu memproses empat kriteria penilaian yaitu saldo rata-rata dalam satu tahun, setoran terbesar dalam satu tahun, tanggal terdaftar, dan intensitas transaksi menabung dalam satu tahun menjadi data calon penerima *reward*. Koperasi memperoleh *output* dari aplikasi berupa data pendukung sebagai bahan referensi untuk menentukan data calon penerima *reward*. Sehingga koperasi tidak lagi memberikan *reward* kepada anggota yang memberikan kontribusi minimum terhadap koperasi.

Kata kunci : *Reward, Koperasi, Kontribusi, Aplikasi, Kriteria*

1. Pendahuluan

Koperasi Simpan Pinjam Utama Karya Jepara merupakan badan usaha tingkat menengah yang

memiliki beberapa program kerja untuk para anggotanya. Salah satu program kerja tersebut adalah pemberian *reward* kepada calon penerima *reward*

yang diberikan dengan maksud untuk menarik dan mempertahankan anggota koperasi. Pada proses pemberian *reward*, pihak Koperasi Simpan Pinjam Utama Karya Jepara masih mengalami permasalahan. Permasalahan tersebut adalah koperasi belum memiliki metode yang sesuai untuk menentukan data calon penerima *reward*. Sehingga berpotensi *reward* akan diterima oleh anggota yang memiliki kontribusi minimum terhadap koperasi.

Untuk memecahkan masalah tersebut, diperlukan sebuah sistem pendukung keputusan pemberian *reward* menggunakan metode AHP yang dapat digunakan oleh manajer koperasi dalam menentukan data calon penerima *reward*. Metode AHP dipilih karena memiliki karakteristik berupa konsistensi logis yang dapat membantu manajer dalam menentukan data calon penerima *reward*.

AHP adalah sebuah metode dalam SPK yang komprehensif. AHP digunakan untuk menyelesaikan masalah multi-objektif dan multi-kriteria yang berdasar pada perbandingan preferensi dari setiap elemen dalam hirarki [7]. Masalah multi-objektif digunakan untuk merancang alternatif terbaik, sedangkan masalah multi-kriteria digunakan untuk menetapkan alternatif terbaik dari sejumlah alternatif berdasarkan beberapa kriteria tertentu [2].

Model proses prototipe adalah sebuah model proses pengembangan perangkat lunak yang terdiri dari 5 tahapan yaitu *communication*, *quick plan*, *modelling* and *quick design*, *construction of prototype*, *deployment delivery and feedback* [4]. Dengan menggunakan model proses prototipe, pengguna aplikasi dapat memberikan gambaran secara umum tentang aplikasi. Kemudian pengembang dapat membuat prototipe untuk membantu pengguna mendapatkan gambaran lebih detail tentang aplikasi yang akan dikembangkan.

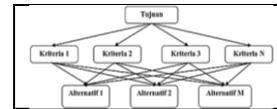
2. Metode AHP

Metode ini merupakan suatu model pendukung keputusan yang dapat memecahkan masalah kompleks, kriteria yang diambil cukup banyak, struktur masalah yang belum jelas, ketidakpastian persepsi pembuat keputusan, dan ketidakpastian tersedianya data statistik yang akurat. AHP pada umumnya menyelesaikan masalah multi-objektif dan multi-kriteria yang berdasar pada perbandingan preferensi dari setiap elemen dalam hirarki [7]. Dalam menggunakan metode AHP terdiri dari beberapa tahap yaitu [2] :

a. Penyusunan Hirarki

Persoalan yang akan diselesaikan, terlebih dahulu diuraikan menjadi sebuah kriteria dan

alternatif dan disusun menjadi sebuah struktur hirarki.



Gambar 2.1. Struktur Hirarki AHP

b. Pembobotan Kriteria dan Alternatif

Kriteria dan alternatif diberikan bobot masing-masing untuk mengetahui nilai *eigen vector* dan nilai konsistensi dari masing-masing kriteria dan alternatif. Bobot yang diberikan untuk masing-masing kriteria dan alternatif bernilai 1 sampai dengan 9.

