

BAB II
TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Beberapa acuan yang digunakan dalam pengembangan sistem dan aplikasi ini yaitu sebagai berikut :

No	Penulis	Hasil Penelitian	Perbandingan dengan Penelitian Sebelumnya
1	Hidayatullah (2015)	Judul: PHP Framework CodeIgniter untuk E-commerce di Kawata Fishing Tackle WebStore Objek: Kawata Fishing Tackle E-commerce Topik : E-commerce	
2	Ahmad Sobri (2015)	Judul : Implementasi Teknologi Web Service Pada Sistem Informasi Akademik Berbasis Android Objek : Mahasiswa Topik : Web Service	
3	Dedy Permadi (2015)	Judul : Aplikasi Penjualan E-book Online Berbasis E-Commerce Objek : Buku Topik : E-commerce	
4	Andre Wiliantara (2015)	Judul : Pembangunan e-commerce sebagai media promosi dan penjualan di GWS Streatwear	

		Yogyakarta Object : GWS Streatwear Topik : E-commerce	
5	Musyafa'ah (2014)	Judul : Implementasi Web Service Studi Kasus Sistem Penggajian Di Pemerintahan Kabupaten Sleman Object : Pegawai Topik : Web Service	

Tabel 2.1 Penelitian Acuan Yang Digunakan Dalam Pengembangan Sistem.

2.2 Dasar Teori

Berdasarkan latar belakang dan tinjauan pustaka diberikan dasar teori sebagai berikut :

2.2.1 REST

Sampai dengan saat ini teknologi web *service* terus berkembang. Salah satu teknologi yang populer saat ini adalah REST (*Representational State Transfer*) atau terkadang disebut dengan RESTful (Saputra dan Ashari , 2012).

REST merupakan akronim dari *Representational State Transfer*, dan jika dibandingkan dengan protokol-protokol lainnya seperti SOAP atau XML-RPC, REST lebih diartikan sebagai seperangkat prinsip daripada sebagai protokol. REST adalah seperangkat ide tentang bagaimana data dapat ditransfer dengan

lancar. REST mengambil keuntungan besar dari fitur-fitur yang ada pada HTTP (Suwignyo dkk, 2015).

Menurut Pautasso (2008), metode REST didasari oleh empat prinsip utama teknologi yaitu :

- *Resource identifier* melalui URI (*Uniform Resource Identifier*), REST web *service* mencari sekumpulan sumber daya yang mengidentifikasi interaksi antar *client*.
- *Uniform Interface* , sumber daya yang dimanipulasi CRUD (*Create, Read, Update, Delete*) menggunakan operasi PUT, GET, POST, dan DELETE.
- *Self-descriptive message*, sumber daya informasi tidak terikat, sehingga dapat mengakses berbagai format konten (HTML, XML, PDF, JPEG, Plain Text dan lainnya). Metadataapun dapat digunakan.
- *Stateful interactions* melalui *hyperlinks*, setiap interaksi dengan suatu sumber daya bersifat *stateless*, yaitu *request messages* tergantung jenis kontennya.

Representational State Transfer disebut juga dengan RESTful adalah suatu arsitektur metode komunikasi dalam pengembangan layanan berbasis web. REST juga didefinisikan sebagai seperangkat ide tentang bagaimana data dapat ditransfer dengan lancar melalui fitur-fitur yang ada pada HTTP.

2.2.2 API

API merupakan *software interface* yang terdiri atas kumpulan instruksi yang disimpan dalam bentuk *library* dan menjelaskan bagaimana agar suatu *software* dapat berinteraksi dengan *software* lain. Penjelasan ini dapat dicontohkan dengan analogi apabila akan dibangun suatu rumah. Dengan menyewa kontraktor yang dapat menangani bagian yang berbeda, pemilik rumah dapat memberikan tugas yang perlu dilakukan oleh kontraktor tanpa harus mengetahui bagaimana cara kontraktor menyelesaikan pekerjaan tersebut. Dari analogi tersebut, rumah merupakan *software* yang akan dibuat, dan kontraktor merupakan API yang mengerjakan bagian tertentu dari *software* tersebut tanpa harus diketahui bagaimana prosedur dalam melakukan pekerjaan tersebut. (Reddy, 2011)

Secara umum, API menghadapkan satu set data dan fungsi untuk memfasilitasi interaksi antara program komputer dan memungkinkan mereka untuk pertukaran informasi. Web API adalah wajah (*Interface*) dari layanan web, langsung mendengarkan dan menanggapi permintaan klien. (Mark Masse, 2012)

2.2.3 Web Service

Web *service* adalah sistem perangkat lunak yang dirancang untuk mendukung interaksi yang bisa beroperasi *machine-to-machine* di atas jaringan. Web *service* mempunyai alat penghubung yang diuraikan di dalam format *machine-processable* (secara spesifik WSDL). Sistem lain saling berhubungan dengan web *service* di dalam cara yang ditentukan oleh deskripsinya yang

menggunakan pesan SOAP, secara khas disampaikan menggunakan HTTP dengan XML *serialization*, bersama dengan standar lain yang terkait dengan web (Booth et al., 2004).

