

BAB II

Tinjauan Pustaka

2.1 Tinjauan Studi

Berikut adalah hasil dari penelitian yang terkait untuk melakukan penelitian ini adalah sebagai berikut :

2.1.1 Integrasi Sosial Media dengan E-learning Teknik Informatika Universitas Tanjungpura

Dari jurnal “Integrasi social media dengan E-learning Teknik Informatika Universitas Tanjungpura” permasalahan yang ada pada jurnal ini adalah kurangnya partisipasi para pendidik dan peserta didik, serta kesulitan untuk mendapatkan informasi dalam satu waktu yang mungkin dikarenakan oleh tidak menariknya konten atau fasilitas yang disajikan. Untuk mengatasi masalah tersebut diperlukan metode lain untuk memulai pola pendidikan yang baru yaitu memanfaatkan teknologi informasi tanpa menghilangkan sistem yang telah berjalan sebelumnya. Salah satunya penerapan *web service*. Keunggulan yang ditawarkan didalam *web service* adalah kemampuan untuk melakukan komunikasi antar sistem yang berbeda *website* dan *database*. Dengan adanya *web service* dan juga tersedianya API dari *facebook* dapat mempermudah bagi *developer* untuk mengintegrasikan sistemnya. Mengintegrasikan e-learning dengan sosial media diharapkan mampu menarik minat *user* untuk menggunakan e-learning tanpa meninggalkan aktivitas di sosial media. *Web service* juga dapat membantu sinkronisasi data yang diperlukan dalam sebuah *website* akan lebih mudah dilakukan [5].

Untuk menggunakan aplikasi yang menggunakan Facebook API yang pertama dilakukan yaitu masuk Facebook Developer (<http://developer.facebook.com>) untuk mendapatkan informasi tentang APP ID dan APP Secret ID. Informasi tersebut sangat berguna agar sistem dikenali oleh facebook. Yang mana pada aplikasi mengintegrasikan NIP / NIM pengguna dengan

akun id facebook pengguna. Proses integrasi dilakukan saat pertama login dengan facebook, dimana sistem akan mengambil data id dan nama facebook dari pengguna kemudian data id akan disimpan di session. Jika id facebook sudah terdaftar maka pengguna akan diarahkan langsung ke halaman *home* sistem. Dan jika belum terdaftar maka pengguna akan diarahkan ke halaman login register yang mana pada halaman login register pengguna akan diminta untuk menginputkan nim dan nip dan password akun e-learningnya, dan sistem akan memeriksa apakah akun e-learning pengguna sudah benar atau tidak, jika benar maka sistem akan memeriksa data tersebut sudah terdaftar atau belum jika sudah terdaftar maka pengguna akan diarahkan kembali ke halaman login register, jika nama facebook sudah yang telah di session dengan nip / nim pengguna yang telah diinputkan. Data id facebook, nama facebook dan nip / nim pengguna disimpan ke dalam database.

Data yang dikirim ke database akan mengirim notifikasi, nip / nim digunakan untuk mencari siapa-siapa saja yang berhak untuk menerima notifikasi tersebut. Yang mana nantinya jika dosen login menggunakan akun login dosen dan mengirimkan materi dan tugas kepada mahasiswanya maka notifikasi akan masuk ke akun pengguna masing-masing mahasiswa. Dosen juga bisa membuat forum, mahasiswa dan dosen bisa mengomentari forum [3].

Perbedaan dari penelitian di atas dengan penelitian yang akan dibuat adalah sama-sama menerapkan *web service* tetapi pada penelitian yang dibuat lebih pada pengiriman informasi produk sedangkan penelitian di atas lebih kepada pembelajaran yaitu *e-learning*.

2.1.2 Rancang Bangun Sistem Informasi Perpustakaan Menggunakan Web Service pada Jurusan Teknik Komputer Polsri

Dari jurnal “Rancang bangun sistem informasi perpustakaan menggunakan web service pada jurusan Teknik Komputer Polsri” permasalahan yang ada pada jurnal ini adalah dalam penyimpanan data pembukuan masih secara manual. Seiring teknologi semakin berkembang sangat pesat terutama dalam pengembangan Aplikasi, diharapkan perpustakaan Teknik Komputer memiliki aplikasi khusus

yang terhubung dengan internet dengan tujuan mempermudah mahasiswa atau dosen Teknik Komputer untuk mengakses perpustakaan dari jarak jauh secara online serta memudahkan admin perpustakaan dalam melaksanakan tugas administrative. Akan tetapi teknologi akan terus semakin berkembang. Aplikasi yang sudah dibuat tidak akan bertahan untuk jangka panjang. Akan ada teknologi teknologi baru yang bermunculan untuk waktu kedepannya. Untuk menghindari pembuatan ulang aplikasi terbaru, maka web service solusinya.

