

I. PENDAHULUAN

1.1 Latar Belakang

Perkembangan zaman membuat dunia usaha berkembang begitu pesat sehingga banyak orang yang memutuskan untuk memulai suatu usaha baik sebagai badan hukum maupun perorangan. Salah satu cara mudah dan cepat untuk memulai usaha yaitu dengan mendaftarkan usaha ke situs *online marketplace*. Kemudahan yang diberikan *marketplace* menyebabkan banyaknya pelaku usaha lain yang mendaftarkan usahanya sehingga menciptakan suatu persaingan. Dengan semakin bertambahnya produsen/perusahaan yang menawarkan barang dalam bentuk produk dan jasa, maka semakin banyak juga pilihan yang dimiliki oleh konsumen ketika menentukan jadi tidaknya membeli (Ekasari & Putri, 2021).

Dommo merupakan toko *online* yang menjual ulang (*reseller*) berbagai produk brand lokal maupun impor di *online marketplace* shopee. Banyaknya produk yang dijual di toko *online* dommo membuat pegawainya kesulitan dalam memonitor harga-harga produk pesaingnya, ditambah lagi banyaknya *event-event* yang sering diadakan oleh *online marketplace* shopee sehingga menyebabkan *update* harga produk sering terjadi. Kesulitan ini dapat menghambat proses pengambilan keputusan dalam penentuan harga jual, terhambatnya proses pengambilan keputusan ini dapat menyebabkan berkurangnya konsumen, karena harga jual produk adalah salah satu faktor utama untuk menarik konsumen.

Harga jual dapat memberikan nilai tertentu terhadap produk, harga jual juga dapat mempengaruhi psikologi konsumen dalam mengambil keputusan pembelian, umumnya konsumen akan membandingkan harga produk dari tiap sumber sebelum melakukan pembelian. Pernyataan ini juga sesuai dengan penelitian yang dilakukan Ekasari & Putri (2021) yang berjudul “Pengaruh Kualitas Pelayanan, Persepsi Harga, dan Lokasi Terhadap Keputusan Pembelian Konsumen” di penelitian tersebut disimpulkan bahwa “Persepsi harga mempunyai pengaruh terhadap keputusan pembelian”, oleh karena itu pelaku usaha diharuskan untuk mempersiapkan data perbandingan harga produknya secara langsung ke produk pesaing, dengan tujuan untuk membandingkan dan menetapkan harga jual produknya. Informasi ini dapat diperoleh dari mengumpulkan data harga-harga produk pesaing yang terdapat pada *online marketplace*.

Web scraping adalah salah satu metode untuk mengumpulkan data yang terdapat pada suatu website. Penelitian tentang *web scraping* deskripsi produk

juga dilakukan oleh Setiawan et al. (2020) dengan judul “Aplikasi *Web Scraping* Deskripsi Produk”. Pada penelitian tersebut proses *web scraping* dilakukan hanya saat pengguna melakukan input alamat website produk. Data yang diperoleh dari *web scraping* dapat kadaluarsa dengan cepat dikarenakan kemudahan yang diberikan oleh *online marketplace* kepada pelaku usaha untuk mengubah data produknya. Untuk itu perlu dilakukan proses *web scraping* secara berkala.

Untuk memonitor data informasi yang didapat dari *web scraping* maka diperlukan sebuah sistem informasi *monitoring*. Umumnya sistem informasi dibangun pada sebuah website (berbasis web), seperti yang dilakukan oleh Tirzasari et al. (2022) pada penelitiannya yang berjudul “Pengembangan Sistem Informasi Manajemen Persediaan Barang Berbasis Web”. Namun seperti yang disampaikan sebelumnya, data-data yang didapatkan dari proses *web scraping* dapat berubah setiap saat. Sebagai manusia yang memiliki keterbatasan, hal ini menyebabkan tidak memungkinkannya bagi pelaku usaha untuk memonitor informasi pada sebuah website selama 24 jam penuh. Serta untuk mengakses sebuah website diperlukan *bandwidth* yang cukup besar, sehingga akan menyulitkan pengaksesan di wilayah yang jaringan internetnya lemah. Untuk itu diperlukan sistem yang dapat memonitor dan memberikan peringatan kepada pelaku usaha ketika terjadi perubahan harga produk kapanpun dan dimanapun.

Pesan Tulisan (*text message*) adalah salah satu media selain web untuk membangun sistem informasi *monitoring*. Salah satu penelitian tentang membangun sistem informasi *monitoring* dengan media pesan tulisan dilakukan oleh Hakim & Nugroho (2019) yang berjudul “Implementasi Telegram Bot untuk Monitoring Mikrotik Router”. Penelitian tersebut memanfaatkan aplikasi telegram sebagai media pengiriman pesan. Telegram sendiri kurang memiliki popularitas pengguna di Indonesia, sehingga kebanyakan pengguna harus mengunduh, menginstal dan mendaftar di aplikasi telegram untuk dapat menggunakan sistem tersebut karena telegram bukan merupakan aplikasi yang umum digunakan. Salah satu aplikasi pesan yang umum digunakan di Indonesia menurut Global Web Index (GWI) adalah WhatsApp. WhatsApp adalah salah satu aplikasi komunikasi yang populer di Indonesia, WhatsApp juga memiliki fitur *notifikasi* dan fitur-fitur lainnya yang dapat digunakan sebagai sistem informasi. Dengan menghubungkan dan mendaftarkan sistem sebagai pengguna whatsapp, pelaku usaha dapat berkomunikasi secara langsung dengan sistem kapanpun dan dimanapun (selama terhubung ke jaringan internet) dan sistem dapat memberi peringatan *update* secara berkala kepada pelaku usaha.

Perancangan sistem informasi dapat menggunakan berbagai macam model, diantaranya adalah model *Waterfall*, RAD (*Rapid Application Development*)

dan *Prototype*. Penelitian yang menggunakan model *prototype* dilakukan oleh Robbi & Yulianti (2019) mengenai “Perancangan Aplikasi *E-Learning* Berbasis Web dengan Model *Prototype* pada SMPN 7 Kota Tangerang Selatan”, Model *prototype* ini digunakan untuk menggali dan menjabarkan spesifikasi kebutuhan pelanggan secara lebih detail sehingga sesuai dengan kebutuhan pelanggan, karena pelanggan sering kali kesulitan menyampaikan keinginannya secara detail.

