# BAB II LANDASAN TEORI

## Kajian Penelitian Terdahulu

### Pemilihan pemain terbaik futsal terbaikmenggunakan metode simple multi attribute rating technique studi kasus: turnamen futsal di Samarinda

Menurut Heliza Rahmania Hatta Dkk. (2017). Dalam jumlah pemain yang banyak, proses perhitungan dengan menggunakan metode ini sangat memudahkan panitia. Panitia akan mendapatkan rekomendasi pemain terbaik dalam satu turnamen tersebut. Hasil perhitungan menggunakan metode SMART tidak jauh berbeda dibandingan yang menggunakan program, Program yang dibuat menggunakan metode SMART pada studi kasus ini adalah benar.

### Penerapan metode VIKOR pada pengambilan keputusan seleksi calon penerima beasiswa Bidikmisi di Universitas Terbuka

Menurut Gede Suwardika Dkk. (2018). Hasil penelitian menunjukkan, metode VIKOR dapat membantu proses seleksi dan penentuan penerimaan beasiswa Bidikmisi berdasarkan kriteria yang telah ditetapkan sebelumnya. Hasil pemeringkatan yang sama, ditunjukkan berdasarkan bobot yang diberiikan, sehingga permasalahan multikriteria sudah mendapatkan solusi kompromi. Urutan mutu sekolah dapat sedikit lebih penting daripada asal daerah dan keterwakilan sekolah.

### Penerapan metode VIKOR dalam penerimaan dana untuk rumah tidak layak huni

Menurut Hotmoko Tumanggor DKK. (2018). Kesimpulan yang diperoleh dari penelitian ini, metode VIKOR dapat menentukan peserta yang berhak menerima dana RUTILAHU dengan tepat dan melakukan pemeringkatan secara efektif.

### Penerapan strategi pemilihan pemain futsal menggunakan metode elektre

Menurut Ema Kurnia DKK. (2018). Aplikasi strategi pemilihan pemain futsal ini dirancang dengan metode Electre. Metode electre dapat digunakan dalam kondisi dimana alternatif yang tidak memenuhi kriteria dihilangkan dan alternatif yang sesuai dapat dihasilkan. Aplikasi ini dirancang dengan menerapkan siklus hidup pengembangan sistem yaitu System Development Live Cycle (SDLC) dengan melalui serangkaian tahapan dari perancangan basis data hingga perancangan metode elektre untuk menghasilkan identifikasi sistem awal untuk kemudian diimplementasikan ke dalam suatu sistem.

### Penerapan metode VIKOR pada pemilihan popok bayi berdasarkan jenis kulit

Menurut Tia Imandasari DKK. (2018). Berdasarkan penelitian menunjukkan bahwa vikor bisa membantu untuk menentukan popok bayi dengan jenis kulit berdasarkan kriteria yang ditentukan dan sudah diberikan penilaian bobot. Metode VIKOR berguna untuk membantu pengambilan keputusan sehingga bisa mengurutkan alternatif dari sebagian alternatif yang sudah ada.

## Pro Futsal League (PFL)

Liga Futsal Pro adalah kompetisi futsal Liga Futsal Profesional Indonesia musim kedua belas yang diselenggarakan oleh Federasi Futsal Indonesia (FFI), sekaligus musim keempat kompetisi futsal dengan nama "Liga Futsal Pro". Kompetisi ini dimulai pada musim 2006 – 2007 dan hingga saat ini sudah terselenggara sebanyak tiga belas kali. Pada awal musim terselenggaranya Liga Futsal Profesional Indonesia ini hanya diikuti 6 klub pro yang sudah disahkan oleh Badan Futsal Nasional (BFN) saat itu, yaitu Biangbola Jakarta, Cosmo Jakarta, Mastrans Jakarta, Yogyakarta Electric, Pro Duta Bandung, dan SWAP Futsal Bogor.