Dengan begitu, ketika akan membuat dan mengembangkan aplikasi perpustakaan terbaru untuk masa kedepannya, tidak lagi diperlukan untuk membuat ulang struktur dan metode pengaksesan sistem informasinya. Hanya saja diperlukan untuk mengembangkan aplikasi dari segi tampilan dan fitur, sedangkan aplikasi itu akan tetap mengakses data yang sama dari server yang memiliki layanan web service. Adapun tujuan dari *Web Service* adalah membuat sistem informasi yang flexible, fungsional dan terintegrasi dengan dengan aplikasi lainnya yang membutuhkan data dari perpustakaan di jurusan Teknik Komputer [6].

Setelah itu, dilakukan pengujian Web Service agar dapat memastikan bahwa Web Service yang dibangun dapat bekerja dengan baik. Untuk pengujian Restfull Web Service bisa menggunakan *tools advanced REST Client* atau POSTMAN.

Dari penelitian di atas dengan penelitian yang dibuat juga menerapkan *Web Service* tetapi penelitian di atas lebih pada pengembangan aplikasi di perpustakaan sedangkan aplikasi pada penelitian ini dibuat untuk melakukan pengiriman informasi produk.

2.1.3 Interkoneksi Jejaring Sosial Twitter dan Sistem Informasi Berbasis Web dengan Menerapkan Web Service, Crontab dan API

Dari jurnal “Interkoneksi jejaring sosial twitter dan sistem informasi berbasis web dengan menerapkan web service, crontab dan API” permasalahan yang ada yaitu mahasiswa yang ingin melakukan bimbingan terkadang sulit untuk bertemu dengan dosen pembimbingnya. Hal ini disebabkan oleh sibuknya jadwal belajar-mengajar dosen dan kegiatan lainnya. Untuk itu dibutuhkan sebuah

perantara yang dapat menghubungkan dosen pembimbing dan mahasiswa agar jadwal bimbingan dapat dilakukan tanpa harus mengganggu jadwal lain. Teknologi media sosial twitter dapat digunakan sebagai antarmuka untuk mengirimkan jadwal bimbingan dengan sistem informasi yang bertindak sebagai penerima, pengumpul dan pengelola data, kemudian membuat aplikasi untuk menyimpan mention khusus sebagai request pada basisdata sehingga mahasiswa dapat melihat jadwal bimbingan yang up to date [7].

Pada penelitian di atas adalah membuat aplikasi yang memudahkan mahasiswa untuk melakukan bimbingan dengan dosen pembimbing sedangkan aplikasi pada penelitian ini dibuat untuk melakukan pengiriman informasi produk dan sama-sama menerapkan Web Service pada penelitian yang akan dibuat.

2.2 Tinjauan Pustaka

Dalam penyusunan penelitian ini, ada beberapa landasan teori atau konsep yang digunakan untuk mendukung pemahaman lebih lanjut mengenai hasil dari analisis yang akan dibahas pada bab berikut.

2.2.1 Media Sosial

Media sosial adalah sebuah media online, dengan para penggunanya bisa dengan mudah berpartisipasi, berbagi, dan menciptakan isi meliputi blog, jejaring sosial, wiki, forum dan dunia virtual. Pendapat lain mengatakan bahwa media sosial adalah media online yang mendukung interaksi sosial dan media sosial menggunakan teknologi berbasis web yang mengubah komunikasi menjadi dialog interaktif [5].