Pada penelitian sistem informasi *monitoring* berbasis pesan yang dilakukan penulis ini, pengguna kesulitan dalam menjelaskan dan mendefinisikan secara keseluruhan tentang gambaran dari sistem. Serta penelitian ini juga ingin melibatkan pengguna untuk berpartisipasi aktif dalam menentukan kebutuhan sistem, oleh karena itu model *prototype* adalah model yang cocok digunakan pada penelitian kali ini.

Berdasarkan latar belakang di atas, maka dari itu penulis melakukan penelitian yang berjudul “RANCANG BANGUN SISTEM *MONITORING UPDATE* HARGA PESAING USAHA BERBASIS PESAN WHATSAPP MENGGUNAKAN MODEL *PROTOTYPE* (STUDI KASUS: TOKO *ONLINE* DOMMO)” dengan harapan dapat menyelesaikan masalah yang dijelaskan sebelumnya.

1.2 Rumusan Masalah

Berdasarkan identifikasi dari latar belakang, maka penulis menarik beberapa rumusan masalah:

1. Bagaimana membangun sistem informasi *monitoring update* harga produk pesaing usaha berbasis pesan whatsapp di toko *online* dommo menggunakan model *prototype*.
2. Bagaimana menguji efisiensi dan fungsionalitas dari sistem informasi *monitoring update* harga produk pesaing usaha berbasis pesan whatsapp di toko *online* dommo.

1.3 Tujuan Penelitian

Tujuan penelitian ini yaitu untuk:

1. Membangun sistem informasi *monitoring update* harga produk pesaing usaha berbasis pesan whatsapp di toko *online* dommo menggunakan model *prototype*.
2. Mengetahui efisiensi dan fungsionalitas dari sistem informasi *monitoring update* harga produk pesaing usaha berbasis pesan whatsapp di toko *online* dommo.

1.4 Batasan Masalah

Batasan masalah pada penelitian ini yaitu, Proses *web scraping* dan *monitoring* terbatas pada data *link* produk pesaing pada website shopee yang diberikan oleh toko *online* dommo

1.5 Manfaat Penelitian

Manfaat yang didapat dari penelitian ini adalah sebagai berikut:

1. Di bidang keilmuan
 - Menambah pengetahuan tentang cara *web scraping* informasi pada website shopee.
 - Menambah pengetahuan tentang cara mengubungkan sistem ke aplikasi pesan whatsapp.
 - Menambah pengetahuan tentang perancangan sistem informasi menggunakan metode *prototype*.
 - Menambah pengetahuan tentang cara menguji efisiensi dan fungsionalitas sistem.
2. Praktis
 - Menyajikan *user interface* yang menarik untuk *monitoring update* harga pesaing usaha meskipun terbatas pada pesan tulisan.
 - Mempermudah pelaku usaha untuk *monitoring update* harga pesaing usaha secara berkala sehingga dapat mempercepat proses pengambilan keputusan dalam penetapan harga produknya.

II. TINJAUAN PUSTAKA

2.1 Kewirausahaan

Kewirausahaan adalah kemampuan manajer resiko (*risk manager*) dalam mengoptimalkan segala sumber daya yang ada, baik itu materil, intelektual, waktu, dan kemampuan kretivitasnya untuk menghasilkan suatu produk atau usaha yang berguna bagi dirinya dan bagi orang lain (Firmansyah & Roosmawarni, 2019).

Strategi Usaha

Porter (1980) menyatakan ada tiga strategi generik yang berpotensi sukses dalam mengungguli persusahaan kompetitor yaitu: strategi biaya rendah (*cost leadership*), strategi pada diferensiasi produk (*differentiation product*), dan strategi yang berfokus pada segmen pasar (*focus strategy*). Penerapan strategi berbiaya rendah berkaitan dengan kemampuan efisiensi biaya produksi sehingga dapat menetapkan harga produk yang lebih rendah dibandingkan pesaing dan dapat meningkatkan pangsa pasar (Primadhita et al., 2021).

Harga merupakan suatu nilai yang dibuat untuk menjadi patokan nilai suatu barang. Harga jual juga dapat mempengaruhi psikologi konsumen dalam mengambil keputusan pembelian, umumnya konsumen akan membandingkan harga produk dari tiap sumber sebelum melakukan pembelian. Metode penetapan harga yang umum digunakan yaitu, penetapan harga yang kompetitif, penetapan harga agar sesuai, penetapan harga untuk mengalahkan pesaing (Firmansyah & Roosmawarni, 2019). Perusahaan harus mengantisipasi perubahan harga pesaing dan menyiapkan respon tertentu (Kotler & Keller, 2018), contohnya: memberikan layanan yang berbeda atau memotong harga (*undercut*).

E-Commerce

E-Commerce merupakan proses pembelian, penjualan, pemasaran, pelayanan produk, jasa atau informasi melalui berbagai jaringan komputer (O'Brien & Marakas, 2011). Pradana (2015) mengklasifikasikan Bisnis *E-Commerce* di Indonesia sebagai berikut:

Tabel 1. Klasifikasi Bisnis *E-Commerce* di Indonesia

Jenis Website	Penjelasan	Contoh (di Indonesia)
E-Commerce		
<i>Listing</i> / iklan baris	Berfungsi sebagai sebuah platform yang di mana para individu dapat memasang barang jualan mereka secara gratis. Pendapatan diperoleh dari iklan <i>premium</i> . Jenis iklan baris seperti ini cocok bagi penjual yang menjual barang kuantitas kecil	OLX, berniaga.com

<i>Online Marketplace</i>	Ini adalah model bisnis dimana website yang bersangkutan tidak hanya membantu mempromosikan barang dagangan saja, tapi juga memfasilitasi transaksi uang secara <i>online</i> . Seluruh transaksi <i>online</i> harus difasilitasi oleh website yang bersangkutan	tokopedia.com bukalapak.com shopee.co.id
<i>Shopping Mall</i>	Model bisnis ini mirip dengan <i>marketplace</i> , tapi penjual yang bisa berjualan disana haruslah penjual atau <i>brand</i> ternama karena proses verifikasi yang ketat.	blibli.com zalora.com
<i>Toko online</i>	Model bisnis ini cukup sederhana, yakni sebuah toko <i>online</i> dengan alamat website (domain) sendiri di mana penjual memiliki stok produk dan menjualnya secara <i>online</i> kepada pembeli	lazada.co.id bhinneka.com
<i>Toko online di media sosial</i>	Banyak penjual di Indonesia yang menggunakan situs media sosial seperti Facebook, Twitter dan Instagram untuk mempromosikan barang dagangan mereka	Siapapun yang Berjualan dengan media sosial
<i>Jenis-jenis website crowdsourcing dan crowdfunding</i>	Website dipakai sebagai platform untuk mengumpulkan orang-orang dengan skill yang sama atau untuk penggalangan dana secara <i>online</i> .	kitabisa.com wujudkan.com

