

## **BAB II**

### **TINJAUAN PUSTAKA DAN LANDASAN TEORI**

#### **2.1 Tinjauan Pustaka**

##### **2.1.1 E-Marketplace**

*E-marketplace* merupakan bagian dari *e-commerce*. Menurut Brunn, Jensen, & Skovgaard (2002), *e-marketplace* adalah wadah komunitas bisnis interaktif secara elektronik yang menyediakan pasar dimana perusahaan dapat ambil andil dalam B2B *e-commerce* dan atau kegiatan *e-business* lain[3]. *E-marketplace* dapat dikatakan sebagai gelombang kedua pada *e-commerce* dan memperluas kombinasi dari bisnis konsumen (B2B, C2B dan C2C) ke dalam B2B. Inti penawaran dari *e-marketplace* adalah mempertemukan pembeli dan penjual sesuai dengan kebutuhan dan menawarkan efisiensi dalam bertransaksi. Menurut Brunn, Jensen, & Skovgaard (2002) terdapat dua jenis *e-marketplaces* :

Penelitian yang sudah dilakukan antara lain oleh Iin Sofiani dengan judul penelitian Rancang Bangun Aplikasi E-Marketplace Hasil Pertanian Berbasis Website Dengan Menggunakan Framework Codeigniter, dimana dalam penelitiannya ia merancang aplikasi E-Marketplace Hasil Pertanian Berbasis Website Dengan Menggunakan Framework Codeigniter(2021)[4].

Penelitian tentang *emarketplace* juga dilakukan oleh Angga Kurnia Putra dkk, dengan judul penelitian Rancang Bangun Aplikasi Marketplace Penyedia Jasa Les Private di Kota Pontianak Berbasis Web, Angga membantun sistem ini karena melihat permasalahan yaitu pencarian jasa les privat masih konvensional yaitu mencari via kontak sehingga membutuhkan waktu yang lama(2017)[5].

Kemudian penelitian oleh Muhammad Ridwan dengan judul penelitian Rancang Bangun Marketplace Berbasis Website menggunakan Metodologi Systems Development Life Cycle (SDLC) dengan Model Waterfall. Dalam penelitiannya M Ridwan menyatakan bahwa *emarketplace* dapat meningkatkan potensi dari bisnis online di Indonesia. Sistem ini dibangun menggunakan metode SDLC(2019)[6].

Penelitian yang senada juga dilakukan oleh Erik Laurin dengan judul penelitian Digital platform growth under contextual limitations. A case study of growing a complementary digital marketplace. Dalam penelitiannya Erik menyatakan bahwa marketplace berkembang dengan

pesat sehingga pemilik bisnis perlu ekstra hati hati dengan tantangan dan risiko dalam membuat keputusan untuk marketplace bisnisnya (2021)[7].

Kemudian penelitian oleh Jiasi Weng dengan judul penelitian Golden Grain: Building a Secure and Decentralized Model Marketplace for MlaaS. Dalam penelitiannya Jiasi menyatakan bahwa Machine Learning saat ini dibutuhkan dalam bisnis marketplace untuk meningkatkan pelayanan kepada pelanggan(2021)[8].

Lalu penelitian oleh Vlasis Koutsos dengan judul penelitian Agora: A Privacy-Aware Data Marketplace. Vlasis mengatakan bahwa dalam marketplace privasi sangat penting, maka perlu dibangun marketplace berbasis blockchain sehingga privasi dari pelanggan benar benar terjaga (2021)[9].

#### 1. *E-marketplaces* horizontal

*E-marketplaces* horizontal dikategorikan berdasarkan fungsi atau produk umum yang ditawarkan perusahaan. Dapat diartikan pasar yang digunakan untuk industri umum. Seperti pasar penjualan *smartphone*, *pc*, baju. Biaya transaksi yang dikeluarkan lebih rendah.

#### 2. *E-marketplaces* vertical

*E-marketplaces* vertikal dapat diartikan pasar yang digunakan untuk industri yang memenuhi kebutuhan khusus pada masing-masing industri. Seperti pasar penjualan beton, baja.

Dalam pengaplikasian *e-marketplace* dibutuhkan strategi untuk mengoptimalkan *e-marketplace*. Terdapat tiga bagian utama yaitu pengaturan, tantangan dan tujuan.