2.2.2 Informasi

Menurut Jogiyanto, Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. Dapat dikatakan bahwa data merupakan bahan mentah, sedangkan informasi adalah bahan jadi atau bahan yang telah siap digunakan, Jadi, sumber dari informasi adalah data. Data merupakan

bentuk jamak dari bentuk tunggal datum atau data-item. Data adalah kenyataan yang menggambarkan suatu kejadian dan kesatuan nyata. Kejadian (event) adalah sesuatu yang terjadi pada saat tertentu. Di dalam dunia bisnis, kejadian- kejadian nyata adalah perubahan dari suatu nilai yang disebut dengan transaksi. Indikator yang dapat digunakan untuk mengukur kualitas informasi adalah *relevance*, *accurate*, *completeness*, dan *understandability* dari informasi yang dihasilkan [8].

2.2.3 Otomatisasi

Menurut Sedarmayanti bahwa otomatisasi adalah cara pelaksanaan prosedur dan tata kerja secara otomatis, dengan pemanfaatan yang menyeluruh dan seefisien mungkin atau mesin, sehingga bahan dan sumber yang ada dapat dimanfaatkan [9].

2.2.4 Produk

Produk adalah barang atau jasa yang bisa ditawarkan dipasar untuk mendapatkan perhatian, permintaan, pemakaian, atau konsumsi yang dapat memenuhi keinginan atau kebutuhan [10].

2.2.5 Web Service

W3C mendefinisikan web service sebagai sebuah sistem perangkat lunak yang dirancang untuk mendukung komunikasi dan interaksi antar mesin ke mesin (*Machine to Machine*) melalui sebuah network (jaringan). Web Service juga termasuk Web APIs yang dapat diakses melalui jaringan seperti misalnya internet, dan dieksekusi melalui sebuah sistem jarak jauh sesuai dengan layanan yang diminta [11].

2.2.6 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 [12].

Pemahaman tentang application programming interface (API) sangat penting untuk memahami pengertian application interfaces. API merupakan mekanisme yang telah didefinisikan dengan baik yang dikembangkan untuk menghubungkan sumber daya yang ada seperti misalnya *application server*, *middleware layer*, atau database. API memungkinkan developer untuk menggunakan layanan dari entitas-entitas yang ada untuk memperoleh nilai yang dimiliki oleh entitas tersebut. Misalnya untuk mengakses informasi pelanggan akan membutuhkan perantara API ke database pelanggan. Alasan dibuat API adalah :

- a. Menyediakan akses ke data dan proses bisnis tanpa harus melalui user interface.
- b. Menyediakan mekanisme yang memungkinkan berbagi informasi [13].

2.2.7 PHP

PHP singkatan dari PHP Hypertext Preprocessor yaitu bahasa pemrograman web *server-side* yang bersifat *open source*. PHP merupakan *script* yang terintegrasi dengan HTML dan berada pada server (*server side HTML embedded scripting*). PHP adalah *script* yang digunakan untuk membuat halaman *website* yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client*. Mekanisme ini menyebabkan informasi yang diterima *client* selalu terbaru/*up to date*. Semua *script* PHP dieksekusi pada server dimana *script* tersebut dijalankan. Beberapa alasan untuk mempelajari PHP adalah karena:

- a. Kesederhanaan. *User* yang baru belajar pemrograman, alasan ini pasti merupakan alasan utama untuk mulai belajar PHP. Karena kesederhanaan tersebut, maka kita menjadi merasa mudah untuk belajar PHP. *User* yang sedikit tahu atau bahkan sama sekali tidak mengerti tentang pemrograman PHP bisa dengan cepat belajar dan mencoba membuat aplikasi *web* PHP. Selain itu, PHP memiliki banyak sekali fungsi *built-in* untuk menangani kebutuhan standar pembuatan aplikasi *web*. Dengan adanya fungsi-fungsi tersebut, maka tentu saja proses belajar PHP terutama dalam pengembangan aplikasi akan jauh

- lebih mudah karena semua sudah tersedia. semua *user* tinggal memakai dan mengembangkannya.
- b. Dalam sisi pemahaman, PHP adalah scripting yang paling mudah karena memiliki referensi yang banyak.
 - c. PHP adalah bahasa *open source* yang dapat digunakan di berbagai sistem operasi seperti: Linux, Unix, Macintosh, dan Windows. PHP dapat dijalankan secara *runtime* melalui *console* serta dapat menjalankan perintah - perintah sistem. *Open source* artinya code-code PHP terbuka untuk umum dan tidak harus membayar biaya pembelian atas keaslian *license* yang biasanya cukup mahal. Karena *source code* PHP tersedia secara gratis, maka hal tersebut memungkinkan komunitas milis-milis dan *developer* untuk selalu melakukan perbaikan, pengembangan, dan menemukan bug dalam bahasa PHP.
 - d. *Web Server* yang mendukung PHP dapat ditemukan di mana-mana mulai dari Apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relatif mudah.

