

## BAB 2 LANDASAN TEORI

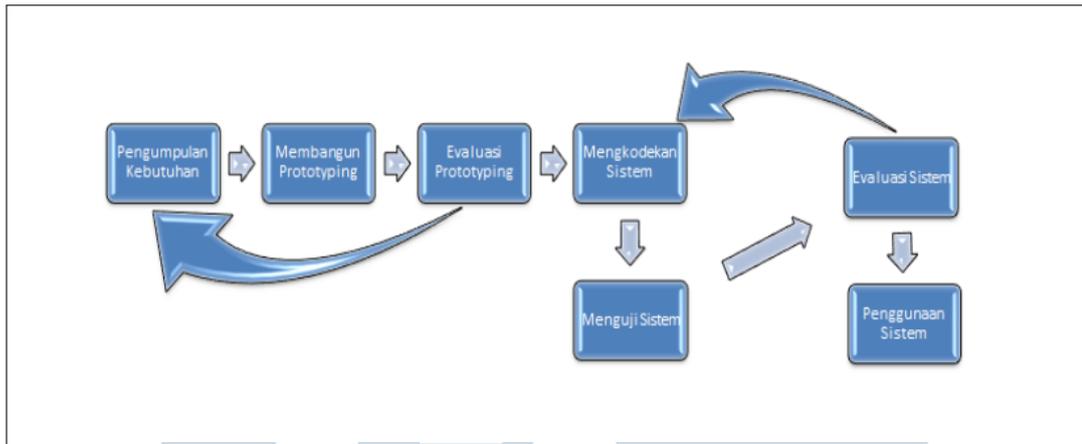
### 2.1 Metode Prototyping

*Prototyping* adalah proses pengembangan sistem yang menggunakan prototipe sebagai dasar pembuatan *software*. Metode ini digunakan untuk memecahkan masalah kesalahpahaman antara pengguna dan pengembang dalam berkomunikasi dalam menentukan *requirement* yang dibutuhkan oleh *software* yang ingin dikembangkan [12], *prototyping* juga digunakan untuk memberikan gambar mengenai aplikasi yang akan dibangun dan juga menjadi representasi dari model aplikasi yang akan dibuat [13].

Metode *Prototyping* digunakan pada penelitian ini untuk membangun *website e-commerce* dengan cepat dan teratur. Metode *prototyping* merupakan metode pengembangan perangkat lunak yang dilakukan secara iteratif dan menghasilkan *prototype* sistem sesuai dengan kebutuhan *user*. Jika terdapat masalah atau ketidaksesuaian dengan *user* pada tahap evaluasi *prototyping* atau evaluasi sistem, maka akan kembali ke tahap sebelumnya untuk menyelesaikan masalah atau ketidakcocokan tersebut [7]. Dengan menerapkan *Software Development Life Cycle (SDLC)* dalam pembuatan aplikasi bertujuan untuk mengembangkan model menjadi perangkat lunak yang terakhir. Artinya sistem akan dikembangkan lebih cepat dan biaya yang dikeluarkan lebih rendah.

Kelebihan metode *prototyping* [14] [11], antara lain :

1. Adanya komunikasi yang baik dalam menentukan *requirement* yang dibutuhkan oleh aplikasi.
2. Implementasi menjadi lebih mudah karena mendapat gambaran awal dari aplikasi melalui *prototype*.
3. Pengguna dapat melihat dan merasakan langsung kejelasan fungsi dan aplikasi jika terdapat perubahan dan modifikasi.
4. Mengurangi resiko kegagalan karena potensi kesalahan bisa dikenali ditahap awal dan dapat mengambil langkah awal untuk mencegah kegagalan.