##### 1. Pengaturan

Apa saja yang menjadi pondasi suksesnya *e-marketplace* yaitu *focus*, pemerintahan, fungsi, teknologi dan kerjasama. *Focus* pada bisnis tentu perlu, perusahaan harus memiliki konsep kuat dan *focus* akan target yang ingin dicapai. Pemerintahan dalam arti membangun perusahaan memerlukan adaptasi dengan peraturan-peraturan yang dibuat oleh pemerintah setempat. Fungsionalitas diartikan sebagai produk atau jasa yang ditawarkan memiliki fungsi yang tepat dan sesuai kebutuhan pasar.

Untuk teknologi, teknologi terus berkembang, perusahaan harus dapat beradaptasi dengan teknologi untuk terus bertahan. Kerjasama adalah salah satu faktor penting untuk membangun

*e-marketplace* yang sukses, karena dengan menjalin kerjasama yang baik dengan berbagai pihak apalagi pihak utama perusahaan maka diharapkan proses bisnis perusahaan lebih lancar. Semua pengaturan menjadi pondasi yang harus kuat bagi perusahaan. Meskipun pengaturan ini bersifat dinamis karena disesuaikan dengan lingkungan perusahaan.

## 2. Tantangan

Perusahaan diharapkan untuk membangun likuiditas dan menangkap nilai sebagai tantangan. Kedua hal tersebut saling berhubungan erat satu sama lain. Untuk itu diperlukan pemikiran yang tepat untuk menjaga kedua hal tersebut agar perusahaan dapat terus bertahan hingga tercapai sukses.

## 3. Tujuan

Pengaturan dan tantangan harus didiskusikan dengan baik karena sebagian *e-marketplace* masih dalam tahap awal. Perlu dipikirkan isu-isu yang berkaitan dengan *e-marketplace* sehingga dapat ditemukan solusi dan tujuan *e-marketplace* yang sukses dapat tercapai.

### 2.1.2 Make UP Artist (MUA)

*Makeup artist* yaitu seseorang yang biasanya memoles wajah agar tampak lebih cantik, megoreksi wajah, dengan menonjolkan kelebihan dan menutupi kekurangan misalnya mengoreksi tulang pipi yang kurang menonjol melalui teknik *shading*, *tin*, dan *highlight*, serta *makeup artist* memahami dan menerapkan prinsip higienis terhadap kosmetik yang dipakai dan perangkat .

Tata rias wajah atau yang biasa dikenal dengan sebutan *makeup* sekarang ini telah menjadi bagian dari rutinitas kehidupan masyarakat *modern* khususnya bagi kaum wanita. Penggunaan tata rias wajah sendiri sudah berkembang menjadi suatu kebutuhan untuk mempercantik diri, menunjukkan jati diri dan kepribadian, serta untuk mengikuti perkembangan mode terutama di dunia *entertainment* di mana berkumpulnya para profesional *makeup artist*. perlu diketahui, bahwa penggunaan tata rias wajah bukan hanya dapat dilakukan oleh seorang profesional *Makeup Artist* (MUA) saja. Wanita biasa pun dapat menggunakan tata rias wajah guna menunjukkan kepedulian terhadap penampilannya, dalam memenuhi aktifitas kehidupannya sehari-hari layaknya seorang profesional *makeup artist* (MUA). Namun

penggunaan tata rias wajah ini pun memiliki tahapan tertentu yang tidak boleh disepelekan. Penggunaan tata rias wajah dengan memperhatikan tahapan yang benar akan membantu mendapatkan hasil kecantikan yang optimal, tanpa merusak kulit wajah penggunanya, karena itu sangatlah penting untuk menggunakan produk dan peralatan kosmetik yang sesuai dengan jenis kulit dan dengan memperhatikan bentuk wajahnya.

Penggunaan tata rias wajah dengan tidak memperhatikan jenis kulit dan bentuk wajah ini akan berdampak negatif bagi kesehatan kulit, seperti munculnya jerawat, kulit mengelupas, bahkan timbulnya komedo. Saat ini khususnya di Indonesia sudah banyak berkembang jasa yang ditawarkan oleh para profesional *makeup artist* untuk memenuhi kebutuhan setiap wanita akan ketergantungannya terhadap tata rias wajah. Tata rias wajah yang mereka tawarkan bervariasi mulai dari penggunaan tata rias wajah untuk keperluan dunia entertainment, pernikahan, acara pesta, pesta kostum, seni, bahkan tata rias wajah untuk sehari-hari. Dan saat ini pun profesional *makeup artist* (MUA) juga banyak menawarkan jasa kursus dalam penggunaan tata rias.