Sumber: Pradana (2015)

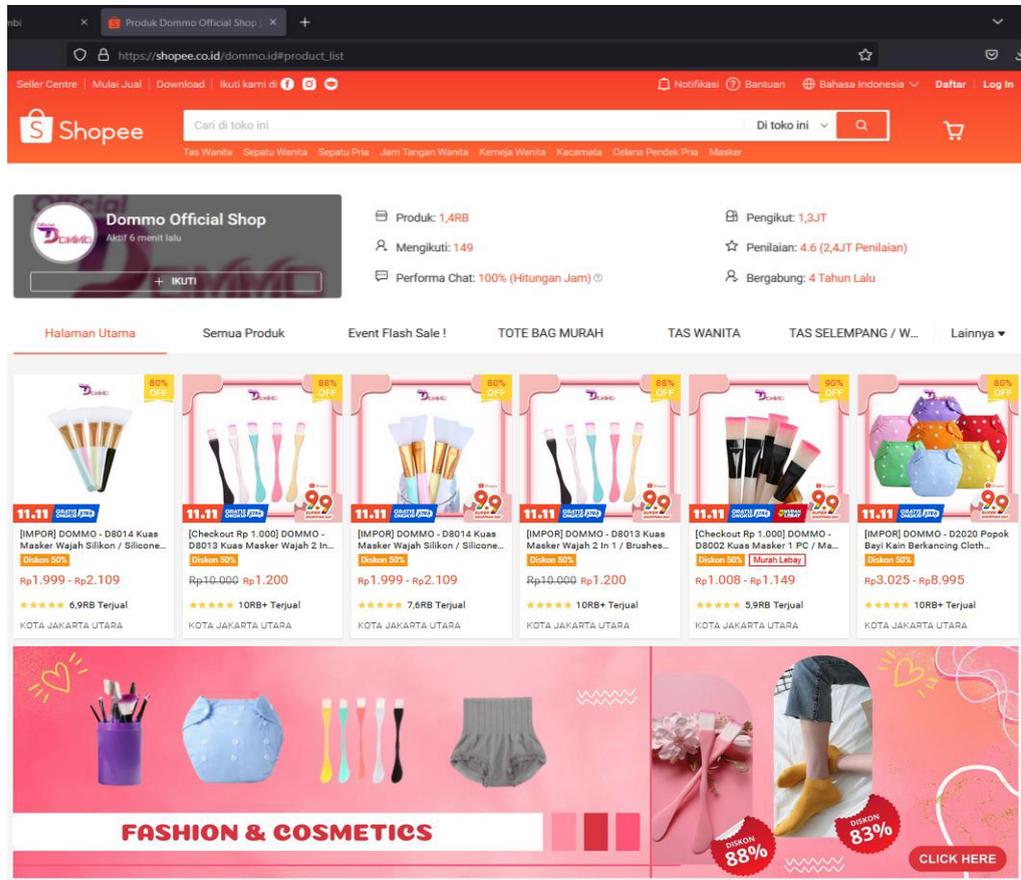
Online Marketplace

Online Marketplace merupakan media *online* berbasis internet (*web based*) tempat melakukan kegiatan bisnis dan transaksi antara pembeli dan penjual (Sofiani & Nurhidayat, 2019). Salah satu contoh *online marketplace* yang ada di indonesia yaitu shopee.com.

Shopee merupakan salah satu *marketplace* yang pertama kali dijalankan oleh Garena yang saat ini berubah nama menjadi SEA Group. Garena sendiri mengusung *mobile marketplace* Bisnis C2C (*customer to customer*). Pada Desember 2015 shopee resmi diperkenalkan di Indonesia di bawah naungan PT Shopee International Indonesia. Produk yang ditawarkan di *marketplace* Shopee banyak variasinya, antara lain produk Kecantikan, Pakaian Pria, Pakaian Wanita, Handphone & *Accesories*, Komputer & *Accesories*, Perlengkapan Rumah, Elektronik, Makanan & Minuman, Pulsa, Tagihan & Tiket, *Fashion* Muslim, *Fashion* Bayi & Anak, Ibu & Bayi, Tas Pria & Wanita, Kesehatan, Fotografi, Olahraga, *Voucher*, Buku & Alat Tulis, Serba Serbi, Sepatu Wanita & Pria,

Souvenir & Pesta, Jam Tangan, Hobi & Koleksi, dan masih banyak lagi (Sulistiyawati & Widayani, 2020).

Dommo



Gambar 1. Toko Online Dommo di Shopee

Dommo merupakan toko *online* yang menjual ulang (*reseller*) berbagai produk *brand* lokal maupun impor di *online marketplace* shopee. Dommo sendiri berfokus pada penjualan produk seperti sepatu, *fashion*, kosmetik dan sebagainya. Sebagai *reseller* dommo sangat bergantung pada berbagai produsen produk (konveksi) seperti Acnes, BioAqua, Wardah dan lain-lain. Dommo memahami bahwa konveksi-konveksi tersebut juga menjual produk-produknya pada *reseller* lain, sehingga menciptakan suatu persaingan. Salah satu strategi usaha yang dilakukan dommo untuk mempertahankan minat pembelian konsumen adalah dengan menetapkan harga yang dapat bersaing. Sebelumnya dommo memonitor harga dari produk pesaing terlebih dahulu sebagai dasar untuk mengambil keputusan pada harga yang akan ditetapkannya.

2.2 Sistem

Whitten & Bentley (2007) menyatakan sistem adalah kumpulan komponen yang saling berhubungan atau terintegrasi bersama untuk mencapai

satu tujuan yang diinginkan. Sistem informasi adalah pengaturan orang, data, proses, dan teknologi informasi yang berinteraksi untuk mengumpulkan, memproses, menyimpan, dan menyediakan hasil informasi yang diperlukan untuk mendukung suatu organisasi.