Tabel 2.1. Tingkat Kepentingan

Skala	Inpretasi
1	O_i dan O_j sama penting
3	O_i sedikit lebih penting daripada O_j
5	O_i kuat tingkat kepentingannya daripada O_j
7	O_i sangat kuat tingkat kepentingannya daripada O_j
9	O_i mutlak lebih penting daripada O_j
2, 4, 6, 8	Nilai – nilai intermediate

c. Menentukan Prioritas Elemen

Setiap kriteria dan alternatif, perlu dilakukan perbandingan berpasangan untuk memperoleh nilai *eigen vector* dari masing-masing kriteria dan alternatif. Berikut adalah contoh matriks perbandingan berpasangan dalam AHP.

$$A = \begin{matrix} & \begin{matrix} O_1 & O_2 & \dots & O_n \end{matrix} \\ \begin{matrix} O_1 \\ O_2 \\ \vdots \\ O_n \end{matrix} & \begin{pmatrix} w_1/w_1 & w_1/w_2 & \dots & w_1/w_n \\ w_2/w_1 & w_2/w_2 & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ w_n/w_1 & w_n/w_2 & \dots & w_n/w_n \end{pmatrix} \end{matrix} \dots\dots(2.1)$$

Untuk memperoleh nilai *eigen vector* dapat dilakukan dengan cara membagi setiap elemen matriks dengan jumlah nilai setiap kolom yang bersangkutan untuk memperoleh matriks baru. Selanjutnya setiap elemen dari matriks baru dijumlahkan perbaris untuk memperoleh jumlah nilai perbaris. Kemudian jumlah nilai perbaris dari matriks baru dibagi dengan jumlah elemen yang ada untuk mendapatkan nilai rata-rata atau vektor bobot.

d. Mengukur Konsistensi

Dalam membuat keputusan, penting untuk mengetahui seberapa baik konsistensi yang ada. Konsistensi penting untuk mendapatkan hasil

yang valid dalam dunia nyata. AHP mengukur konsistensi pertimbangan dengan *consistency ratio*. Nilai *consistency ratio* harus kurang dari atau sama dengan 0,1. Jika nilai *consistency ratio* lebih dari batas *ratio* tersebut maka perbandingan matriks harus dilakukan pembobotan ulang pada tahap 2.

Hitung perkalian antara matriks awal dengan matriks *eigen*.

$$(A) * (W^T) \dots \dots \dots (2.2)$$

Hitung nilai t

$$t = \frac{1}{n} \sum_{i=1}^{i=n} \left(\frac{\text{elemen ke - i pada } (A)(W^T)}{\text{elemen ke - i pada } W^T} \right) \dots \dots \dots (2.3)$$

Menghitung *consistency index*

$$CI = \frac{t - n}{n - 1} \dots \dots \dots (2.4)$$

Menghitung *consistency ratio*

$$CR = \frac{CI}{RI} \dots \dots \dots (2.5)$$

e. Konsistensi Logis

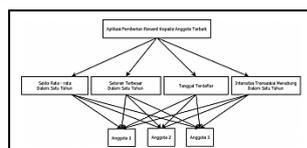
Semua elemen dikelompokkan secara logis dan diperingkatkan secara konsisten sesuai dengan suatu kriteria yang logis. Konsistensi logis merupakan karakteristik penting dalam AHP. Hal ini dicapai dengan mengagregasikan seluruh *eigen vector* yang diperoleh dari berbagai tindakan hirarki dan selanjutnya diperoleh suatu *vector composite* yang menghasilkan urutan pendukung keputusan. Tahap ini adalah tahap menjumlahkan bobot yang diperoleh setiap alternatif pada masing-masing kriteria setelah diberi bobot dari kriteria tersebut.