Federasi Futsal Indonesia (FFI) merupakan badan resmi dibawah PSSI yang mengurusi olahraga futsal di Indonesia. FFI adalah Federasi Futsal Indonesia dan anggota dari Persatuan Sepak Bola Seluruh Indonesia (PSSI). Organisasi ini merupakan satu-satunya induk organisasi Futsal nasional di wilayah hukum Republik Indonesia, serta diberikan kewenangan penuh oleh PSSI, untuk mengatur, mengurus dan menyelenggarakan seluruh kegiatan Futsal di wilayah Republik Indonesia.

Aturan permainan futsal resmi berdasarkan FIFA sebagai berikut :

1. Waktu

Futsal menggunakan sistem dua babak. Setiap putaran memiliki waktu 20 menit, tidak termasuk saat bola diset atau pertandingan terhenti dikarenakan suatu insiden. Jika habis, akan mendapatkan perpanjangan waktu jika terjadi pinalti atau tendangan bebas. Mendapatkan waktu istirahat selama 15 menit.

1. Time Out

Setiap babak aka nada tim yang mendapat waktu Time-out satu menit. Pelatih kepala dapat meminta Time-out ketika timnya sedang menguasai bola. Apabila tim tidak mengambil time-out di babak pertama, itu akan otomatis hilang dan tidak bisa digabungkan saat babak kedua.

1. Pelanggaran

Hal – hal yang tidak boleh dilakukkan oleh pemain futsal saat pertandingan, ialah :

1. Pelanggaran berat: Menendang lawan, Menyerang atau menjegal lawan, Melompati lawan, Memukul lawan, Mendorong lawan, Menarik baju lawan, Meludahi lawan, Menghalangi lawan yang tidak menguasai bola dan sengaja memegang bola.
2. Pelanggaran ringan: Penjaga gawang tidak boleh mengambil bola dari rekan setim tanpa menyentuh lawan terlebih dahulu. Penjaga gawang menguasai bola melebihi empat detik, kecuali melakukannya diarea lawan serta Mencegah kiper lawan melepaskan bola.
3. Kartu
4. Kartu Kuning

Secara individual, sebelum diberikan kartu kuning, pemain akan diberikan peringatan terlebih dahulu. Pemain mendapat kartu kuning apabila Pemain tidak sportivitas, Menggunakan bahasa yang kasar, Terus melanggar walau tidak diberi peringatan, Mengulur waktu, Memutus jarak saat tendangan penalti, Melanggar prosedur pergantian pemain, meninggalkan lapangan dengan sengaja tanpa seizin wasit.

1. Kartu Merah

Seorang pemain yang diberikan kartu merah diharuskan meninggalkan lapangan permainan dan tidak diperbolehkan bermain lagi. Kartu merah diberikan jika: Bersalah, bermain sangat kasar, Bersalah, berperilaku tidak pantas, Meludahi lawan atau orang lain, Menghalangi gawang tim lawan dengan menggunakan tangan, menggagalkan peluang pemain dalam mencetak gol, melontarkan kata yang tidak pantas , Menerima dua kartu kunin. Ada ketentuan perbedaan kartu merah dalam futsal dan sepakbola. Dalam futsal, tim yang pemainnya dikartu merah berhak memasukkan pemain pengganti dua menit setelah kartu merah diberikan, kecuali sebelum dua menit terjadi gol. Apabila terjadi gol, keadaan seperti :

* 1. Apabila posisinya 4 lawan 5 pemain, ada tim yang mendapat gol sebelum 2 menit, maka tim yang pemainnya lebih banyak dapat mengganti pemain.
  2. Apabila kedudukan tim pemain sama dan terjadi gol, tidak akan merubah apapun dan tetap melanjutkan pertandingan
  3. Apabila kedudukannya lima versus tiga ataupun empat versus tiga, tim yang mendapat gol sebelum 2 menit, tim yang berisi tiga pemain hanya boleh memasukkan satu pemain.
  4. Apabila kedudukan tim pemain sama dan tercipta gol, tidak merubah apapun dan tetap melanjutkan pertandingan seperti biasa.
  5. Apabila tim dengan pemain lebih sedikit dan mencetak gol sebelum dua menit, tidak akan merubah apapun dan tetap melanjutkan pertandingan.