PHP juga dilengkapi dengan berbagai macam pendukung lain seperti *support* langsung ke berbagai macam *database* yang populer, misal: Oracle, PostgreSQL, MySQL, dan lain-lain [14].

2.2.8 Xampp

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU General Public License dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis [15].

2.2.9 C#

C# merupakan salah satu aplikasi yang memiliki kemampuan dalam penguatan Framework.NET. C# dibuat sejalan dengan perkembangan Framework.NET, C# sendiri dikembangkan oleh Microsoft. Dalam penerapannya C-Sharp (C#) menjanjikan produktifitas, fleksibilitas serta kemudahan yang ada dari aplikasi sebelumnya yaitu Visual Basic, Java dan C++. C# mengadopsi kemampuan dari penggabungan aplikasi sebelumnya (mempelajari-c) Microsoft membuat C# seiring dengan pembuatan Framework.NET. Chief Architect dalam pembuatan C# adalah Anders Hejlsberg yang sebelumnya berperan dalam pembuatan Borland Delphi dan Turbo Pascal. C# menjanjikan produktifitas dan kemudahan yang ada di Visual Basic dengan kemampuan dan fleksibilitas yang ada di C/C++ [16].

2.2.10 Xamarin Forms

Xamarin Forms merupakan sebuah library yang dikembangkan oleh Xamarin yang memungkinkan pengembangan lintas mobile platform secara native, termasuk di dalamnya iOS, Android, dan Windows Phone. Dengan Xamarin Forms, setiap kontrol dan halaman akan diterjemahkan sesuai dengan sistem operasi yang menjalankannya [17].

2.2.11 UML

Rancangan sistem adalah merancang atau mendesain sistem yang baik, isinya adalah langkah-langkah operasi dalam pengelolaan data dan prosedur untuk operasi sistem. Kegiatan yang dilakukan pada tahap ini yaitu merancang atau mendesain sistem yang baik, mendesain permodelan sistem yang baik, mengenali dan mendefinisikan masalah pembuatan sistem ini sehingga jika ada kesalahan alternatif pemecahannya.

Adapun rancangan model sistem yang digunakan adalah Diagram UML karena berbasis web. Diagram UML (Unified Modeling Language) sendiri adalah sebuah bahasa untuk menentukan, visualisasi, konstruksi, dan mendokumentasikan artifact (bagian dari informasi yang digunakan atau dihasilkan dalam suatu proses

pembuatan perangkat lunak. Artifact dapat berupa model, deskripsi atau perangkat lunak) dari system perangkat lunak, seperti pada pemodelan bisnis dan system non perangkat lunak lainnya.

UML merupakan bahasa standar untuk penulisan blueprint software yang digunakan untuk visualisasi, spesifikasi, pembentukan dan pendokumentasian alatalat dari sistem perangkat lunak.

UML memiliki beberapa jenis yaitu Use Case Diagram, Activity Diagram, Package Diagram, State Machine Diagram, Sequence Diagram, Class Diagram, Communication Diagram, Composite Structure Diagram, Object Diagram, Timing Diagram, Component Diagram, Deployment Diagram dan Interaction Overview Diagram [18].

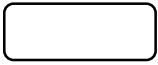
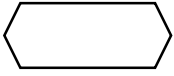
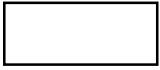
2.2.12 Flowchart

Flowchart adalah representasi secara simbolik dari suatu algoritma atau prosedur untuk menyelesaikan suatu masalah, dengan menggunakan *flowchart* akan memudahkan pengguna melakukan pengecekan bagian-bagian yang terlupakan dalam analisis masalah, disamping itu *flowchart* juga berguna sebagai fasilitas untuk berkomunikasi antara pemrogram yang bekerja dalam tim suatu proyek.

Flowchart membantu memahami urutan-urutan logika yang rumit dan panjang. *Flowchart* membantu mengkomunikasikan jalannya program ke orang lain (bukan pemrogram) akan lebih mudah [19].