$$S_j = \sum_i (S_{ij})(W_i) \dots \dots \dots (2.6)$$

3. Mengolah data menggunakan metode AHP

Berikut adalah contoh proses mengolah data anggota koperasi untuk menentukan calon penerima *reward* menggunakan metode AHP :

a. Penyusunan Hirarki



Gambar 3.1. Struktur Hirarki Aplikasi Pemberian *Reward*

b. Pembobotan Kriteria dan Alternatif

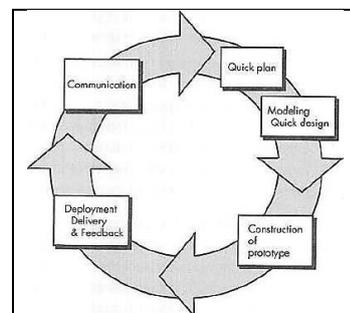
Tabel 3.1. Bobot Kriteria

Kriteria	Saldo Rata - rata Dalam 1 Tahun	Setoran Terbesar Dalam 1 Tahun	Tanggal Terdaftar	Intensitas Transaksi Menabung Dalam 1 Tahun
Skala	9	7	8	3

- c. Menentukan Prioritas Elemen
Setiap kriteria dan alternatif, perlu dilakukan perbandingan berpasangan untuk memperoleh nilai *eigen vector* dari masing-masing kriteria dan alternatif. Untuk mengetahui nilai *eigen vector* dapat menggunakan matriks (2.1).
- d. Mengukur Konsistensi
Normalisasi data kriteria dan alternatif adalah mengubah nilai kriteria dan alternatif menjadi antara 0 sampai 1. Untuk mengukur konsistensi terdapat empat langkah dengan menggunakan rumus (2.2) sampai dengan (2.5). Jika CR bernilai kurang dari sama dengan 0,1 maka CR konsisten. Apabila CR bernilai kurang dari sama dengan 0,1 maka CR tidak konsisten. Sehingga perlu dilakukan pembobotan ulang terhadap masing-masing kriteria dan alternatif.
- e. Konsistensi Logis
Untuk mencari peringkat setiap anggota koperasi didapatkan dengan mengalikan data *eigen vector* kriteria dengan *eigen vector* alternatif, seperti pada rumus (2.6).
- f. Menampilkan Peringkat
Peringkat anggota koperasi ditampilkan berdasarkan hasil dari perkalian antara *eigen vector* kriteria dengan *eigen vector* alternatif yang sudah dilakukan pada tahap konsistensi logis.

4. Model Proses Prototipe

Model proses protipe merupakan sebuah model proses pengembangan perangkat lunak yang terdiri dari beberapa tahapan yaitu *communication*, *quick plan*, *modeling quick design*, *contruction of prototype*, *deployment delivery and feedback* [4].



Gambar 3.1. Model Proses Prototipe

Model proses prototipe dimulai dengan tahapan *communication* pada tahapan ini, pengembang

perangkat lunak bertemu dengan pengguna perangkat lunak untuk mendefinisikan tujuan perangkat lunak dan menentukan kebutuhan perangkat lunak.

Lalu berlanjut ke tahapan *quick plan* melakukan perencanaan iterasi *prototyping*, *modelling* membuat pemodelan tentang perangkat lunak yang akan dibuat. *Quick desain* membuat rancangan tampilan perangkat lunak, misal *interface* pengguna dan format keluaran.

Tahapan *construction of prototype* membuat prototipe berdasarkan model dan desain yang sudah dibuat pada tahapan sebelumnya. Lalu pada tahapan *deployment delivery & feedback* pengembang perangkat lunak memberikan prototipe ke pengguna, lalu pengguna melakukan pengujian terhadap prototipe.

5. UML

Unified Modeling Language (UML) merupakan bahasa standar untuk penulisan mengenai gambaran umum dari perangkat lunak [1]. UML dapat digunakan untuk memvisualisasikan, membangun dan mendokumentasikan pengembangan perangkat lunak.