### 2.1.3 Website

*Website* merupakan kumpulan dari beberapa laman yang saling terhubung dan file-filenya saling terkait. *Website* terdiri dari *page* atau halaman, dan kumpulan halaman tersebut dinamakan *homepage*. *Homepage* terletak pada posisi teratas, dengan halaman-halaman terkait yang berada di bawahnya. Biasanya setiap halaman di bawah *homepage* disebut *child page*, dimana berisi *hyperlink* untuk menuju ke halaman lain dalam *web*.

Menurut Suhartanto M, *website* adalah suatu sistem yang berhubungan atau terkait dengan dokumen yang digunakan sebagai media untuk menampilkan teks, gambar, multimedia dan lainnya pada jaringan *internet*[10]. Jadi *website* adalah kumpulan laman yang saling terhubung yang digunakan sebagai media menampilkan teks, gambar dan media lainnya pada jaringan internet.

## 2.2 Alat Pengembangan Sistem

### 2.2.1 UML ( Unified Modelling Language )

UML (*Unified Modeling Language*) adalah ‘bahasa’ pemodelan untuk sistem atau perangkat lunak yang berparadigma ‘berorientasi objek’. Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami. Menurut Nugroho (2009)[11], “UML (*Unified Modeling Language*) adalah Metodologi kolaborasi antara metoda-metoda Booch, OMT (*Object Modeling Technique*), serta OOSE (*object Oriented Software Enggineering*) dan beberapa metoda lainnya, merupakan metodologi yang paling sering digunakan saat ini untuk analisa dan perancangan sistem dengan metodologi berorientasi objek mengadaptasi maraknya penggunaan bahasa “pemrograman berorientasi objek” (OOP)”[12]. Berdasarkan pendapat yang dikemukakan di atas dapat ditarik kesimpulan bahwa *Unified Modelling Language (UML)* adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasikan, menspesifikasikan, membangun dan pendokumentasian dari sebuah sistem pengembangan perangkat lunak berbasis Objek (*Object Oriented programming*).

### 2.2.2 Langkah-Langkah Penggunaan Unified Modeling Language (UML)

Menurut Henderi[13] , langkah-langkah penggunaan *Unified Modeling Language (UML)* sebagai berikut:

1. Buatlah daftar *business process* dari level tertinggi untuk mendefinisikan aktivitas dan proses yang mungkin muncul.
2. Petakan *use case* untuk setiap *business process* untuk mendefinisikan dengan tepat fungsional yang harus disediakan oleh sistem, kemudian perhalus *use casediagram* dan lengkapi dengan *requirement*, *constraints* dan catatan-catatan lain.
3. Buatlah *deployment diagram* secara kasar untuk mendefinisikan arsitektur fisik sistem.
4. Definisikan *requirement* lain *non fungsional*, *security* dan sebagainya yang juga harus disediakan oleh sistem.
5. Berdasarkan *use case diagram*, mulailah membuat *activity diagram*.

6. Definisikan obyek - obyek level atas *package* atau *domain* dan buatlah *sequence* dan/atau *collaboration* untuk tiap alur pekerjaan, jika sebuah *use case* memiliki kemungkinan alur normal dan *error*, buat lagi satu diagram untuk masing-masing alur.
7. Buatlah rancangan *user interface model* yang menyediakan antar muka bagi pengguna untuk menjalankan *skenario use case*.
8. Berdasarkan model-model yang sudah ada, buatlah *class diagram*. Setiap *package* atau *domain* dipecah menjadi *hirarki class* lengkap dengan *atribut* dan metodenya. Akan lebih baik jika untuk setiap *class* dibuat *unit test* untuk menguji *fungsiionalitas class* dan interaksi dengan *class* lain.
9. Setelah *class diagram* dibuat, kita dapat melihat kemungkinan pengelompokkan *class* menjadi komponen-komponen karena itu buatlah *component diagram* pada tahap ini.
10. Perhalus *deployment diagram* yang sudah dibuat. Detilkan kemampuan dan *requirement* piranti lunak, sistem operasi, jaringan dan sebagainya. Petakan komponen ke dalam *node*.
11. Mulailah membangun sistem. Ada dua pendekatan yang tepat digunakan:
  - 1) Pendekatan *use case* dengan meng *assign* setiap *use case* kepada tim pengembang tertentu untuk mengembangkan unit kode yang lengkap dengan *test*.
  - 2) Pendekatan komponen yaitu *meng assign* setiap komponen kepada tim pengembang tertentu.