Tabel 2. 1 Tabel Flowchart

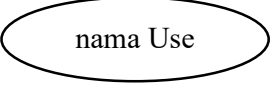
Simbol	Fungsi
	Permulaan Sub Program
	Perbandingan, Pernyataan , Penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	Penghubung bagian-bagian flowchart yang berada pada satu halaman.

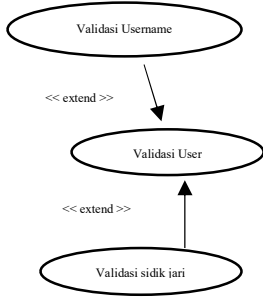
	Penghubung bagian-bagian flowchart yang berada pada satu halaman berbeda
	Permulaan / akhir program
	Arah aliran program
	Proses inisialisasi / pemberian harga awal
	Proses penghubung / proses pengolahan data
	Proses input / output data

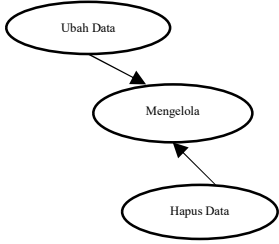
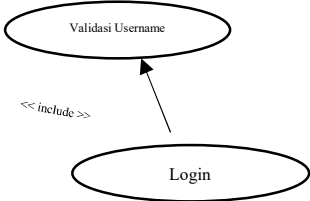
2.2.13 Use Case

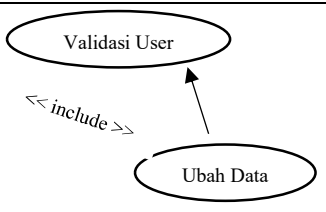
Use case atau diagram use case merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem informasi yang akan dibuat. Secara kasar, use case juga digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi – fungsi itu.

Tabel 2. 2 Tabel Simbol Use Case

Simbol	Deskripsi
 Use case	Fungsionalitasnya yang disediakan sistem sebagai unit – unit yang saling bertukar pesan antara unit atau actor, biasanya dinyatakan dengan menggunakan kata kerja di awal di awal frase nama use case.
	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang

Actor	akan dibuat itu sendiri, jadi walaupun simbol dari aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
_____ Asosiasi / association	Komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor.
<< extend >> ▶ Ekstensi / extend	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu, mirip dengan prinsip inheritance pada pemrograman berorientasi objek, biasanya use case tambahan memiliki nama depan yang sama dengan use case yang ditambahkan, misal :  <pre> graph TD A([Validasi Username]) B([Validasi User]) C([Validasi sidik jari]) A -.-> << extend >> B C -.-> << extend >> B </pre> arah panah mengarah pada use case yang ditambahkan, biasanya use case yang menjadi extend-nya merupakan jenis yang sama dengan use case yang menjadi induknya.
————▶	Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah use case

Generalisasi / generalization	dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya, misalnya :  arah panah mengarah pada use case yang menjadi generalisasinya (umum)
Menggunakan / include / uses	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini.
<< include >>  << uses >> 	ada dua sudut pandang yang cukup besar mengenai include di use case: • Include berarti use case yang ditambahkan akan selalu di panggil saat use case tambahan di jalankan, misal pada proses berikut :  • Include berarti use case yang tambahan akan selalu melakukan pengecekan apakah use case akan ditambahkan telah dijalankan sebelum use case tambahan dijalankan, misal pada proses berikut:

	 <pre> graph BT A([Ubah Data]) -- "<< include >>" --> B([Validasi User]) </pre> Kedua interpretasi diatas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan.
--	--

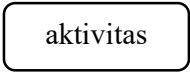
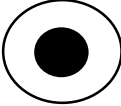
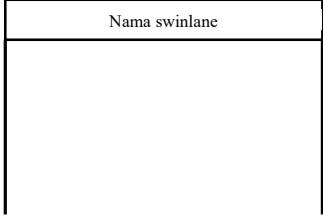
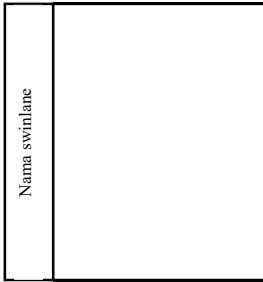
2.2.14 Activity Diagram

Diagram aktivitas atau activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

- Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
- Urutan atau pengelompokan tampilan dari sistem / user *interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
- Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
- Rancangan menu yang ditampilkan pada perangkat lunak.