6. OOAD

OOAD adalah sebuah metode yang menjelaskan tentang pemecahan masalah berbasis objek. OOAD mendefinisikan suatu proses untuk mengembangkan sistem aplikasi yang rumit, serta menawarkan banyak model logika dan model fisik yang berhubungan dengan aspek pengembangan sistem. Tahap OOAD terdiri dari analisis dan desain [3]. Pada tahap analisis terdiri dari empat belas step. Diantaranya adalah *Developing the System Idea and Objective, Identifying Stakeholders, Identifying Business Processes, Identifying Stakeholders' Interests, Identifying Business Use Cases, Describing the Essence of Use Cases, Identifying System Use Cases Collecting and Studying Materials, Describing the Requirements, Identifying Business Classes, Creating a Technical Dictionary, Developing a Use Case Process Model, Describing the System Interface, Explorative Interface Prototyping.*

Sedangkan pada tahap desain terdiri dari dua belas step. Diantaranya adalah *Defining the Application Architecture, Identifying Domain Components, Developing Component-specific Class Models, (Further) Developing State Models, Identifying and, if Necessary, Restructuring Component Dependencies, Designing Component Interfaces, Developing Collaboration Models, Developing Process-oriented Component Tests, Developing Class Tests, Defining Attributes, Specifying Dialogs, Discussion of Design.*

7. Pemrograman Berorientasi Objek Pada *Visual Basic .NET*

Visual basic .net adalah salah satu bentuk bahasa pemrograman yang sudah mendukung pemrograman berorientasi objek. Dengan menggunakan *visual basic .net*, kita dapat membuat kelas dan menerapkan *encapsulation, inheritance* dan *polymorphism*.

8. Prototipe Aplikasi Iterasi Pertama

Pada subbab ini akan membahas tahap *communication, quick plan, modelling quick design, construction of prototype, dan deployment delivery and feedback* dari prototipe aplikasi pemberian *reward* iterasi pertama.

a. *Communication*

Aplikasi pemberian *reward* dibangun dalam upaya untuk memberikan kemudahan berupa data pendukung bagi pihak koperasi dalam menentukan calon penerima *reward*. Anggota yang layak untuk menerima *reward* adalah anggota yang memenuhi syarat kriteria penilaian yang sudah ditentukan oleh pihak koperasi. Kriteria penilaian tersebut antara lain adalah saldo rata-rata dalam satu tahun, setoran terbesar dalam satu tahun, tanggal terdaftar, dan intensitas transaksi menabung dalam satu tahun.

b. *Quick Plan*

Kegiatan ini akan membuat perencanaan iterasi pada prototipe aplikasi pemberian *reward* menggunakan metode AHP. Berdasarkan informasi kegiatan yang sudah diperoleh maka dapat dibuat daftar spesifikasi aplikasi pemberian *reward* yang ditampilkan pada tabel 7.1.

Tabel 7.1. Spesifikasi Aplikasi Pemberian *Reward* Iterasi Pertama

Kode	Use case
U1	Memasukkan data anggota
U2	Mengubah data anggota
U3	Menghapus data anggota
U4	Mengelola nilai perbandingan berpasangan kriteria
U5	Mengelola nilai perbandingan berpasangan alternatif
U6	Mengelola aturan <i>range</i> penilaian

c. *Modelling and Quick Design*

Pada tahap ini akan mengerjakan model dan desain yang diperlukan untuk membuat aplikasi.

d. *Construction of Prototype*

Pada tahap ini akan menampilkan hasil dari prototipe iterasi pertama yang sudah dibuat pada tahap *modeling and quick design*.

e. *Deployment Delivery and Feedback*

Pada tahap ini *user* akan melakukan evaluasi pada prototipe dan memberikan *feedback* tentang prototipe.

9. Prototipe Aplikasi Iterasi Kedua

Pada subbab ini akan membahas tahap *communication*, *quick plan*, *modelling quick design*, *construction of prototype*, dan *deployment delivery and feedback* aplikasi pemberian *reward* iterasi kedua.

- a. *Communication*
Berdasarkan hasil komunikasi yang telah dilakukan dengan admin dan manajer koperasi, terdapat beberapa kegiatan yang akan dikerjakan pada iterasi kedua. Kegiatan tersebut antara lain menentukan prioritas tertinggi dan mencetak laporan.
- b. *Quick Plan*
Berdasarkan informasi kegiatan yang sudah diperoleh maka dapat dibuat daftar spesifikasi aplikasi pemberian *reward* yang ditampilkan pada tabel 8.1.

