

BAB 2

LANDASAN TEORI

2.1 *E-Learning* (Pembelajaran Daring)

E-Learning adalah penggunaan teknologi informasi dan komunikasi yang terhubung secara online dalam belajar dan mengajar [10]. E-Learning memanfaatkan teknologi internet atau koneksi internet untuk memberikan materi, tugas, dan penilaian secara online. E-Learning adalah pembelajaran berbasis teknologi yang memiliki fleksibilitas dan aksesibilitas dalam pembelajaran karena siswa dapat belajar tanpa batasan waktu dan jarak dan potensinya dapat meningkatkan kualitas Pendidikan [11]. E-Learning digunakan sebagai platform utama untuk mengerjakan tugas dan pemberian umpan balik melalui ITS dan Bayesian Network.

2.1.1 Desain Sistem ITS pada E-Learning

Tujuan utama sistem ini adalah untuk mengidentifikasi kelemahan mahasiswa dan memberikan umpan balik (*feedback*) serta tugas tambahan untuk meningkatkan pemahaman dan nilai akademik mahasiswa.

Pemilihan jumlah 10 soal pada setiap tugas didasarkan pada prinsip pengukuran kemampuan yang optimal tanpa membebani memori kerja siswa. *Standard Error of Measurement (SEM)* menunjukkan bahwa jumlah soal ideal berada di kisaran 6 hingga 12 untuk mengukur kemampuan secara akurat [12]. Jumlah 10 soal dipilih karena berada di tengah kisaran tersebut dan dianggap ideal. Selain itu, terlalu banyak soal dapat meningkatkan beban kognitif dan menurunkan efektivitas pembelajaran [13].

Pembagian soal ke dalam tiga tingkat kesulitan merupakan pendekatan efektif dalam asesmen. Pengelompokan soal berdasarkan kesulitan dapat membedakan tingkat performa mahasiswa dengan lebih akurat [14]. Komposisi 4 soal mudah, 3 sedang, dan 3 sulit dipilih karena:

- Soal mudah berfungsi sebagai penguat dasar pemahaman mahasiswa,
- Soal sedang dan sulit berfungsi sebagai alat ukur diferensiasi kemampuan siswa,

- Desain ini sejalan dengan panduan penyusunan soal dari para ahli asesmen [15].

Tugas Tambahan akan diberikan dalam dua kondisi:

1. Jika ada satu atau lebih tugas yang tidak lulus, maka tingkat kesulitan soal ditentukan dari analisis kesalahan pada tugas yang gagal.
2. Jika semua tugas lulus tetapi nilai rata-rata di bawah A (85), maka analisis dilakukan berdasarkan seluruh tugas tersebut.

Sistem E-Learning ini dirancang untuk:

- Memberikan empat (4) tugas (*task*) pada setiap mata kuliah selama satu semester.
- Setiap tugas terdiri dari 10 soal pilihan ganda dengan komposisi: 4 soal mudah (*easy*), 3 soal sedang (*medium*), 3 soal sulit (*hard*).

Setelah keempat tugas diselesaikan, ITS akan secara otomatis:

- Menganalisis hasil jawaban mahasiswa.
- Menghitung rata-rata nilai dan mendeteksi tugas yang tidak lulus (nilai di bawah C atau 55).
- Memberikan **umpan balik otomatis** mengenai performa mahasiswa.
- Jika terdapat satu atau lebih tugas yang tidak lulus, maka sistem akan memberikan **Tugas Tambahan** secara otomatis.

Tugas tambahan ini terdiri dari 20 soal yang diambil dari *Bank Soal*. Distribusi tingkat kesulitan (mudah, sedang, sulit) akan dihitung otomatis menggunakan *Model Bayesian Network*. Penyesuaian ini dilakukan berdasarkan kesalahan mahasiswa pada tugas sebelumnya.

Pendekatan ini didukung oleh teori *deliberate practice*, yaitu metode pembelajaran melalui latihan yang terarah, dengan umpan balik, serta pengulangan yang cukup [16]. Karena jumlah soal awal adalah 10, maka jumlah soal pada tugas tambahan digandakan menjadi 20 sebagai bentuk pengulangan yang memadai untuk perbaikan performa. Selain itu, pemberian bantuan melalui penyesuaian kesulitan soal juga terbukti efektif untuk membantu siswa dalam mencapai tujuan pembelajaran [17].