Tabel 2. 3 Tabel Simbol Activity Diagram

Simbol	Deskripsi
 Status Awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.

 Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
 Percabangan / decision	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
 Penggabungan / join	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
Swinlane  atau 	Memisalkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

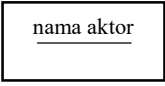
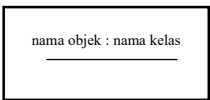
2.2.15 Sequence Diagram

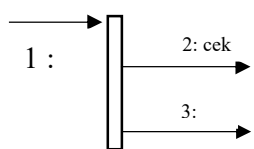
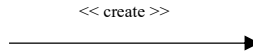
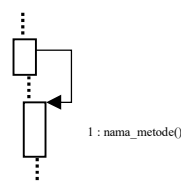
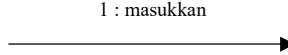
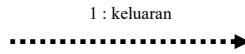
Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode

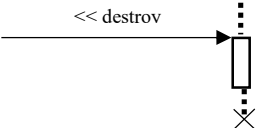
yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*.

Banyaknya diagram sekuen yang harus Digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak *use case* yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak.

Tabel 2. 4 Tabel Simbol Sequence Diagram

Simbol	Deskripsi
Aktor  nama aktor atau  tanpa waktu aktif	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 adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal frase nama aktor.
Garis hidup / lifeline 	Menyatakan kehidupan suatu objek.
Objek 	Menyatakan objek yang berinteraksi pesan.
Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnya:

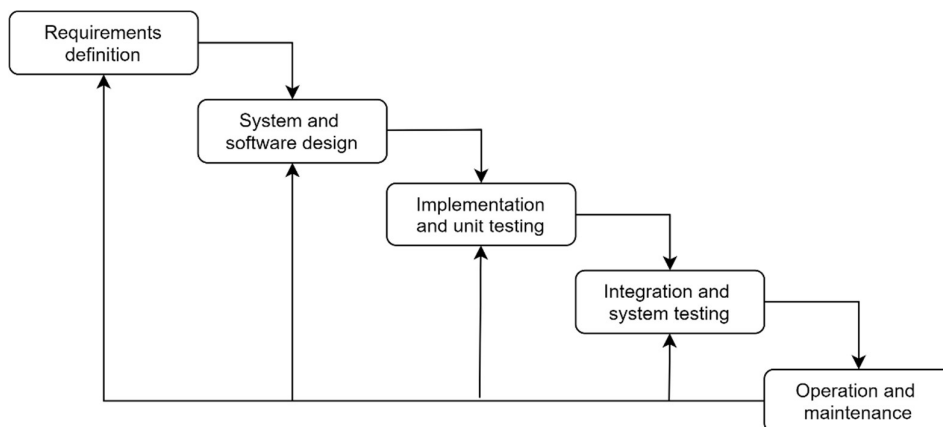
	 Maka cekStatusLogin() dan Open() dilakukan di dalam metode login() Aktor tidak memiliki waktu aktif
Pesan tipe create 	Menyatakan suatu objek membuat objek lain, arah panah mengarah pada objek yang dibuat.
Pesan tipe call 	Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri,  arah panah mengarah pada objek yang memiliki operasi atau metode, karena ini memanggil operasi / metode maka operasi / metode yang dipanggil harus sesuai dengan kelas objek yang berinterasi.
Pesan tipe send 	Menyatakan bahwa suatu objek mengirimkan data / masukkan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
Pesan tipe return 	menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu objek tertentu, arah panah mengarah pada objek yang menerima kembalian.

Pesan tipe destroy 	Menyatakan suatu objek mengakhiri hidup objek lain, arah panah mengarah pada objek yang diakhiri sebaiknya jika ada create maka ada destroy.
---	--

Penomoran pesan berdasarkan urutan interaksi pesan. Penggambaran letak pesan harus berurutan, pesan yang lebih atas dari lainnya adalah pesan yang berjalan terlebih dahulu [20].