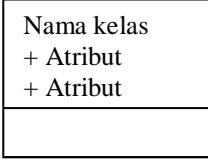


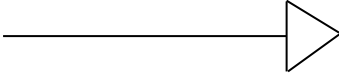

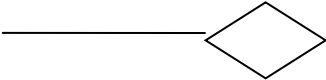
### 2.2.3 Model UML (Unified Modeling Language)

Beberapa *literature* menyebutkan bahwa UML menyediakan sembilan jenis diagram, yang lain menyebutkan delapan karena ada beberapa diagram yang digabung, misalnya diagram komunikasi, diagram urutan dan diagram pewaktuan digabung menjadi diagram interaksi. Namun demikian model-model itu dapat dikelompokkan berdasarkan sifatnya yaitu statis atau dinamis. Jenis diagram itu antara lain:

#### 1. *Class Diagram*

Bersifat statis, Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula diagram kelas memuat kelas-kelas aktif.




**Tabel 2. 1** Simbol-simbol pada *Class Diagram*

Nama Komponen	Keterangan
Kelas 	Kelas pada struktur sistem
Antar muka/ <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi/ <i>association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Generalisasi 	Relasi antar kelas dengan makna generalisasi-specialisasi (umum-khusus)
Kebergantungan/ <i>dependency</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi/ <i>aggregation</i> 	Relasi antar kelas makna semua bagian

## 2. Use Case Diagram

Bersifat statis. Diagram ini memperlihatkan himpunan *use case* dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna. "Sebuah *use case* diagram pada yang paling sederhana adalah representasi dari interaksi pengguna dengan sistem dan menggambarkan spesifikasi dari kasus penggunaan. Sebuah *diagram use case* dapat menggambarkan berbagai jenis pengguna sistem dan berbagai cara yang mereka berinteraksi dengan sistem. Jenis diagram biasanya digunakan dalam hubungannya dengan kasus penggunaan tekstual dan akan sering disertai dengan jenis lain dari diagram juga.

**Tabel 2. 2** Simbol\_simbol pada *Use Case Diagram*



Simbol	Keterangan
<p><i>Use case</i></p> 	<p><i>Use case</i> digunakan sebagai lingkaran elips dengan nama use case dituliskan didalam elips tersebut.</p>
<p>Aktor/<i>actor</i></p> 	<p>Aktor adalah pengguna sistem. Aktor tidak terbatas hanya pada manusia saja, jika sebuah sistem berkomunikasi dengan aplikasi lain dan membutuhkan <i>input</i> atau memberikan <i>output</i>, maka aplikasi tersebut juga bisa dianggap sebagai aktor</p>
<p><i>Association</i></p> 	<p>Asosiasi digunakan untuk menghubungkan aktor dengan <i>use case</i>. Asosiasi digambarkan dengan sebuah garis yang menghubungkan antara aktor dengan <i>use case</i></p>






### 3. *Sequence Diagram*

Bersifat dinamis. Diagram urutan adalah interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu. Sebuah *Sequence* diagram adalah diagram interaksi yang menunjukkan bagaimana proses beroperasi dengan satu sama lain dan dalam rangka apa. Ini adalah *konstruksi dari Bagan Message Sequence*. Sebuah diagram urutan menunjukkan interaksi objek diatur dalam urutan waktu. Ini menggambarkan objek dan kelas yang terlibat dalam skenario dan urutan pesan yang dipertukarkan antara objek yang dibutuhkan untuk melaksanakan *fungsi skenario*. *Sequence diagram* biasanya terkait dengan *realisasi use case* dalam *Logical View sistem* dalam pengembangan. *Sequence diagram* kadang-kadang disebut diagram peristiwa atau skenario acara.