Tabel 8.1. Spesifikasi Aplikasi Pemberian *Reward* Iterasi Kedua

Kode	Use case
U7	Menentukan prioritas tertinggi
U8	Mencetak laporan

- c. *Modelling and Quick Design*
Pada tahap ini akan mengerjakan model dan desain yang diperlukan untuk membuat aplikasi pemberian *reward*.
- d. *Construction of Prototype*
Pada tahap ini akan menampilkan hasil dari prototipe iterasi kedua yang sudah dibuat pada tahap *modeling and quick design* dan *feedback* pada iterasi pertama.
- e. *Deployment Delivery and Feedback*
Pada tahap ini *user* akan melakukan evaluasi pada protipe dan memberikan *feedback* tentang prototipe.
- f. Hasil Akhir
Pada bagian hasil akhir akan menampilkan hasil akhir dari prototipe aplikasi.

10. Analisis dan Perancangan Sistem

Pada bab ini akan membahas analisa dan perancangan sistem dari prototipe aplikasi pemberian *reward* dengan menggunakan metode OOAD. Subabb pada bab ini terdiri dari analisis dan desain.

10.1. Analisis

Pada subbab analisis membahas tentang *Describing the Requirements*, *Identifying System Use Cases*, dan *Describing the System Interface* aplikasi pemberian *reward* yang akan disajikan pada subbab di bawah.

- a. *Describing the Requirements*

Dalam membangun aplikasi pemberian *reward* diperlukan spesifikasi kebutuhan yang jelas sebagai tujuan utamanya agar tidak keluar dari rencana yang telah ditetapkan. Beberapa spesifikasi sistem yang didefinisikan diantaranya adalah :

Tabel 9.1. Spesifikasi Kebutuhan Fungsional Aplikasi Pemberian *Reward*

SRS ID	Deskripsi
SRS-SPK-R-F-01	Memasukkan data anggota
SRS-SPK-R-F-02	Mengubah data anggota
SRS-SPK-R-F-03	Menghapus data anggota
SRS-SPK-R-F-04	Mengelola nilai perbandingan berpasangan kriteria
SRS-SPK-R-F-05	Mengelola nilai perbandingan berpasangan alternatif
SRS-SPK-R-F-06	Mengelola aturan <i>range</i> penilaian
SRS-SPK-R-F-07	Menentukan prioritas tertinggi
SRS-SPK-R-F-08	Mencetak laporan

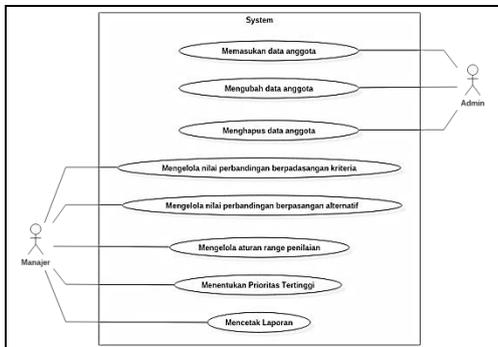
- b. *Identifying System Use Cases*
Use case diagram digunakan untuk memodelkan hubungan aktor dengan *use case*. Berikut adalah daftar aktor dalam aplikasi pemberian *reward* :

Tabel 9.2. Daftar Actor

Aktor	Deskripsi
Manajer koperasi	Manajer memiliki hak akses mengelola nilai perbandingan berpasangan kriteria, mengelola nilai perbandingan berpasangan alternatif, menentukan prioritas tertinggi, mengelola aturan <i>range</i> penilaian, dan mencetak laporan.
Admin koperasi	Admin bertugas mengelola data anggota Koperasi Simpan Pinjam Utama Karya Jepara yaitu memasukan data anggota dan kelola data anggota yang nantinya akan digunakan untuk penilaian anggota terbaik.