1. Tendangan Hukuman
   1. Tendangan bebas tidak langsung. Akan diberikan ke tim lawan apabila melakukkan pelanggaran. Dilakukkan di garis kotak penalty terdekat apabila terjadi pelanggaran di dalam kotak penalti. Dinyatakan menjadi gol apabila bola menyentuh pemain lain sebelum masuk kegawang. Dapat membuat tembok pertahanan minimal 5 meter dari bola.
   2. Tendangan bebas langsung. Jika melakukkan pelanggaran serius. Gol bisa langsung tercipta lewat tendangan ini. Dapat membuat tembok pertahanan min 5m dari bola.
2. Akumulasi Pelanggaran

Apabila tim melakukan pelanggaran lebih dari 5x dalam 1 babak, tim lawan memperoleh tendangan bebas dengan ketentuan :

* 1. Tidak ada tembok pertahanan dikarenakan ini merupakan second penalty yang diberikan ke tim lawan.
  2. Tendangan langsung ke gawang.
  3. Pemain tidak boleh menyentuh bola sebelum bola mengenai pemain lawan atau tiang gawang.
  4. Apabila terjadi didekat atau didalam kotak pinalti, tendangan dilakukan dari titik pinalti terdekat.
  5. Tendangan bebas akan dilakukan dari titik penalti kedua.

## Sistem Pendukung Keputusan

Sistem pendukung keputusan adalah suatu proses pendukung keputusan yang dibantu dengan menggunakan komputer untuk membantu para pengambil keputusan dengan menggunakan data dan model tertentu untuk memecahkan beberapa masalah yang ada. Keberadaan Sistem ini sangat membantu mereka sebagai alat dalam pengambilan suatu keputusan.

Masalah yang tidak terstruktur akan didukung menggunakan data yang telah dikelola menjadi informasi. Dalam penerapan SPK, hasil dari sistem, hal yg mendasari pengambilan keputusan tetap berada dalam kendali seseorang yang menggunakan sistem keputusan. Output yang diberikan akan menghitung data untuk pertimbangan pengambil keputusan, sehingga akan memudahkan pekerjaan dalam pengambilan keputusan. (Wibowo , 2011).

SPK ini dibuat dalam beberapa tahapan tinggkatan seperti identifikasi masalah, pemilihan data yg akurat, dan menentukan pendekatan hingga menemukan alternatif pilihan. (Fitrianii , 2012).

Adapaun karateristik sistem penunjang keputusan dari Wibowo (Wibowo, 2011) :

1. SPK ini dirancang agar memudahkan dalam menyelesaikan semi terstruktur atau tidak terstruktur dengan menambahkan informasi manusia dan komputerisasi.
2. Dalam pengerjakannya, SPK ini menggabungkan analisis model dengan teknik entri data konvensional dan fungsi pencarian/interogasi informasi.
3. Agar digunakan dengan mudah, sistem ini dirancang sedemikian mudah mungkin.
4. Dibuat sesuai dengan adaptasi tinggi dan fleksibilitas.
5. Dengan berbagai karakter yang dijelaskan di atas, dapat diambil manfaatnya seperti :
6. Meluaskan keputusan dalam mengolah informasi dan data untuk pengguna.
7. Memudahkan dalam menyelesaikan suatu masalah yang tidak terstruktur dan sangat kompleks.
8. SPK ini membuat rekomendasi alternatif lebih cepat dan hasilnya dapat digunakan.
9. Meskipun terkadang sistem ini kurang bisa menyelesaikan masalah, dikarenakan memiliki berbagai alternatif dalam pemecahannya, SPK ini bisa membantu dalam mengambil suatu keputusan.

## VIšekriterijumsko Kompromisno Rangiranje (VIKOR)

VIKOR adalah metode analisis pengambilan keputusan dengan pengambilan keputusan multi-atribut yang dikembangkan oleh Serafim Opricovic untuk menyelesaikan pemecahan masalah dengan kriteria yang saling bertentangan dan dari unit yang berbeda, dengan asumsi kompromi dapat diterima sebagai penyelesaian dari konflik yang ada. Pengambil keputusan menginginkan solusi yang mendekati ideal dan setiap alternatif sesuai dengan kriteria yang telah ditentukan. Vikor membuat peringkat alternatif dan menentukan solusi yang mendekati solusi kompromi yang ideal.