**Tabel 2. 3** Simbol-simbol pada *Sequence Diagram*

Simbol	Keterangan
Aktor/ <i>actor</i> 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda diawal frase nama aktor
Garis hidup/ <i>lifeline</i> .....	Menyatakan himpunan suatu objek
Objek 	Menyatakan objek yang berisi pesan

<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi pesan</p>
<p>Pesan tipe <i>create</i></p> 	<p>Menyatakan suatu objek membuat objek lain, arah panah objek yang dibuat</p>
<p>Pesan tipe <i>call</i></p> 	<p>Menyatakan suatu objek memanggil operasi.metode</p>
<p>1: nama metode()</p> 	<p>Ada pada objek lain atau dirinyasendiri.</p>
<p>pesan tipe <i>send</i></p> <p>1: masukan</p> 	<p>Menyatakan bahwa suatu objek mengirimkan data/masukan informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.</p>
<p>Pesan tipe <i>return</i></p> <p>1: masukan</p> 	<p>Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian</p>
<p>Pesan tipe <i>destroy</i></p> 	<p>Menyatakan suatu objek mengakhiri objek lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destory</p>

**4. Diagram komunikasi (*communication diagram*).**

Bersifat dinamis. Diagram sebagai pengganti diagram kolaborasi UML 1.4 yang menekankan organisasi struktural dari objek-objek yang menerima serta mengirim pesan.

**5. Diagram aktivitas (*activity diagram*).**

Diagram aktivitas adalah representasi grafis dari seluruh tahapan alur kerja. Diagram ini mengandung aktivitas, pilihan tindakan, perulangan dan hasil dari aktivitas tersebut. Pada pemodelan UML, diagram ini dapat digunakan untuk menjelaskan proses bisnis dan alur kerja operasional secara langkah demi langkah dari komponen suatu sistem.

**Tabel 2. 4** Simbol\_simbol pada *Activity Diagram*


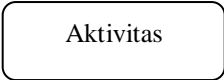
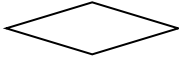



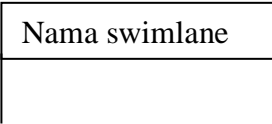
Simbol	Keterangan
Status 	Status awal aktivitas system sebuah diagram aktivitas memiliki sebuah status awal
Aktivitas 	Aktivitas yang dilakukan system, aktivitas biasanya diawali dengan kata kerja
Percabangan/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan/ <i>join</i> 	Asosiasi percabangan dimana lebih dari satu aktivitas digabungkan menjadi satu

Table	Semua file computer dari mana data dibaca atau direkam selama kejadian bisnis
Dokumen 	Menunjukkan dokumen sumber atau laporan
Status akhir 	Status akhir yang dilakukan system sebuah diagram aktivitas memiliki sebuah status akhir
Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

#### 2.2.4 Basis Data

Basis data merupakan penyimpanan informasi yang terorganisasi secara komputerisasi sehingga memudahkan pemakai dalam pengolahannya dan penggunaannya (*Subekti, 20013*). Pada saat ini sistem basis data sudah dikembangkan pada mesin-mesin komputer kecil sampai komputer yang lebih besar seperti mainframe. Tujuan dari hal tersebut secara keseluruhan adalah untuk melakukan perawatan informasi dan menyajikannya kapan saja dibutuhkan oleh pengguna.

#### 2.2.5 Web Server

*Web server* adalah sebuah bentuk server yang khusus digunakan untuk menyimpan halaman *website* atau *homepage*. Komputer dapat dikatakan web server jika komputer tersebut memiliki suatu program server yang disebut *Personal Web Server (PWS)* [14].

### 2.2.6 MySQL

Menurut Kustiyahningsih (2011), “MySQL adalah sebuah basis data yang mengandung satu atau jumlah table. Table terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel”[15].

MySQL adalah database server open source yang cukup populer keberadaanya. Dengan berbagai keunggulan yang dimiliki, membuat software database ini banyak digunakan oleh praktisi untuk membangun suatu project. Adanya fasilitas API (*Application Programming Interface*) yang dimiliki oleh Mysql, memungkinkan bermacam-macam aplikasi Komputer yang ditulis dengan berbagai bahasa pemrograman dapat mengakses basis data MySQL (Komputer W, 2010).

MYSQL termasuk jenis RDBMS (*Relational Database Management System*). Sedangkan RDMS sendiri akan lebih banyak mengenal istilah seperti table, baris dan kolom digunakan dalam perintah-perintah di MYUML.

*~Halaman Ini Sengaja Dikosongkan~*