Setelah memiliki daftar *actor* dan daftar *use case* aplikasi pemberian *reward*, langkah

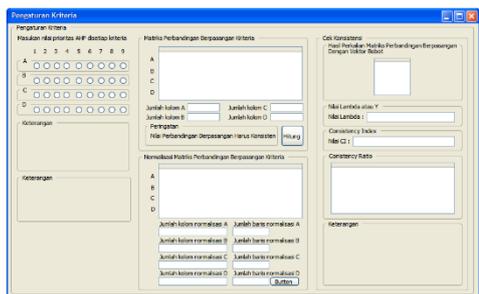
selanjutnya adalah membuat *use case* diagram. Berikut adalah *use case* diagram dari aplikasi pemberian *reward*.



Gambar 9.1. *Use Case* Diagram Aplikasi Pemberian *Reward*

c. *Describing the System Interface*

Pada subbab ini akan mengerjakan bagian desain antarmuka aplikasi pemberian *reward* yang dibuat berdasarkan *use case*. Berikut adalah salah satu contoh dari antarmuka aplikasi pemberian *reward*.



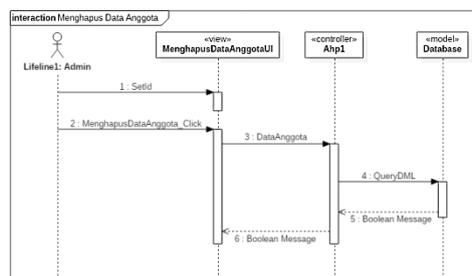
Gambar 9.2. Antarmuka Mengelola Nilai Perbandingan Berpasangan Kriteria

10.2. Desain

Pada subbab desain membahas tentang *Developing Collaboration Models*, *Developing Component-specific Class Models*, dan *Defining Attributes* aplikasi pemberian *reward* yang akan disajikan pada subbab di bawah.

a. *Developing Collaboration Models*

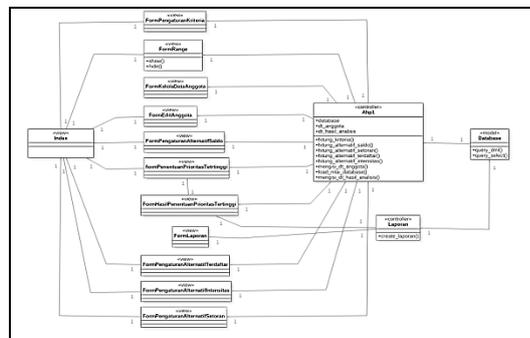
Sequence diagram digunakan untuk menggambarkan rincian dari sebuah skenario yang terjadi pada masing-masing *use case*. Melalui *sequence* diagram juga dapat terlihat urutan-urutan operasi yang terjadi di dalam skenario dan lalu lintas pesan yang terjadi. Berikut adalah salah satu contoh dari *sequence* diagram aplikasi pemberian *reward*.



Gambar 9.3. *Sequence* Diagram Menghapus Data Anggota

b. *Developing Component-specific Class Models*

Class diagram adalah sebuah diagram yang digunakan untuk menggambar struktur dan relasi sekumpulan *class*. Berikut adalah *class* diagram dari aplikasi pemberian *reward*.



Gambar 9.4. *Class* Diagram Aplikasi Pemberian *Reward*

c. *Defining Attributes*

Perancangan basis data mendeskripsikan macam-macam tabel dalam sistem yang digunakan untuk membangun aplikasi pemberian *reward*. Berikut adalah salah satu contoh dari perancangan basis data aplikasi pemberian *reward* :

Tabel 9.3. Daftar Tabel Bobot Kriteria

Field	Type
id	int
nama_kriteria	varchar(100)
nilai_kriteria	float

11. Implementasi

Bab ini berisi tentang implementasi aplikasi pemberian *reward*. Implementasi aplikasi meliputi implementasi *class*, implementasi basis data, dan implementasi antarmuka.

a. Implementasi *class*

Class *ahp1*, *Class* *database*, dan *Class* *laporan* yang sudah dibuat pada subbab *Developing Component-specific Class Models* akan diimplementasikan ke dalam bentuk bahasa pemrograman VB .Net.

Berikut adalah salah satu contoh dari implementasi *Class* laporan pada aplikasi pemberian *reward*.