Menurut Putra & Febriani (2013) *monitoring* adalah pemantauan yang dapat memberikan informasi tentang status dan kecenderungan bahwa pengukuran dan evaluasi yang diselesaikan berulang dari waktu ke waktu. *Monitoring* adalah proses pengumpulan dan analisis informasi berdasarkan indikator yang ditetapkan secara sistematis dan kontinu tentang kegiatan program sehingga dapat dilakukan tindakan koreksi untuk penyempurnaan program kegiatan itu selanjutnya (Hendini, 2016). Peran *monitoring* adalah untuk mengenali dan mengevaluasi perkembangan yang terjadi akibat tindakan yaitu mengenali apakah pelaksanaan tindakan sesuai dengan rencana tindakan dan apakah telah terjadi peningkatan dengan adanya tindakan (Widayanti, 2008).

2.3 Aplikasi Mobile

Menurut Irsan (2015) Aplikasi *Mobile* adalah perangkat lunak yang berjalan pada perangkat mobile seperti *smartphone* atau tablet PC, Aplikasi *Mobile* juga dikenal sebagai aplikasi yang dapat diunduh dan memiliki fungsi tertentu sehingga menambah fungsionalitas dari perangkat *mobile* itu sendiri. Aplikasi *mobile* merupakan aplikasi yang dibuat untuk platform khusus berbasis *mobile* (perangkat bergerak) dengan tujuan tertentu untuk membantu pengguna sesuai dengan jenis aplikasi yang disediakan (Purnamasari & Panjaitan, 2020).

WhatsApp merupakan teknologi *Instant Messaging* seperti SMS dengan berbantuan data internet berfitur pendukung yang lebih menarik, WhatsApp Messenger memiliki berbagai fitur yang dapat digunakan untuk berkomunikasi dengan bantuan layanan internet. Fitur-fitur yang terdapat dalam WhatsApp yaitu *Gallery* untuk menambahkan foto, *Contact* untuk menyisipkan kontak, *Camera* untuk mengambil gambar, *Audio* untuk mengirim pesan suara, *Maps* untuk mengirimkan berbagai koordinat peta, bahkan *Document* untuk menyisipkan file berupa dokumen (Jumiatmoko, 2016). Berikut beberapa kelebihan dari aplikasi WhatsApp menurut Rusni (2017), yaitu:

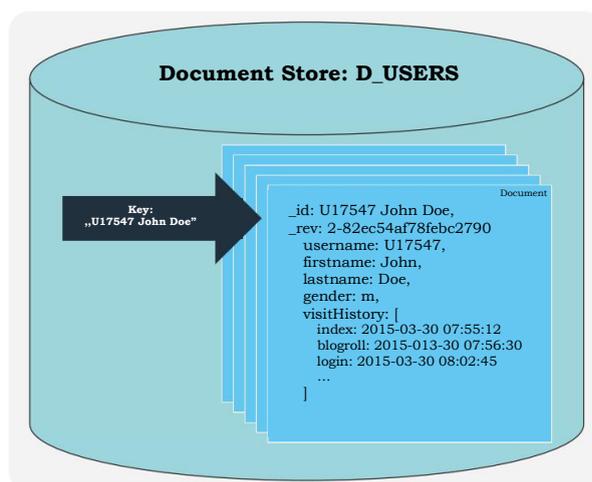
1. Tidak memerlukan uang untuk memasang aplikasi whatsapp ditelepon pintar.
2. Dapat mengirim pesan, gambar, video, audio dan pesan suara dengan mudah.
3. Dapat berkomunikasi dengan orang lain maksimal 50 (512) orang dalam satu chat group.
4. Mengkonsumsi data lebih kecil dari aplikasi-aplikasi lain.

2.4 Web scraping

Menurut Turland (2010) *Web scraping* adalah proses pengambilan sebuah dokumen semi-terstruktur dari internet, umumnya berupa halaman-halaman web dalam bahasa markup seperti HTML atau XHTML, dan menganalisis dokumen tersebut dengan tujuan untuk mengambil data tertentu yang dapat digunakan untuk kepentingan lain. *Web scraping* sering dikenal sebagai *screen scraping*, *Web scraping* tidak dapat dimasukkan dalam bidang *data mining* karena *data mining* menyiratkan upaya untuk memahami pola semantik atau tren dari sejumlah besar data yang telah diperoleh. Aplikasi *web scraping* (juga disebut *intelligent*, *automated*, atau *autonomous agents*) hanya fokus pada cara memperoleh data melalui pengambilan dan ekstraksi data dengan ukuran data yang bervariasi (Josi et al., 2014).

2.5 NoSQL

NoSQL (*Not only SQL*) merupakan *database* yang mendukung *Big Data* yang tidak bergantung pada skema database yang tetap, untuk itu model/tabel database tidak saling terhubung (Meier & Kaufmann, 2019). MongoDB adalah *open-source database* berbasis dokumen dan merupakan NoSQL database yang paling sering digunakan (Punia & Aggarwal, 2014). Pada MongoDB yang ada hanyalah koleksi dan dokumen, dokumen yang terdapat dalam mongoDB dapat memiliki atribut yang berbeda dengan dokumen lain walaupun berada dalam satu koleksi (Farozi, 2019). MongoDB menawarkan kinerja cepat, ketersediaan besar, dan skalabilitas yang mudah (Bhugul, 2015).



Gambar 2. Ilustrasi *Document* (Meier & Kaufmann, 2019)

Meier & Kaufmann (2019) Menyatakan *Document store database* atau *database* berbasis dokumen merupakan sistem manajemen *database* yang memiliki properti sebagai berikut:

1. Menggunakan metode penyimpanan *key-value*. *Value* dapat disimpan pada setiap *key* yang ada.
2. Kumpulan data *key-value* yang disimpan dinamakan *document*, dan dapat diidentifikasi menggunakan *key* tadi.

2.6 Bahasa Pemrograman

Bahasa pemrograman, atau sering diistilahkan juga dengan bahasa komputer atau bahasa pemrograman komputer, adalah instruksi standar untuk memerintah komputer. Bahasa pemrograman ini merupakan suatu himpunan dari aturan *sintaks* dan semantik yang dipakai untuk mendefinisikan program komputer. Bahasa ini memungkinkan seorang programmer dapat menentukan secara persis data mana yang akan diolah oleh komputer, bagaimana data ini akan disimpan/diteruskan, dan jenis langkah apa secara persis yang akan diambil dalam berbagai situasi (Saragih, 2016).