2.2.16 Metode Waterfall

Model pengembangan software yang diperkenalkan oleh Winston Royce pada tahun 70-an ini merupakan model klasik yang sederhana dengan aliran sistem yang linier — keluaran dari tahap sebelumnya merupakan masukan untuk tahap berikutnya. Pengembangan dengan model ini adalah hasil adaptasi dari pengembangan perangkat keras, karena pada waktu itu belum terdapat metodologi pengembangan perangkat lunak yang lain. Proses pengembangan yang sangat terstruktur ini membuat potensi kerugian akibat kesalahan pada proses sebelumnya sangat besar dan acap kali mahal karena membengkaknya biaya pengembangan ulang.



Gambar 2. 1 Metode Waterfall

Metode Waterfall adalah suatu proses pengembangan perangkat lunak berurutan, di mana kemajuan dipandang sebagai terus mengalir ke bawah (seperti air terjun) melewati fase-fase perencanaan, pemodelan, implementasi (konstruksi), dan pengujian. Dalam pengembangannya metode waterfall memiliki beberapa tahapan yang runtut: requirement (analisis kebutuhan), design sistem (system design), Coding & Testing, Penerapan Program, pemeliharaan.

a. Requirement (analisis kebutuhan)

Dalam langkah ini merupakan analisa terhadap kebutuhan sistem. Pengumpulan data dalam tahap ini bisa melakukan sebuah penelitian, wawancara atau study literatur. Seseorang system analisis akan menggali informasi sebanyak-banyaknya dari user sehingga akan tercipta sebuah sistem komputer yang bisa melakukan tugas-tugas yang diinginkan oleh user tersebut. Tahapan ini akan menghasilkan dokumen user requirement atau bisa dikatakan sebagai data yang berhubungan dengan keinginan user dalam pembuatan sistem. Dokumen inilah yang akan menjadi acuan system analisis untuk menterjemahkan kedalam bahasa pemrograman.

b. Design System (design sistem)

Proses design akan menterjemahkan syarat kebutuhan sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat coding. Proses ini berfokus pada : struktur data, arsitektur perangkat lunak, representasi interface, dan detail (algoritma) prosedural. Tahapan ini akan menghasilkan dokumen yang disebut software requirement. Dokumen inilah yang akan digunakan programmer untuk melakukan aktivitas pembuatan sistemnya.

c. Coding & Testing (penulisan sinkode program / implementation)

Coding merupakan penerjemahan design dalam bahasa yang bisa dikenali oleh komputer. Dilakukan oleh programmer yang akan menterjemahkan transaksi yang diminta oleh user. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu sistem. Dalam artian penggunaan computer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem yang telah dibuat tadi.

Tujuan testing adalah menemukan kesalahan/kesalahan terhadap system tersebut dan kemudian bisa diperbaiki.

d. Penerapan / Pengujian Program (Integration & Testing)

Tahapan ini bisa dikatakan final dalam pembuatan sebuah sistem. Setelah melakukan analisa, design dan pengkodean maka sistem yang sudah jadi akan digunakan oleh user.

e. Pemeliharaan (Operation & Maintenance)

Perangkat lunak yang sudah disampaikan kepada pelanggan pasti akan mengalami perubahan. Perubahan tersebut bisa karena mengalami kesalahan karena perangkat lunak harus menyesuaikan dengan lingkungan (peripheral atau system operasi baru) baru, atau karena pelanggan membutuhkan perkembangan fungsional [21].

2.2.17 Postman

Postman adalah sebuah aplikasi (berupa plugin) untuk browser chrome, yang berfungsi sebagai REST Client, yang digunakan untuk melakukan uji coba REST API. Postman juga dapat diunduh sebagai aplikasi desktop. Postman merupakan platform GUI yang powerful untuk membuat pengembangan API lebih cepat dan mudah, mulai dari membangun API melalui testing, dokumentasi, dan sharing. Postman direkomendasikan untuk sistem operasi Mac, Windows atau juga Linux [22].

Postman merupakan sebuah REST client berbasis web yang tersedia dalam bentuk ekstensi pada Google Chrome. Sebuah tool yang membantu dalam mengembangkan REST Web Services. Postman adalah HTTP client yang kuat untuk menguji layanan web. Dibuat oleh Abhinav Asthana, programmer dan desainer yang berbasis di Bagalore, India. Postman memudahkan untuk menguji, mengembangkan dan API (Application Programmin Interface) dokumen dengan memungkinkan pengguna untuk dengan cepat mengumpulkan baik permintaan HTTP sederhana dan kompleks [23].