Gambar 2.1. Model *Prototyping*

*Prototyping* memiliki beberapa tahapan[15], antara lain :

1. Tahap Pengumpulan Kebutuhan  
 Pada tahap ini dilakukan pengumpulan data kepada pihak PT<sub>i</sub> Bumi Baureksa Pratama yaitu Bapak Andry melalui wawancara untuk mengetahui kebutuhan sistem, alur dari sistem, dan fitur yang akan dibuat.
2. Tahap Prototyping  
 Tahap ini membuat *wireframe* dan *mockup* menggunakan bantuan aplikasi figma untuk membuat gambaran *website* secara kasar.
3. Tahap Evaluasi Prototyping  
 Pada tahap ini dilakukan perwakilan perusahaan akan melakukan pengecekan terhadap *prototype* yang dibangun dengan tujuan untuk memeriksa dan memastikan bahwa sistem yang dibangun sudah memenuhi kebutuhan dan persyaratan. Apabila belum sesuai kebutuhan maka akan dilakukan perbaikan dan kembali ke tahap sebelumnya.
4. Tahap Mengkodekan Sistem  
*Prototype* yang telah diuji dan disetujui akan diubah kedalam bentuk kode dengan menggunakan bahasa pemrograman.
5. Tahap Pengujian Sistem  
 Pada tahap ini dilakukan pengujian sistem terhadap *website*, *website* akan diuji menggunakan *User Acceptance Test (UAT)* untuk mengecek apakah setiap fitur didalam *website* sudah berjalan sesuai harapan.

## 6. Tahap Evaluasi Sistem

Pada tahap ini dilakukan evaluasi terhadap *website* yang sudah dibangun kepada klien untuk memastikan kembali apakah *website* yang dibangun telah memenuhi kebutuhan menggunakan metode *USE Questionnaire*. Jika sudah maka *website* dapat digunakan, tetapi jika tidak memenuhi kebutuhan maka harus kembali ke langkah sebelumnya untuk memperbaiki ketidaksesuaian itu.

## 7. Tahap Menggunakan Sistem

Sistem yang telah dibangun dan berhasil melewati tahap pengujian sistem dan evaluasi sudah dapat digunakan.

### 2.2 User Acceptance Testing

*User Acceptance Testing (UAT)* digunakan sebagai metode pengujian dalam pembuatan aplikasi *e-commerce* berbasis *website*. *UAT* adalah sebuah pengujian yang melibatkan pengguna akhir untuk dapat berinteraksi langsung menggunakan sistem dan juga memverifikasi terkait fungsionalitas dari aplikasi tersebut dan pengujian ini dilakukan untuk memastikan apakah sistem yang dibangun sudah sesuai dengan kebutuhan [16].

Terdapat beberapa manfaat dan tujuan dari penggunaan *UAT*, yaitu :

#### A Manfaat

1. Untuk meningkatkan kepercayaan klien mengenai persyaratan perangkat lunak yang sudah terpenuhi.
2. Mendapatkan sistem yang sesuai dengan spesifikasi fungsional sistem.
3. Kepuasan klien meningkat, karena mereka lebih yakin bahwa sistem sudah memenuhi persyaratan.
4. Mendapatkan sistem yang sesuai dengan persyaratan dan fungsional sistem.

#### B Tujuan

1. Menguji kesesuaian sistem berdasarkan persyaratan dan fungsional sistem.

2. Memberikan keyakinan terhadap klien bahwa sistem telah memenuhi persyaratan dan berjalan dengan baik

### 2.3 E-commerce

*E-commerce* adalah distribusi, penjualan, pembelian dan pemasaran produk menggunakan jaringan komunikasi. Arti kata *e-commerce* sendiri dapat didefinisikan sebagai penggunaan pemrosesan informasi digital dalam melakukan transaksi untuk membuat, mengubah dan mendefinisikan kembali hubungan antara penjual dan pembeli [17].