JavaScript

Menurut Alamsyah (2003) JavaScript adalah bahasa yang berbentuk kumpulan skrip yang pada fungsinya berjalan pada suatu dokumen HTML, sepanjang sejarah internet bahasa ini adalah bahasa skrip pertama untuk web. Bahasa ini adalah bahasa pemrograman untuk memberikan kemampuan tambahan terhadap bahasa HTML dengan mengizinkan pengeksekusian perintah perintah di sisi *user*, yang artinya di sisi *browser* bukan di sisi *server web*.

JavaScript yang pada mulanya hanyalah bahasa pemrograman sisi-klien yang hanya berfungsi sebagai utilitas dan pelengkap untuk membangun sebuah website tetapi seiring berjalannya waktu dan pembaruan maka sebetulnya JavaScript sudah tidak bisa hanya dianggap sebagai utilitas dan pelengkap saja (Prastyo, 2022).

Bahasa ini berada di masa yang disebut banyak orang sebagai kebangkitan, yang disebabkan oleh kecanggihan aplikasi berbasis *browser* yang meningkat pesat dan meningkatnya popularitas JavaScript di *server*, berkat Node.js (Uzayr et al., 2019). Popularitas ini juga dibantu oleh npm, Qiu et al. (2018) menyatakan npm berfungsi sebagai *repository package (library/module)* perangkat lunak berbasis JavaScript, dan menampung lebih dari 650.000 *package* JavaScript menjadikannya ekosistem perangkat lunak terbesar, dengan jutaan paket diinstal dari *repositori* npm setiap harinya.

Node.JS

Node.JS adalah sistem perangkat lunak yang didesain untuk pengembangan aplikasi web. Node.JS dapat juga disebut sebagai runtime *environment*. Aplikasi ini ditulis dalam campuran Bahasa C++ dan juga JavaScript, mempunyai model *event driven* (basis *event*) dan *asynchronous* I/O. Tidak seperti kebanyakan bahasa JavaScript yang dijalankan pada *web browser*, Node.JS dieksekusi sebagai aplikasi server. Node.JS dapat berjalan di *server* karena dukungan dari V8 Engine buatan Google dan beberapa modul bawaan yang terintegrasi seperti modul *http*, modul *filesystem*, modul *security* dan beberapa modul penting lainnya (Fajrin, 2017). Salah satu kelebihan dari Node.js adalah adanya teknik *non-blocking*. Teknik *non-blocking* adalah teknik dengan Node.js akan melakukan eksekusi secara independen. Node.js akan mengeksekusi sebuah operasi tanpa harus menunggu operasi sebelumnya selesai dieksekusi (Luqman, 2016).

Package

npm adalah *package* manager yang populer digunakan di ekosistem JavaScript dan npm juga terpasang otomatis disaat melakukan peng-*install*-an Node.js. Salah satu fitur utama npm yaitu membantu pengguna meng-*install* lebih dari satu *package/module* secara bersamaan (Oberschweiber, 2016). *Package* merupakan kumpulan file yang dibutuhkan dari sebuah *module*. *Module* merupakan JavaScript *libraries* (kumpulan fungsi JavaScript) yang dapat digunakan untuk membatu/mempersingkat proses dalam membangun suatu sistem. Umumnya sebuah *module* dikelola oleh komunitas dan dipublikasikan ke *repositori online* sebagai *software open source*.

Berikut daftar *module* yang penulis gunakan untuk membantu membangun sistem ini:

1. Puppeteer, adalah *library* yang menyediakan fungsi untuk mengontrol *browser* Chrome/Cromium. Penulis menggunakan *library* ini untuk membuat fungsi *web scraping*
2. csv parser, adalah *library* yang menyediakan fungsi untuk mengonversikan file csv ke bentuk JSON. Penulis menggunakan *library* ini untuk membuat fungsi *import dataset* yang berextensi .csv
3. canvas-node, adalah *library* yang menyediakan fungsi untuk membuat objek 2 dimensi menggunakan kode program. Penulis menggunakan *library* ini untuk menampilkan data dari *database* ke bentuk file gambar/pdf.

4. whatsapp-web.js, adalah *library* yang menyediakan fungsi untuk dapat mengakses fitur yang ada di WhatsApp Web. Penulis menggunakan *library* ini untuk menghubungkan sistem ke aplikasi WhatsApp.
5. Mongoose, adalah *library* yang menyediakan fungsi untuk *Object Data Modeling* pada MongoDB. Penulis menggunakan *library* ini untuk mengatur hubungan dan data pada sistem dan *database*.

2.7 Software Development Life Cycle (SDLC)

Menurut Dwanoko (2016) *Software Development Life Cycles* (SDLC) adalah tahapan-tahapan pekerjaan yang dilakukan oleh analis sistem dan *programmer* dalam membangun sistem informasi. Terdapat beberapa model SDLC yang dapat digunakan, semuanya mempunyai kelebihan dan kekurangan masing-masing pada tiap tahapannya, hal yang paling penting yaitu mengenali type pelanggan/*customer* dan memilih menggunakan model SDLC yang sesuai dengan karakter pelanggan dan sesuai dengan karakter pengembang (Hasanah & Untari, 2020). Sabele & Dani (2012) menjabarkan hasil perbandingan antara model-model SDLC berdasarkan fitur-fitur yang dimilikinya dalam bentuk tabel berikut ini:

Tabel 2. Perbandingan Model *Software Development Life Cycles*

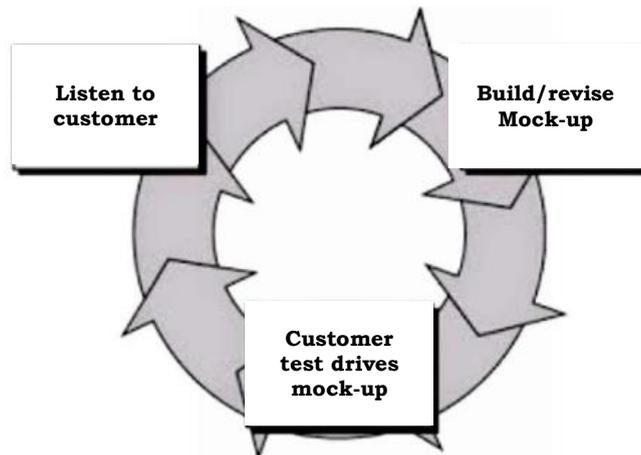
Model/ Features	Waterfall	V-shape	CMM	RUP	Prototype	Incremental	Spiral	RAD	JAD	Agile
Requirement specification	Beginning	Beginning	At second level	Beginning	Frequently changed	Beginning	Beginning	Time-box released	Prototyped	Frequently changed
Understanding requirements	Well understood	Easily understood	Easily understood	Difficult to understand	Not well understood	Well understood	Well understood	Well understood	Easily understandable	Well understood
Cost	Low	Expensive	High	Expensive	High	Low	Expensive	Low	Expensive	Veri High
Guarantee of success	Low	High	High	Not guaranteed	Good	High	High	Good	Low but for long period	Veri High
Resource control	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	No
Cost control	Yes	Yes	Varies	Yes	No	No	Yes	Yes	Yes	Yes
Simplicity	Simple	Intermediate	Intermediate	Simple and clear	Simple	Simple	Intermediate	Intermediate	Very simple	Complex
Risk involvement	High	Low	Varies acc to level	Critical risk in the early stages	low	Easily manage	Low	Very Low	Varies	Reduced
Expertise required	High	Medium	Varies acc to level	Yes	Medium	High	High	Medium	High	Veri high
Change incorporated	Difficult	Difficult	Medium	Easy	Easy	Easy	Easy	Easy	Medium	Difficult
Risk analysis	Only at beginning	Yes	Yes	Yes	No risk analysis	No risk analysis	Yes	Low	Yes	Yes
User involvement	Only at the beginning	Only at the beginning	Only at the beginning	At the beginning and the last phase	High	Intermediate	High	Only at the beginning	In design and development	High
Overlapping phases	No such phase	No	No	Yes	Yes	No	Yes	No	No	Yes
Flexibility	Rigid	Little flexible	Highly flexible	considerable	Highly flexible	Less flexible	Flexible	High	Flexible	Highly flexible
Maintenance	Least glamorous	Least	Typical	Promote maintainability	Routine maintenance	promote maintainability	Typical	Easily maintained	Rigorously at all times	Promote maintainability
Integrity and security	vital	Limited	Limited	Veryimportant	Weak	Robust	High	Vital	Limited	Demonstrable
Reusability	Limited	To some extend	Yes	Supports reusability of exiting classes	Weak	Yes	Yes	Some extend	Limited	Usecase reuse
Interface	Minimal	Minimal	Crucial	User interface	Crucial	Crucial	Crucial	Minimal	Crucial	Model driven
Documentation & Training required	Vital	Yes	Yes	Yes	Weak	Yes	Yes	Limited	Limited	Yes
Time Frame	Long	Acc to project size	Quite Long	Short time frame	Short	Very long	Long	Short	Medium	Lest possible

Sumber: Sabele & Dani (2012)

Dari perbandingan model yang dijabarkan di atas, penulis memilih model *prototype* sebagai model yang akan digunakan untuk penelitian ini. Alasan utama penulis menggunakan model ini karena pengguna sistem tidak dapat menjelaskan dan mendefinisikan secara keseluruhan tentang gambaran dari sistem yang akan dibangun. Alasan ini sesuai dengan fitur yang disediakan dari model *prototype* yaitu :

1. *Requirement Spesification – Frequently Changed*, pengguna sistem dapat menambah, mengubah atau mengurangi spesifikasi sistem yang dibutuhkan.
2. *Understanding requirements – Not well understood*, pengguna sistem tidak mampu menjelaskan dan mendefinisikan secara keseluruhan tentang kebutuhan yang diinginkannya

Prototype



Gambar 3. Model *Prototype* (Ali, 2019)

Menurut Andrew & Setiawan (2019) *prototype* merupakan suatu proses pembuatan aplikasi yang bersifat berulang dan dengan perencanaan yang cepat dimana terdapat umpan balik yang memungkinkan terjadinya perulangan dan perbaikan aplikasi sampai aplikasi tersebut memenuhi kebutuhan dari pengguna. Menurut Hasanah & Untari (2020) tahapan pengembangan model *prototype* yaitu:

1. Mendengarkan pelanggan, pada tahap ini dilakukan pengumpulan kebutuhan dari sistem dengan cara mendengar keluhan dari pelanggan. Untuk membuat suatu sistem yang sesuai kebutuhan, maka harus diketahui terlebih dahulu bagaimana sistem yang sedang berjalan untuk kemudian mengetahui masalah yang terjadi.
2. Merancang dan Membuat *Prototype*, pada tahap ini, dilakukan perancangan dan pembuatan *prototype* sistem. *Prototype* yang dibuat

disesuaikan dengan kebutuhan sistem yang telah didefinisikan sebelumnya dari keluhan pelanggan atau pengguna.

3. Uji coba Pada tahap ini, *Prototype* dari sistem diuji coba oleh pelanggan atau pengguna. Kemudian dilakukan evaluasi kekurangan-kekurangan dari kebutuhan pelanggan. Pengembangan kemudian kembali mendengarkan keluhan dari pelanggan untuk memperbaiki *Prototype* yang ada.

Ali (2019) menjabarkan kelebihan dan kekurangan model *prototype* sebagai berikut :

1. Kelebihan

- Memberikan model yang dapat digunakan kepada pengguna di awal proses, memungkinkan penilaian awal dan meningkatkan kepercayaan pengguna.
- Pengembang mendapatkan pengalaman dan wawasan dengan mengembangkan prototipe di sana dengan menghasilkan implementasi persyaratan yang lebih baik.
- Model *prototype* berfungsi untuk memperjelas persyaratan, yang tidak jelas, sehingga mengurangi ambiguitas dan meningkatkan komunikasi antara pengembang dan pengguna.
- Ada keterlibatan besar pengguna dalam pengembangan perangkat lunak. Oleh karena itu, persyaratan pengguna dipenuhi sejauh mungkin.
- Membantu mengurangi risiko yang terkait dengan perangkat lunak.