2.2 Bayesian Network

Bayesian Network adalah model grafis probabilitas berbasis *Teorema Bayes*, yang digunakan untuk memodelkan ketidakpastian dalam hubungan antar variable [7]. Pada sistem ini digunakan Bayesian Network Prior dan Posterior untuk menghitung soal yang akan diberikan menurut kinerja mahasiswa.

2.2.1 Probabilitas Prior

Dalam konteks Bayesian Network, pendekatan ini digunakan untuk memodelkan ketidakpastian menggunakan probabilitas, sebagaimana dijelaskan dalam Teorema Bayes. Dalam Intelligent Tutoring System (ITS), *prior probability* digunakan untuk menetapkan probabilitas awal tingkat kesulitan soal sebelum mempertimbangkan performa siswa, misalnya, untuk tugas yang terdiri dari 10 soal dengan distribusi 4 soal mudah, 3 soal sedang, dan 3 soal sulit.

$$P(B = b) \quad (2.1)$$

Prior probability $P(B = b)$ adalah probabilitas awal bahwa sebuah soal memiliki tingkat kesulitan b , seperti yang digunakan dalam Bayesian Network untuk merepresentasikan probabilitas variabel sebelum mempertimbangkan bukti baru. Dalam ITS, ini mencerminkan distribusi awal tingkat kesulitan soal [7].

- B : Variabel acak yang mewakili tingkat kesulitan soal dalam ITS, dengan nilai seperti *easy*, *medium*, dan *hard*.
- b : Nilai spesifik dari variabel B , misalnya *easy*, *medium*, atau *hard*.

Dalam sistem *Intelligent Tutoring System* (ITS), *Prior Probabilitas* adalah probabilitas awal yang digunakan untuk menghitung kemungkinan sebuah soal memiliki tingkat kesulitan tertentu sebelum mempertimbangkan data tambahan, seperti kesalahan siswa. Rumus 2.1 menunjukkan perhitungan *Prior* berdasarkan distribusi soal dalam tugas.

2.2.2 Error Rate dan Success Rate

Error Rate adalah proses yang digunakan untuk mengukur total kesalahan yang dilakukan oleh mahasiswa pada tugas-tugas yang tidak lulus. Proses ini

menghitung tingkat ketidakberhasilan dalam menjawab soal untuk tingkat kesulitan tertentu, memberikan gambaran tentang tantangan yang dihadapi mahasiswa dalam memahami materi [18]. *Success Rate* adalah kebalikan dari *error rate* sehingga untuk mencari *success rate* hanya memerlukan rumus kebalikannya dengan *error rate* atau 1 dikurang *error rate*. Kedua proses ini penting untuk mengevaluasi efektivitas pembelajaran dalam sistem ITS dan membantu mengidentifikasi kemampuan mahasiswa untuk tingkat kesulitan dari tugas tambahan.

Rumus 2.2 menunjukkan perhitungan *Tingkat Kesalahan*, yaitu rasio jumlah jawaban salah terhadap total soal untuk tingkat kesulitan $D = d$ pada tugas yang tidak lulus. Variabel e_d menyatakan jumlah soal yang dijawab salah (*error*) untuk tingkat kesulitan d , sedangkan q_d adalah jumlah total soal untuk tingkat kesulitan tersebut.

$$\text{Tingkat Kesalahan}(D = d) = \frac{e_d}{q_d} \quad (2.2)$$

Tingkat Keberhasilan dihitung sebagai proporsi jawaban benar untuk tingkat kesulitan $D = d$ dalam sistem *Intelligent Tutoring System* (ITS), sebagaimana ditunjukkan pada Rumus 2.3. Rumus ini dihitung dengan mengurangi *Tingkat Kesalahan* dari 1, sehingga menghasilkan *Tingkat Keberhasilan*.

$$\text{Tingkat Keberhasilan}(D = d) = 1 - \frac{e_d}{q_d} \quad (2.3)$$

2.2.3 Probabilitas Posterior

Dalam konteks *Intelligent Tutoring System* (ITS), *posterior probability* adalah probabilitas yang diperbarui untuk tingkat kesulitan soal setelah mempertimbangkan bukti baru, seperti kesalahan siswa pada Tugas 1 hingga 4 dengan nilai di bawah 55. *Bayesian Networks* menggunakan probabilitas posterior untuk menentukan distribusi tingkat kesulitan soal dalam Tugas Tambahan (remedial) yang berisi 20 soal, berdasarkan kesalahan dari tugas yang tidak lulus.

$$P(B | A) = \frac{P(A | B) \cdot P(B)}{P(A)} \quad (2.4)$$

Posterior probability $P(B | A)$ adalah probabilitas bahwa sebuah soal memiliki tingkat kesulitan tertentu, setelah mempertimbangkan bukti seperti kesalahan siswa pada tugas sebelumnya [7]. Dalam ITS, probabilitas ini digunakan

untuk menyesuaikan distribusi soal pada Tugas Tambahan.