Implementasi *Class* laporan Aplikasi Pemberian *Reward*

```
Public Class Laporan
Public Sub create_laporan(ByVal
data_input As DataTable)
Public Sub cetak()

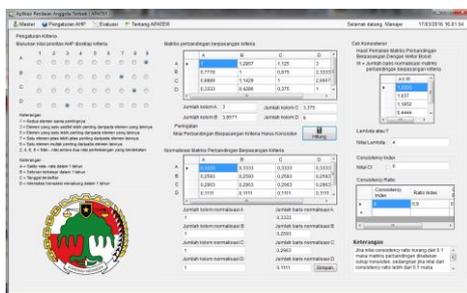
End Class
```

- b. Implementasi basis data
Pada tahap implementasi basis data akan membuat tabel dan *field* berdasarkan *Defining Attributes* yang berisi tentang perancangan basis data aplikasi pemberian *reward*. Berikut adalah salah satu contoh dari implementasi basis data aplikasi pemberian *reward*.

Tabel 10.1. Implementasi Tabel Bobot Kriteria

Nama field	Type & Length	Keterangan	Deskripsi
id	Int	Primary Key	Identitas kriteria penilaian
nama_kriteria	Varchar (100)	-	Nama kriteria penilaian
nilai_kriteria	Float	-	Nilai bobot setiap kriteria penilaian

- c. Implementasi antarmuka
Pada tahap ini akan mengimplementasikan desain antarmuka pengguna yang sudah dibuat pada tahap *Describing the System Interface*. Berikut adalah salah satu contoh dari implementasi antarmuka aplikasi pemberian *reward*.



Gambar 10.1. Implementasi Antarmuka Mengelola Nilai Perbandingan Berpasangan Kriteria

Bab ini berisi tentang kesimpulan dan saran dalam penyusunan tugas akhir yang berjudul Sistem Pendukung Keputusan Pemberian *Reward* Kepada Anggota Terbaik Menggunakan Metode *Analytical Hierarchy Process*.

- a. Kesimpulan
Kesimpulan yang dapat diambil dari pembuatan tugas akhir ini adalah dengan menggunakan model proses prototipe, sistem pendukung keputusan pemberian *reward* kepada anggota terbaik dibuat sesuai dengan kebutuhan pengguna. Dengan adanya sistem pendukung keputusan pemberian *reward* kepada anggota terbaik menggunakan metode *Analytical Hierarchy Process*, manajer koperasi dapat memperoleh data pendukung yang dapat digunakan untuk menentukan calon penerima *reward*.
- b. Saran
Aplikasi pemberian *reward* sebaiknya ditambahkan fitur untuk mengelola jumlah kriteria penilaian jika suatu saat terjadi penambahan kriteria penilaian oleh manajer koperasi. Pada fitur mengelola aturan *range* penilaian sebaiknya manajer koperasi diberi kemudahan klasifikasi secara otomatis dalam menentukan batasan *range* untuk setiap alternatif.

Referensi

- [1]. Booch, G., Rumbaugh, J. & Jacobson, I., 2005. *The Unified Modeling Language User Guide*. 2nd ed. Addison Wesley Professional.
- [2]. Kusumadewi, S., Hartati, S., Wardoyo, R. & Harjoko, A., 2006. *Fuzzy Multi-Attribut Decision Making (Fuzzy MADM)*. 1st ed. Yogyakarta: Penerbit Graha Ilmu.
- [3]. Oestereich, B., 2002. *Developing Software With UML*. Second Edition ed. London: Oldenbourg.
- [4]. Pressman, R.S., 2012. *Rekayasa Perangkat Lunak*. 7th ed. Yogyakarta: Penerbit ANDI.
- [5]. Rosa & Shalahuddin, 2013. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- [6]. Stephens, R., 2010. *Visual Basic 2010 : Programmer's Reference*. Canada: Wiley Publishing Inc.
- [7]. Suryadi, K. & Ramdhani, A., 1998. *Sistem Pendukung Keputusan : Suatu Wawancara Structural Idealisasi dan Implementasi Konsep Pendukung Keputusan*. Bandung: Penerbit PT Remaja Rosdakarya.

12. Kesimpulan dan Saran