### Kelebihan Metode VIKOR

Kelebihan metode VIKOR, diantaranya :

1. Metode VIKOR memiliki kelebihan dalam proses pemeringkatan dengan memiliki tingkat preferensi untuk pemeringkatan dan dapat mengatasi pemeringkatan banyak alternatif dengan lebih mudah.
2. Dalam mengatasi kriteria yang bertentangan didalam pemeringkatan, metode VIKOR lebih memiliki keunggulan. Beberapa kriteria yang bertentangan yaitu terdapat kriteria namun dari kriteria tersebut dapat berbeda beda. Penilaian bisa dilihat dari penilaian tertinggi terbaik dan penilaian terendah terbaik.

### Langkah-langkah penyelesaiaan MADM menggunakan VIKOR

Rumusan0masalah0MADM: Tentukan0solusi0terbaik0pada0beberapa atribut0dari0himpunan0alternatif J A1, A2, ..., AJ, kemudian0dievaluasi0menurut himpunan n fungsi kriteria. Data0masukannya0adalah0matriks0keputusan (kinerja) fij, dimana0fij0adalah0nilai0fungsi0kriteria ke-i untuk setiap alternatif Aj.1.0Tentukan0nilai0fi\*0terbaik0dan0nilai0fi0terburuk0untuk0semua0fungsi0kriteria, i= 1,2,..,n; Jika fungsi kriteria ia adalah kriteria pemenang, maka fi\* = max(fij, j=1,...,J) dan fi-=min(fijj = 1,...,J), sedangkan jika kriteria i bernilai kriteria, yaitu. fi\*=min(fij, j=1,...J), dan fi-=max(fij, j=1,...,J). ). 2. Hitung nilai Sj dan Rj, j=1,2,...J, di mana wi adalah bobot kriteria yang, berdasarkan penilaian pengambil keputusan, memberikan kepentingan yang cukup bagi kriteria tersebut. 3. Dengan menggunakan persamaan, S\* = min Sj lakukan j=1,...,J, hitung nilai Qj, j=1,2,...,J; S- = maks Sj untuk j=1,...,J R\*= min Rj untuk j=1,...,J; R- = max Rj for j=1,...,J v = bobot strategis aturan mayoritas atau utilitas maksimum grup, dalam hal ini v = 0,5 4. Buatlah alternatif rangking sesuai dengan aslinya nilai minimum S, R dan Q, dimana hasilnya adalah 3 derajat. 5. Mengusulkan solusi kompromi alternatif A(1) yang (setidaknya) yang terbaik dalam ukuran Q sehingga memenuhi dua kondisi berikut: C1 : "Keunggulan yang Dapat Diterima - Keuntungan yang Dapat Diterima" , di mana Q( A( 2)) - Q(A(1)) DQ, di mana A(2) adalah alternatif menggunakan posisi sekunder Q; DQ=1/(J - 1), di mana J adalah jumlah alternatif C2: "Stabilitas yang Dapat Diterima dalam Pengambilan Keputusan", di mana alternatif A(1) juga harus dinilai terbaik oleh S dan/atau R. Solusi konsensus ini tetap dalam proses pengambilan keputusan, yaitu: o "mayoritas" (jika v > 0,5) atau o "dengan konsensus" di mana v » 0,5 atau o "dengan veto" untuk . v < 0,5 Jika salah satu kondisi tidak terpenuhi, serangkaian solusi kompromi diusulkan, yang meliputi: o alternatif A(1) dan A(2), jika hanya kondisi C2 yang tidak terpenuhi, atau o alternatif A (1). ), A(2),...,A(M) jika kondisi C1 tidak terpenuhi; dimana A(M) adalah relasi Q(A(M)) - Q(A(1)) < DQ untuk M max (posisi alternatif ini saling berdekatan). Alternatif terbaik yang dicantumkan oleh Q adalah alternatif dengan nilai Q terkecil.