- B : Variabel acak yang mewakili tingkat kesulitan soal dalam ITS, seperti *easy*, *medium*, atau *hard*.
- A : Variabel acak yang mewakili bukti, seperti kesalahan siswa pada tugas yang tidak lulus.
- $P(A | B)$: Probabilitas bukti A terjadi jika tingkat kesulitan soal adalah B .
- $P(B)$: Prior probability, yaitu probabilitas awal tingkat kesulitan soal.
- $P(A)$: Probabilitas total bukti A , sebagai faktor normalisasi.

Sebagai contoh, dalam tugas dengan 10 soal (4 mudah, 3 sedang, 3 sulit), prior probability adalah $P(B = \text{easy}) = 0.4$, $P(B = \text{medium}) = 0.3$, dan $P(B = \text{hard}) = 0.3$. Probabilitas posterior $P(B | A)$ menyesuaikan distribusi ini berdasarkan kesalahan siswa untuk Tugas Tambahan.

2.2.4 Distribusi Soal Tugas Tambahan

Setelah memperoleh nilai *Posterior Probabilitas* untuk setiap tingkat kesulitan soal, langkah selanjutnya dalam sistem *Intelligent Tutoring System* (ITS) adalah menghitung proporsi atau jumlah tingkatan kesulitan soal yang akan digunakan dalam Tugas Tambahan. Jumlah soal ditentukan dengan cara menormalisasi nilai *Posterior* dan mengalikannya dengan total soal, yaitu 20 soal. Dengan demikian, distribusi soal untuk tingkat kesulitan *easy*, *medium*, dan *hard* dapat diperoleh langsung dari hasil perkalian antara nilai posterior dan jumlah total soal.

2.2.5 Bobot Nilai

Dalam sistem *Intelligent Tutoring System* (ITS), setiap soal pada Tugas Tambahan memiliki bobot nilai yang berbeda berdasarkan tingkat kesulitannya. Bobot nilai untuk soal *easy*, *medium*, dan *hard* masing-masing adalah 5, 10, dan 15. Total skor maksimal dari 20 soal harus tetap 100. Oleh karena itu, dilakukan normalisasi bobot agar pembobotan soal tetap adil dan total nilainya tetap valid. Rumus 2.5 menunjukkan proses perhitungannya [19].

$$\text{Weight}(D) = \frac{100 \times \text{Default Weight}(D) \times n_D}{\text{Total Default}} \quad (2.5)$$

$\text{Weight}(D)$ adalah bobot nilai aktual dari tingkat kesulitan D dalam sistem, setelah dinormalisasi terhadap total 100. Rumus ini memastikan bahwa total nilai seluruh soal tetap 100 meskipun jumlah soal pada tiap tingkat kesulitan berbeda.

- n_D adalah jumlah soal untuk tingkat kesulitan D , berdasarkan hasil distribusi posterior.
- $\text{Default Weight}(D)$ adalah bobot dasar per soal sesuai kesulitan: 5 untuk *easy*, 10 untuk *medium*, dan 15 untuk *hard*.
- Total Default adalah jumlah skor awal dari jawaban yang salah.
- Validasi akhir: jumlah total bobot dari semua soal jika dijawab benar harus = 100.

2.3 Laravel

Laravel merupakan *framework* PHP *open-source* yang dirancang untuk mempercepat dan mempermudah proses pengembangan aplikasi web. Dengan sintaks yang elegan serta fitur-fitur seperti manajemen basis data, autentikasi, *routing*, dan pembuatan antarmuka pengguna yang intuitif, Laravel memungkinkan pengembang membangun aplikasi yang stabil, aman, dan skalabel secara efisien, termasuk mendukung fitur seperti *dashboard* untuk visualisasi data [20].

Laravel juga dilengkapi dengan *Artisan* untuk otomatisasi tugas, dukungan protokol HTTPS, pembuatan *RESTful API*, serta penggunaan *Composer* untuk manajemen pustaka (*library*), menjadikannya pilihan ideal dalam pengembangan aplikasi berbasis web. Fungsi utama Laravel adalah menyederhanakan dan mempercepat proses pembangunan aplikasi web melalui struktur kode yang rapi, modular, dan mudah dipelihara [21].