2. Kekurangan

- Jika pengguna tidak puas dengan *prototype* yang dikembangkan, maka prototipe yang baru dapat dikembangkan. Proses ini berlangsung hingga *prototype* sempurna dikembangkan. Dengan demikian, model ini memakan waktu dan mahal.
- Pengembang kehilangan fokus tujuan *prototype* yang sesungguhnya dan karenanya, dapat berkompromi dengan kualitas perangkat lunak. Misalnya, pengembang dapat menggunakan beberapa algoritma yang tidak efisien atau bahasa pemrograman yang tidak tepat saat mengembangkan *prototype*.
- *Prototype* dapat menyebabkan harapan yang salah bagi pengguna. Misalnya, sebuah situasi yang dibuat di mana pengguna percaya bahwa pengembangan sistem telah selesai padahal sesungguhnya belum.

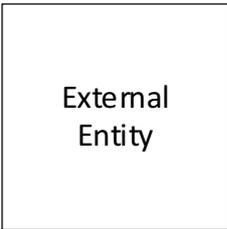
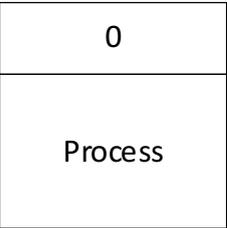
- Tujuan utama dari *prototype* adalah pengembangan cepat, sehingga desain sistem dapat menyulitkan karena dikembangkan secara seri tanpa mempertimbangkan integrasi semua komponen lainnya.

2.8 Data Flow Diagram

Data Flow Diagram adalah alat representasi grafis yang menggunakan beberapa simbol sederhana untuk mengilustrasikan aliran data antara entitas *eksternal*, aktivitas pemrosesan dan elemen penyimpanan data (O'Brien & Marakas, 2011). DFD merupakan sebuah teknik analisis yang digunakan untuk menggambarkan aliran masukan (*input*) dalam sebuah sistem yang diolah oleh proses dan menghasilkan suatu keluaran (*output*), Diagram ini menggambarkan apa yang terjadi dalam sebuah sistem (Hasanah & Untari, 2020). DFD digunakan untuk menunjukkan bagaimana data mengalir melalui urutan langkah pemrosesan (Sommerville, 2011).

Menurut Hasanah & Untari (2020) berikut adalah penjelasan komponen-komponen DFD yang penulis tuliskan dalam bentuk tabel

Tabel 3. Komponen-komponen *Data Flow Diagram*

Simbol	Penjelasan
	disimbolkan dalam bentuk persegi panjang, yang mewakili entity luar dimana sistem berkomunikasi. Biasanya notasi ini melambangkan orang atau kelompok orang misalnya organisasi diluar sistem, grup, departemen, perusahaan pemerintah, dan berada di luar kontrol sistem yang dimodelkan. Pada sejumlah kasus dapat merupakan sistem lain, sebagai contoh adalah <i>entity</i> : pegawai, mahasiswa, peserta didik dll.
	Komponen proses menggambarkan suatu transformasi <i>input</i> menjadi <i>output</i> . Terdapat beberapa hal yang harus diperhatikan tentang komponen proses dalam diagram alur data antara lain adalah: 1) Setiap komponen proses harus memiliki input dan output; 2) proses dapat dihubungkan dengan komponen <i>terminator</i> , <i>data store</i> atau alur data;
	<i>Data store</i> digunakan untuk memodelkan kumpulan data atau paket data. Penyimpanan data kadang kala didefinisikan sebagai suatu mekanisme diantara dua proses yang dibatasi oleh jangka waktu tertentu. <i>Data store</i> dapat berupa file atau basis data yang tersimpan

dalam unit penyimpanan seperti disket, harddisk. Alur data digunakan untuk menerangkan perpindahan data/paket data dari satu bagian ke bagian lainnya. Alur data dapat berupa kata, pesan, formulir atau informasi. Terdapat beberapa hal Yang harus diperhatikan dalam pendefinisian *data store* anatra lain ialah : 1) Alur data dari proses menuju data store. Hal ini berarti bahwa data store berfungsi sebagai tujuan atau tempat penyimpanan dari suatu proses (*proses write*).; 2) Alur data dari *data store* ke proses. Hal ini berarti *data store* berfungsi sebagai sumber atau suatu proses memerlukan data (*proses read*); 3) Alur data dari proses menuju data store dan sebaliknya. Hal ini berarti alur data berfungsi Sebagai sumber dan tujuan (*proses update*)

—Data Flow—▶

Alur data digunakan untuk menerangkan perpindahan data atau paket data dari satu bagian ke bagian lainnya. Alur data menunjukkan aliran data yang dapat berupa masukkan untuk sistem atau hasil proses sistem.

Sumber : Hasanah & Untari (2020)

2.9 Pengujian Perangkat Lunak

Pengujian Perangkat lunak bertujuan untuk memastikan bahwa program *computer* menjalankan apa yang seharusnya dilakukan dan sebaliknya, memastikan program agar tidak melakukan hal yang tidak diharapkan (Hasanah & Untari, 2020).

Pengujina Fungsional

Pengujian *software* sangat diperlukan untuk memastikan *software* yang sudah atau sedang dibuat dapat berjalan sesuai dengan fungsionalitas yang diharapkan (Hidayat & Putri, 2019). Hasanah & Untari (2020) menjabarkan kelebihan dan kekurangan *Black Box* dan *White Box* sebagai berikut:

Tabel 4. Kelebihan dan Kekurangan *Black Box* dan *White Box*

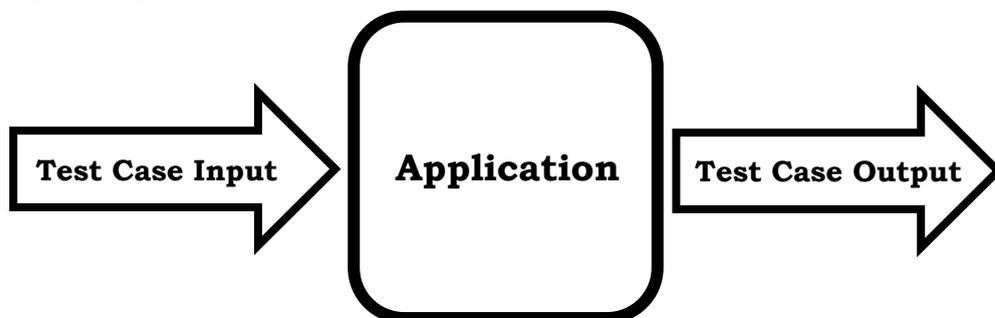
Nama Pengujian	Kelebihan	Kekurangan
<i>Black box testing</i>	<ul style="list-style-type: none"> • Efisien untuk segmen kode besar. • Akses kode tidak diperlukan • Pemisahan antara perspektif pengguna dan pengembang 	<ul style="list-style-type: none"> • Cakupan terbatas karena hanya sebagian kecil dari skenario pengujian yang dilakukan • Pengujian tidak efisien karena keberuntungan tester dari