*E-commerce* dapat diterapkan oleh suatu perusahaan, bisnis, dan usaha Mikro, Kecil dan Menengah (UMKM) untuk memberikan dampak pada akselerasi perkembangan dunia usaha baik usaha skala kecil, menengah maupun kelas atas. Dampak positif ini tentu akan dirasakan apabila perusahaan dapat menggunakan *e-commerce* dengan tepat dan disesuaikan dengan jenis dan karakter usahanya. Salah satu fungsi dari pemanfaatan *e-commerce* ini adalah adanya efisiensi terhadap dunia usaha. Baik efisien secara materil (biaya) maupun secara non-materil (tenaga dan waktu). Dari segi biaya, perusahaan dapat menekan biaya misalnya dengan memanfaatkan telepon dan internet sebagai media penawaran dan promosi barang atau jasa. Karena hal tersebut akan lebih murah dibandingkan dengan cara tradisional atau *offline*. Di sisi lain, efisiensi biaya ini juga bisa terjadi karena adanya pengurangan tenaga kerja pada posisi tertentu[2].

### 2.4 Insertion Sort Algorithm

*Insertion Sort Algorithm* adalah algoritma penyortiran berbasis perbandingan algoritma yang sederhana, algoritma ini membandingkan dua elemen pertama di dalam sebuah *array*, jika elemen pertama lebih besar dibandingkan elemen kedua, maka elemen pertama akan ditukar oleh elemen kedua. Proses ini akan terus berlangsung sampai semua elemen di dalam *array* sudah berurutan. Dalam membandingkan algoritma *selection sort* dengan *insertion sort*, kedua algoritma memiliki keuntungan yang berbeda dalam situasi yang berbeda. Algoritma *insertion sort* dalam mengurutkan array mengungguli *selection sort* dalam waktu eksekusi[18].

Kompleksitas waktu pengurutan *best case* untuk *insertion sort* adalah  $O(n)$  sedangkan untuk *average* dan *worst case* adalah  $O(n^2)$ [10]. Penerapan

algoritma *insertion sorting* dilakukan terhadap produk di *website e-commerce* untuk menyortir produk berdasarkan harga, waktu, dan peringkat secara *ascending* ataupun *descending*, karena harga adalah hal pertama yang dilihat pelanggan sebelum mereka membeli sesuatu. Jadi, menyortir adalah fitur terpenting yang dibutuhkan setiap orang ketika kita ingin berbelanja[8].

## 2.5 Usefulness, Satisfaction, and Ease of Use (USE) Questionnaire

*USE Questionnaire* merupakan suatu paket kuisioner yang diusulkan oleh Lund digunakan untuk mengukur *usability* karena berisi 3 standar pengukuran menurut ISO 9241, yaitu efisiensi, efektivitas, dan kepuasan[19]. *USE Questionnaire* menyajikan 30 pertanyaan yang dikelompokkan ke dalam 4 dimensi, yaitu kegunaan (*usefulness*), kemudahan dalam penggunaan (*ease of use*), kemudahan untuk dipelajari (*ease of learning*), dan kepuasan (*satisfaction*)[20]. Daftar pertanyaan yang telah dikelompokkan dan disesuaikan dalam bahasa Indonesia tersebut dijabarkan sebagai berikut[21] :

### 2.5.1 Usefulness

Dibawah ini merupakan daftar pertanyaan dimensi *usefulness*.

1. Sistem ini membantu saya menjadi lebih efektif
2. Sistem ini membantu saya menjadi lebih produktif
3. Sistem ini bermanfaat
4. Sistem ini membantu saya terhadap aktivitas yang saya lakukan
5. Sistem ini membuat hal-hal yang ingin saya capai lebih mudah untuk dilakukan
6. Sistem ini menghemat waktu saya ketika menggunakannya
7. Sistem ini sesuai dengan kebutuhan saya
8. Sistem ini bekerja sesuai apa yang saya harapkan

### **2.5.2 Ease of Use**

Dibawah ini merupakan daftar pertanyaan dimensi *Ease of Use*.