Menurut Ghifary dan Gede Karya (2011), beberapa alasan mengapa digunakan web *service* adalah sebagai berikut :

- Web *Service* dapat digunakan untuk mentransformasikan satu atau beberapa bisnis *logic* atau *class* dan objek yang terpisah dalam satu ruang lingkup yang menjadi satu, sehingga tingkat keamanan dapat ditangani dengan baik.
- Web *service* memiliki kemudahan dalam proses *deployment*, karena tidak memerlukan registrasi khusus ke dalam suatu sistem operasi. Web *Service* cukup di *upload* ke web *server* dan siap diakses oleh pihak pihak yang telah diberikan otorisasi.
- Web *service* berjalan di port 80 yang merupakan protokol standar HTTP, dengan demikian web *service* tidak memerlukan konfigurasi khusus disisi *firewall*.
- Web *service* dapat meminimalisir kesalahan entri data/informasi yang timbul pada komunikasi *machine to human* karena komunikasi dapat dilakukan secara *machine to machine*.

2.2.4 E-commerce

Definisi *e-commerce* menurut David Baum (1999) yaitu: *ecommerce is a dynamic set of technologies, applications, and bussines process that link enterprises, consumers, and communities through electronics transactions and the electronic exchange of goods, services, and informations*. Diterjemahkan oleh Onno. W. Purbo (2002) : *e-commerce* merupakan satu set dinamis teknologi, aplikasi, dan proses bisnis yang menghubungkan perusahaan, konsumen, dan komunitas tertentu melalui transaksi elektronik dan perdagangan barang, pelayanan, dan informasi yang dilakukan secara elektronik.

Jenis-jenis transaksi pada *e-commerce* yaitu :

- *Busines to Busines (B2B)* adalah sistem komunikasi bisnis *online* antar pelaku bisnis, B2B terdiri atas Transaksi *Inter-Organizational system (IOS)* dan Transaksi Pasar Elektronik (*Electronic Market Transfer*), (Onno W. Purbo, 2000:2)
- *Busines to Cunsumer (B2C)* adalah mekanisme toko *online* yang artinya transaksi antara *e-merchant* dengan *e-customer*, (Onno W. Purbo, 2000:2).
- *Consumer to Consumer (C2C)* adalah individu yang mengiklankan produk barang atau jasa, pengetahuan, maupun keahliannya di salah satu situs lelang, (Munir Fuady, 2005:408).
- *Consumer to Bussines (C2B)* adalah individu yang menjual produk atau jasa kepada organisasi dan individu yang mencari penjual dan melakukan transaksi (Munir Fuady, 2005:408).

- *Non-Bussines Electronic Commerce* adalah kegiatan non bisnis seperti kegiatan lembaga pendidikan, organisasi nirlaba, keagamaan dan lain-lain (Munir Fuady, 2005:408).

- *Intrabussines (Organizational) Electronic Commerce* adalah kegiatan yang meliputi aktivitas internal organisasi melalui internet untuk melakukan pertukaran barang, jasa dan informasi, menjual produk perusahaan kepada karyawan dll. (Munir Fuady, 2005:408).

2.2.5 Pemrograman Client Server

Pembagian proses kerja adalah bagian utama dari konsep *client/server*. pembagian kerja pada *client* dan *server* telah diatur secara lebih spesifik :

- Two-tier : Membagi proses *load* kedalam dua bagian. Aplikasi utama secara logika dijalankan/ berjalan pada sisi *client* yang biasanya mengirimkan *request* dalam bentuk sintaks SQL ke sebuah *database server* yang berfungsi sebagai media penyimpanan data. Kita bisa juga menyebutnya dengan arsitektur *fat client* karena bagian terbesar atau yang utama dari aplikasi berjalan pada sisi komputer *client*.
- Three-tier : Membagi proses *loading* antara 1) komputer *client* menjalankan *graphical user interface (GUI) logic*, 2) aplikasi *server* menjalankan *business logic*, dan 3) *database* dan/ atau *legacy application*. Karena 3-tier memindahkan *application logic* ke *server* sehingga sering juga disebut sebagai arsitektur *fat server*.