<i>White box testing</i>	<ul style="list-style-type: none"> • Efisien dalam menemukan kesalahan dan masalah • Diperlukan pengetahuan tentang internal perangkat lunak yang sedang diuji bermanfaat untuk pengujian menyeluruh • Memungkinkan menemukan kesalahan tersembunyi • Membantu mengoptimalkan kode 	<p>pengetahuan tentang perangkat lunak internal</p> <ul style="list-style-type: none"> • Membutuhkan pengetahuan tingkat tinggi dari perangkat lunak internal yang sedang diuji • Membutuhkan akses kode
--------------------------	--	--

Sumber: Hasanah & Untari (2020)

Dari informasi tabel diatas penulis memilih pengujian perangkat lunak *Black box testing* pada pengujian fungsionalitas sistem karena efisiensi yang ditawarkannya pada segmen kode besar. Selain fungsionalitas, penulis akan melakukan pengujian pada efisiensi sistem dengan menghitung berapa waktu yang dibutuhkan sistem untuk melakukan respon dari setiap *query* perintah yang digunakan.

Menurut Mustaqbal et al (2015) *Black Box Testing* berfokus pada spesifikasi fungsional dari perangkat lunak. *Black Box Testing* atau yang sering dikenal dengan sebutan pengujian spesifikasi fungsional merupakan metode pengujian untuk mengetahui apakah fungsi-fungsi masukan dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Dalam pengujian ini, penguji menyadari apa yang harus dilakukan oleh program tetapi tidak memiliki pengetahuan tentang bagaimana melakukannya (Hasanah & Untari, 2020). Ali (2019) menjelaskan proses *Black box testing* yang mana penguji dalam hal ini, memiliki satu set nilai *input* dan hasil yang diinginkan masing-masing. Pada saat memberikan masukan, jika *output* sesuai dengan hasil yang diinginkan, program diuji 'oke', dan bermasalah sebaliknya.



Gambar 4. Ilustrasi Proses *Black Box Testing* (Sumber: Ali, 2019)

Pengujian Efisiensi

Efficiency mencakup berapa waktu yang dibutuhkan sistem untuk melakukan respon dan bagaimana resources dimanfaatkan secara efisien artinya dalam software berbasis web dapat diketahui berapa lama waktu untuk *me-load* halaman web dan bagaimana *source code* diprogram secara efisien (Padayachee et al., 2010). Sedangkan pada software berbasis pesan berapa lama waktu yang dibutuhkan sistem untuk merespon pesan yang dikirim pengguna.

Berdasarkan riset yang dikemukakan oleh Google melalui layanan Think with Google waktu yang direkomendasikan untuk *me-load* halaman website adalah paling lama 3 detik karena berdasarkan hasil survei mereka, 53% pengguna akan meninggalkan halaman website apabila waktu *load* yang dibutuhkan melebihi 3 detik (Think with Google, 2017).

2.10 Penelitian Terdahulu

Berikut daftar penelitian terdahulu yang memiliki karakteristik menyerupai penelitian yang akan dilakukan:

Tabel 5. Penelitian Terdahulu

No	Judul Penelitian dan Penulis	Hasil Penelitian
1	Rancang Bangun “Telegram Bot API” untuk Layanan Sistem Informasi Akademik di Unwaha Menggunakan Metode Long Polling (Maulayya et al., 2019)	Dengan menggunakan aplikasi sistem informasi berbasis BOT Telegram, mahasiswa dapat lebih mudah dan cepat didalam mendapatkan informasi akademik.
2	Pengembangan Sistem Informasi Manajemen Persediaan Barang berbasis Web (Tirzasari et al., 2022)	Sistem Informasi Manajemen Persediaan barang dapat membantu aktivitas pengelolaan persediaan barang pada BYW Studio.
3	Implementasi Telegram Bot untuk Monitoring Mikrotik Router (Hakim & Nugroho, 2019)	Dengan menggunakan bot telegram dapat mempermudah admin atau teknisi dalam mendapatkan informasi konektivitas jaringan secara <i>realtime</i> , dalam mendapatkan informasi user yang <i>login</i> dan <i>logout</i> kedalam jaringan dan dalam mendapatkan informasi kondisi fisik mikrotik router.

- 4 Perancangan Sistem Informasi Monitoring Laporan Penjualan Multi Cabang Berbasis Web Dengan Metode *Prototype* Studi Kasus Toko King Cellular (Widiyanto & Wicaksono, 2022). Dengan adanya sistem ini proses pemeriksaan hasil laporan penjualan dapat dilakukan kapan dan dimanapun dengan mudah oleh pemilk toko dan admin cabang toko.
- 5 Implementasi WhatsApp Gateway pada Apikasi Manajemen Keuangan Pondok Pesantren Tanwirul Qulub Lamongan (Azizah & Prisma, 2022) Aplikasi manajemen keuangan berbasis website dengan fitur *notification reminder* menggunakan WhatsApp gateway dapat menjadi solusi bagi bendahara pondok pesantren Tanwirul Qulub dalam mengelola keuangan terutama permasalahan pembayaran iuran santri yang tidak disiplin. Dengan menggunakan fitur *notification reminder*, bendahara pondok pesantren dapat mengirimkan pesan tagihan santri dan dokumen rinciannya berupa *link* kepada wali santri.
- 6 Aplikasi Web Scraping Deskripsi Produk (Setiawan et al., 2020) Telah dibangun Sistem Informasi *Web Scraping* berbasis Web dengan menggunakan Framework Laravel 5.7.21 untuk mempermudah pelaku penjual *online* dalam mencari barang dan mengisi deskripsi produk dengan mudah dan cepat. Pelaku penjual *online* juga dapat mengelola produk pada *e-marketplace* Tokopedia, Bukalapak, Lazada, Blanja.com, Zilingo, Zalora, Kriya, Amazon, dan Alibaba.

Sumber: Penelitian Terdahulu