1. Sistem ini mudah digunakan
2. Sistem ini praktis untuk digunakan
3. Sistem ini mudah dipahami oleh pengguna
4. Langkah-langkah pengoperasian sistem ini praktis
5. Sistem ini dapat disesuaikan dengan kebutuhan
6. Saya merasa tidak kesulitan menggunakan sistem ini
7. Saya dapat menggunakannya tanpa instruksi tertulis
8. Saya tidak melihat adanya ketidak konsistenan selama menggunakan sistem ini
9. Pengguna yang jarang maupun rutin menggunakan akan menyukai sistem ini
10. Saya dapat kembali dari kesalahan dengan cepat dan mudah
11. Saya dapat menggunakan sistem ini dengan berhasil setiap kali saya menggunakannya

### **2.5.3 Ease of Learning**

Dibawah ini merupakan daftar pertanyaan dimensi *Ease of Learning*.

1. Saya belajar menggunakan sistem ini dengan cepat
2. Saya mudah mengingat bagaimana cara menggunakan sistem ini
3. Sistem ini mudah untuk dipelajari cara penggunaannya
4. Saya cepat menjadi terampil dengan sistem ini

#### 2.5.4 Satisfaction

Dibawah ini merupakan daftar pertanyaan dimensi *satisfaction*.

1. Saya puas dengan sistem ini
2. Saya akan merekomendasikan sistem ini kepada rekan
3. Sistem ini menyenangkan untuk digunakan
4. Sistem ini bekerja seperti yang saya inginkan
5. Sistem ini sangat bagus
6. Saya merasa harus memiliki / menggunakan sistem ini
7. Sistem ini nyaman untuk digunakan

#### 2.6 Skala Likert

*Skala likert* adalah skala pengukuran yang digunakan untuk mengukur sifat, sikap dan perilaku untuk membentuk atau menghasilkan sebuah nilai. *Skala likert* merupakan skala psikometrik yang paling banyak digunakan dalam penelitian [22]. Berikut merupakan tabel dari *skalar likert*

Tabel 2.1. Tabel Skala Likert

No	Kriteria	Skor
1	Sangat Setuju (SS)	5
2	Setuju (S)	4
3	Ragu-Ragu (R)	3
4	Tidak Setuju (TS)	2
5	Sangat Tidak Setuju (STS)	1

##### 2.6.1 Mencari Total Skor

Untuk mengetahui skor total, kalikan setiap skor untuk setiap kategori dengan jumlah total responden yang memilih kategori tersebut. Perhitungan untuk menemukan skor total ditunjukkan dalam rumus :

$$TotalSkor = (1 * STS) + (2 * TS) + (3 * KS) + (4 * S) + (5 * SS) \quad (2.1)$$

Keterangan:

- STS = jumlah sangat tidak setuju
- TS = jumlah tidak setuju
- KS = jumlah kurang setuju
- S = jumlah setuju
- SS = jumlah sangat setuju

### 2.6.2 Mencari Skor Maksimum

Skor maksimum diperoleh dengan mengalikan jumlah responden, skor *likert* maksimum dan juga jumlah pertanyaan yang diajukan kepada responden. Skor maksimum dapat ditemukan dengan perhitungan berikut:

$$SkorMaks = JumlahResponden * SkorMaksLikert * JumlahPertanyaan \quad (2.2)$$

### 2.6.3 Mencari Nilai Indeks (%)

Nilai indeks diperoleh dari hasil bagi skor total dan juga skor maksimum kemudian dikalikan 100. Nilai indeks dapat dicari dengan perhitungan sebagai berikut:

$$Indeks = \frac{TotalSkor}{SkorMaks} * 100\% \quad (2.3)$$

### 2.6.4 Menentukan Interval Penilaian

Penentuan interval penilaian ditentukan seperti yang dijabarkan pada Tabel 2.2

Tabel 2.2. Interval Penilaian

<b>Indeks</b>	<b>Kategori</b>
0% - 19,99%	Sangat Buruk
20% - 39,99%	Buruk
40% - 59,99%	Kurang Baik
60% - 79,99%	Baik
80% - 100%	Sangat Baik