## UML (*Unified Modeling Language*)

UML (Unified Modeling Language) adalah keluarga notasi grafis yang didukung oleh meta-model tunggal yang digunakan dalam deskripsi dan desain sistem perangkat lunak, terutama sistem yang dibangun menggunakan pemrograman berorientasi objek (Fowler, 2005).

Menurut Windu Gata, Grace (2013:4), *Unified Modeling Language* (UML) adalah spesifikasi standar bahasa yang digunakan untuk membangun perangkat lunak. UML merupakan pengaturan dalam pengembangan sistem berorientasi objek dan juga merupakan alat yang mendukung pengembangan sistem.

UML (*Unified Modeling Language*) merupakan gabungan metodologi dari metode Booch, OMT (*Object Modeling Technique*), dan OOSE (*Object Oriented Software Engineering*) dan beberapa metode lainnya, UML sendiri merupakan metodologi yang paling sering digunakan saat ini untuk menganalisis dan mendesain suatu sistem. metodologi berorientasi objek.

### Use Case Diagram

*Use Case* adalah teknik yang digunakan dalam perancangan suatu perangkat lunak untuk melengkapi kebutuhan fungsional dari suatu sistem yang bersangkutan, *Use Case* menunjukkan interaksi antara 'aktor', dari interaksi sistem dengan sistem yang ada, *Use Cas*e diwakili oleh urutan langkah sederhana.

Aktifitas sistem adalah bagaimana sistem bertindak dan bereaksi. Ini merupakan aktivitas sistem yang dapat diamati dan diukur. Sistem ini ditangkap dalam *Use Case. Use case* itu sendiri menggambarkan suatu sistem, lingkungan sistem, dan hubungan antara sistem dan sistem lainnya. Manfaat memiliki *use case* adalah::

1. Digunakan untuk berkomunikasi dengan *end user* dan domain *expert.*
2. Untuk memahami tentang kebutuhan sistem.
3. Digunakan untuk mengidentifikasi siapa yang berinteraksi dengan sistem dan apa yang harus dilakukan sistem.
4. *Interface* yang harus dimiliki sistem.
5. Digunakan untuk verifikasi.

Sedangkan dari segi karakteristik, karakteristik dari *Use Case* adalah sebagai berikut:

1. *Use cases* ialah hubungan antara sistem dan *actor*, termasuk tindakan yang dilakukan oleh sistem dan aktor.
2. *Use cases* dipelopori oleh aktor dan mungkin melibatkan peran aktorlain.
3. *Use cases* memiliki perluasan yang diperjelas tindakan khusus dalam interaksi dan use case lain mungkin disisipkan.
4. *Use case* class memiliki suatu objek yang biasa disebut skenario. Skenario mendiskripsikan urutan pesan dan tindakan tunggal.

Untuk mendukung dalam pembuatan *Use Case*, berikut adalah penjelasan mengenai komponen yang terdapat pada *Use Case* yaitu:

**Tabel 2.** Simbol Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Gambar** | **Nama** | **Keterangan** |
| 1 |  | *Actor* | Menggambarkan seseorang yang berinteraksi dengan sistem, yang hanya dapat memasukkan dari menerima informasi dan tidak mengontrol use case. Dan aktor biasanya digambarkan sebagai stickman. |
| 2 |  | *Use case* | Gambaran fungsional sistem, untuk memudahkan pengguna dalam memahami sistem. |
| 3 |  | *Association* | Penghubung diatara elemen yang menghubungkan antara *Actor* dan *Use Case*. |
| 4 |  | *Generalization* | Sebuah elemen yang menjadi spesialisasi elemen lain. |
| 5 |  | *Dependency* | Sebuah elemen yang bersandar dalam beberapa cara terhadap elemen lain. |
| 6 |  | *Include* | Menentukan bahwa *use case* adalah sumber eksplisit. |
| 7 |  | *Extend* | Menentukan bahwa *use case* target memperluas perilaku use case pada titik tertentu. |
| 8 |  | *System* | Menspesifikasikan paket yang menampilkan sistem secara terbatas. |
| 9 |  | *Collaboration* | Interaksi aturan-aturan dan elemen-elemen lain yang bekerja sama memberikan perilaku yang lebih besar dari penjumlahan elemen-elemen tersebut. |
| 10 |  | *Note* | Elemen fisik yang ada saat aplikasi dijalankan dan mewakili sumber daya komputasi. |