Dengan menggunakan Laravel, waktu pengerjaan menjadi lebih singkat karena pengaksesan basis data dapat dilakukan langsung melalui *controller*, yang secara otomatis dapat terhubung dengan bagian *frontend*. Laravel sangat cocok digunakan baik untuk proyek skala kecil maupun besar karena memiliki

dokumentasi yang lengkap serta dukungan komunitas yang luas yang memudahkan proses pengembangan.

2.4 Black Box

Pengujian *black box* merupakan salah satu metode pengujian perangkat lunak yang esensial untuk mengevaluasi fungsionalitas sistem tanpa perlu mengetahui struktur internal maupun kode programnya. Metode ini juga dikenal sebagai *pengujian fungsional* atau *pengujian perilaku*. Pengujian dilakukan dari sudut pandang pengguna akhir (*end-user*) dengan fokus pada spesifikasi kebutuhan untuk memastikan bahwa setiap fungsi, masukan, dan keluaran dari perangkat lunak telah sesuai dengan yang diharapkan [22].

2.4.1 Pengujian Sistem Menggunakan Metode Black Box

Dalam website *e-learning* berbasis ITS, akan digunakan metode pengujian *black box* untuk mengevaluasi berbagai fitur sistem. Pengujian ini merupakan bagian yang sangat penting untuk memastikan bahwa fitur yang dikembangkan telah memenuhi tujuan awal. Pengujian *black box* dipilih karena dilakukan tanpa mengetahui struktur kode internal, melainkan berfokus pada fungsionalitas eksternal dari perspektif mahasiswa [23].

Adapun fitur-fitur utama yang diuji menggunakan metode *black box* antara lain:

1. **Manajemen User** Pengujian pada proses registrasi, login, dan pemilihan jurusan, untuk memastikan mahasiswa dapat mengakses mata kuliah sesuai dengan program studi yang dipilih.
2. **Alur Pengerjaan Tugas** Meliputi pengujian tombol (*button*), validasi error, dan konfirmasi saat mahasiswa mengerjakan tugas agar sistem memberikan respons yang sesuai dan jelas.
3. **Evaluasi Sistem ITS** Fokus pada pengujian fitur utama sistem, yaitu kemampuan untuk menghasilkan tugas tambahan secara otomatis berdasarkan kesalahan mahasiswa pada tugas sebelumnya.

2.5 End-User Computing Satisfaction (EUCS)

End-User Computing Satisfaction (EUCS) merupakan model yang digunakan untuk mengukur tingkat kepuasan pengguna terhadap sistem komputer, termasuk perangkat lunak dan sistem informasi. Model ini pertama kali diperkenalkan oleh Doll dan Torkzadeh pada tahun 1988 sebagai alat untuk menilai efektivitas sistem dari sudut pandang pengguna.

Model *EUCS* terdiri dari lima dimensi utama, yaitu [24]:

- **Konten (Content):** seberapa relevan dan lengkap informasi yang diberikan sistem.
- **Ketepatan (Accuracy):** seberapa akurat data dan hasil keluaran dari sistem.
- **Format (Format):** kualitas tampilan, keterbacaan, dan penyajian informasi.
- **Kemudahan Penggunaan (Ease of Use):** seberapa mudah pengguna mengoperasikan sistem tanpa pelatihan rumit.
- **Ketepatan Waktu (Timeliness):** kecepatan sistem dalam menampilkan informasi yang dibutuhkan.

2.5.1 Penggunaan Skala Likert

Pengukuran EUCS dilakukan menggunakan Skala Likert 5 poin, yang terdiri dari:

Tabel 2.1. Skala Likert 5 Poin

Nilai	Tingkat Kepuasan
1	Sangat Tidak Setuju
2	Tidak Setuju
3	Netral
4	Setuju
5	Sangat Setuju

Skala ini digunakan untuk mengubah pendapat pengguna menjadi angka agar bisa dianalisis secara kuantitatif. Setiap pernyataan dalam kuesioner EUCS

dinilai berdasarkan skala tersebut. Nilai-nilai ini kemudian dijumlahkan atau dirata-ratakan untuk tiap dimensi dan keseluruhan sistem. Hasil akhirnya digunakan untuk menilai tingkat kepuasan pengguna.

Skala 5 poin dipilih karena mudah digunakan dan cukup untuk menilai pendapat dalam studi awal. Namun, peneliti perlu memastikan pernyataan dalam kuesioner jelas, agar data yang diperoleh akurat dan mewakili persepsi pengguna secara benar [25].