2.2.18 JSON

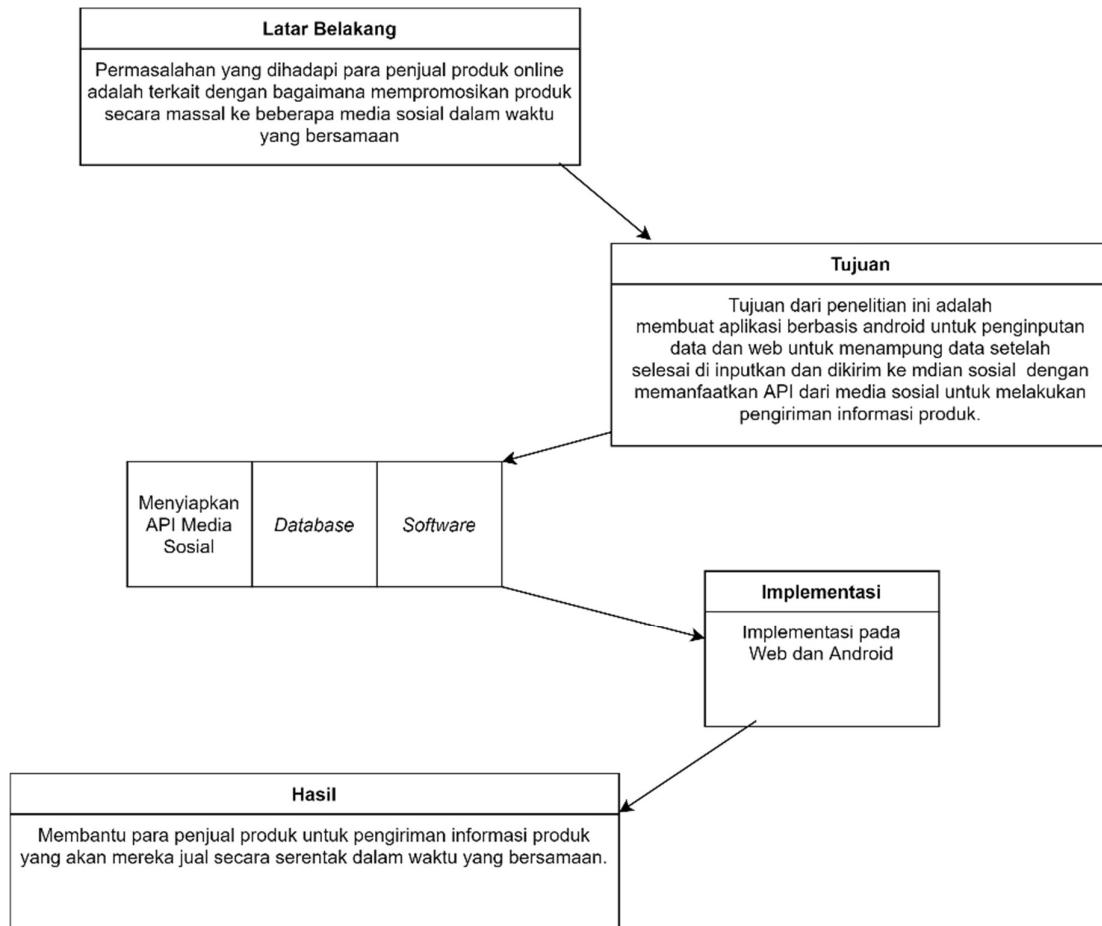
pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh komputer. Format ini dibuat berdasarkan bagian dari bahasa pemrograman JavaScript. JSON terdiri dari dua struktur, yaitu:

1. Pasangan nama dengan nilai. Pada beberapa bahasa hal ini dinyatakan sebagai object, record, struct, dictionary, hashtable, keyedlist atau associative array.
2. Daftar nilai terurutkan (anorderedlist of values). Pada kebanyakan bahasa, hal ini dinyatakan sebagai array, vector, list atau sequence (Abdul dkk, 2013:2) [6].

JSON (Javascript Object Notation) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh computer, menurut Json Org. JSON(2005). JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran data [24].

2.3 Kerangka Pemikiran

Berikut adalah kerangka pemikiran dari penelitian yang akan dibuat untuk pengiriman informasi produk.



Gambar 2. 2 Kerangka Pemikira