### Activity Diagram

***Activity Diagrams* merupakan sebuah Teknik dalam penggambaran jalur kerja, proses bisnis, serta logika prosedural. Diagram mirip dengan aliran data, tetapi perbedaan utama antara diagram aktivitas lebih menguntungkan untuk *parallel behavior.***

***Activity diagram*** merupakan diagram yang menggambarkan aktivitas pada suatu sistem. Dari awal hingga akhir, diagram ini menunjukkan langkah-langkah apa saja yang terjadi dalam proses sistem kerja yang dibuat. Sehingga dengan menggunakan diagram aktivitas dapat mempermudah proses pembuatan sistem (Sukamto dan Shalahuddin, 2013).

**Tabel 3.** Simbol Activity Diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Simbol** | **Nama Simbol** | **Keterangan** |
| 1 |  | *Activity* | Menunjukkan bagaimana setiap kelas antarmuka berinteraksi dengan atau satu sama lain. |
| 2 |  | *Initial* *Node* | objek diawali atau dibentuk. |
| 3 |  | *Activity Final Node* | objek diakhiri. |
| 4 |  | *Decision* | Untuk mengambil tindakan yang harus dilakukan dalam kondisi tertentu. |
| 5 |  | *Line Conector* | Penghubung antar simbol. |

### Sequence Diagram

*Sequence diagrams* adalah deskripsi perilaku tunggal seperti menampilkan sejumlah objek sampel dan data yang akan lewat objek ini dalam *use case* (Sukamto dan Shalahuddin, 2013).

**Tabel 4.** Simbol Sequence Diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Simbol** | **Nama Simbol** | **Keterangan** |
| 1 |  | *Actor* | Mewakili entitas yang berada di luar sistem, dapat berupa manusia, atau perangkat sistem lainnya. |
| 2 |  | *LifeLine* | Objek entity, antarmuka yang berinteraksi. |
| 3 | Message() | Message | Spesifikasi komunikasi antar objek yang berisi informasi tentang aktivitas yang terjadi. |
| 4 | Message() | *Self Message* | Relasi ini menunjukkan suatu objek ingin memanggil dirinya sendiri. |
| 5 |  | *LifeLine Boundary* | Digunakan untuk menggambarkan bentuk. |
| 6 |  | *LifeLine Control* | Digunakan untuk menghubungkan boundary dengan tabel. |
| 7 |  | *LifeLine Entity* | Menjelaskan hubungan antara kegiatan. |

### Class Diagram

Sukamto dan Shalahuddin (2013:141), “*Class* diagram atau diagram kelas menjelaskan tentang struktur sistem dalam hal menggambarkan tingkatan yang dirancang pada sistem”. Memiliki operasi dan atribut.

* 1. Sebuah kelas memiliki variable atribut biasa.
  2. Suatu kelas memiliki fungsi seperti operasi dan metode.

Adapun susunan struktur kelas yang harus dimiliki seperti :

1. Sistem dijalankan ketika ada yang memiliki fungsi awal.
2. Berfungsi untuk menampilkan kelas yang menjelaskan dan membuat tampilan terhadap user.
3. Menjelaskan *use case (controller)* yang menangani sebuah fungsi yang harus diambil dari penjelasan sebuah *use case*, kelas ini disebut juga sebagai kelas yang melayani proses aktivitas untuk perangkat lunak.
4. Penjelasan data (model) yang digunakan sebagai menyatukan data menjadi kesatuan yang disimpan kedalam data.

**Tabel 5.** Simbol Class Diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Gambar** | **Nama** | **Keterangan** |
| 1 |  | *Nary Association* | Fungsi untuk menghindari asosiasi lebih dari 2 objek. |
| 2 |  | *Class* | Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama. |