



## Web Scraping pada Web Media Digital untuk Membangun Aplikasi Android

Asri Mulyani<sup>1</sup>, Dede Kurniadi<sup>2</sup>, Ikbal Lukmanul Hakim<sup>3</sup>

Jurnal Algoritma  
Sekolah Tinggi Teknologi Garut  
Jl. Mayor Syamsu No. 1 Jayaraga Garut 44151 Indonesia  
Email : [jurnal@itg.ac.id](mailto:jurnal@itg.ac.id)

<sup>1</sup>asrimulyani@itg.ac.id  
<sup>2</sup>dede.kurniadi@itg.ac.id  
<sup>3</sup>1606018@itg.ac.id

**Abstrak** – Perkembangan teknologi otomatisasi pada era saat ini sangat pesat, efisiensi waktu dalam melakukan pekerjaan menjadi sangat tinggi, tidak terkecuali dalam penyalinan informasi dari *website*, pengambilan data yang dilakukan secara konvensional akan memakan waktu yang lama serta banyak hal yang dilakukan, dengan memanfaatkan teknik *web scraping* pengambilan data atau penyalinan informasi dari *website* dilakukan secara otomatis yang mana efisiensi waktu menjadi sangat tinggi. Adapun salah satu pemanfaatan dari teknik *web scraping* adalah untuk membuat *REST-API* tanpa *database* dan diintegrasikan ke *mobile apps*. Tujuan dari penelitian ini untuk mengimplementasikan teknik *web scraping* untuk membangun aplikasi berbasis android. Adapun metodologi yang digunakan dalam penelitian ini adalah *Rational Unified Process (RUP)*, tahapan yang dilakukan dalam metodologi ini mulai dari *inception*, *elaboration construction*, sampai dengan *transition*. Hasil dari penelitian ini adalah berupa aplikasi media digital berbasis *android* sebagai *client*-nya dan *REST-API* sebagai *server*-nya.

**Kata Kunci** – *Android*; Media Digital; Penyalinan; *REST-API*; *Web Scraping*.

### I. PENDAHULUAN

Informasi ataupun data yang beredar di *website-website* internet sangatlah banyak, *Website* merupakan kumpulan halaman yang berisikan informasi dengan bentuk berupa *text*, *image*, *audio*, *video* dan animasi yang dapat diakses melalui internet [1], data dari *website* tersebut dapat dimanfaatkan untuk mengembangkan aplikasi berbasis *android*. Pada Umumnya dalam mengumpulkan data atau informasi dari *website* dilakukan dengan penyalinan secara konvensional sehingga memakan waktu yang cukup lama serta rentan ketidakakuratan data jika sumber mengalami perubahan, serta data yang dikumpulkan sulit untuk diintegrasikan dengan aplikasi berbasis *android*, untuk mengatasi masalah tersebut diperlukan suatu teknik penyalinan secara otomatis dari *website* serta dibangun ke dalam arsitektur aplikasi *back end*. Salah satu teknik untuk melakukan pengambilan data secara otomatis adalah dengan menggunakan teknik *web scraping*, *web scraping* adalah proses pengambilan sebuah dokumen semi-terstruktur dari internet, biasanya berupa *website* yang dibangun dengan bahasa markup seperti *HTML* atau *XHTML* yang bertujuan untuk mengambil data atau informasi dari halaman tersebut baik secara keseluruhan atau sebagian untuk digunakan bagi kepentingan lain [2].

Terdapat beberapa penelitian yang relevan dengan topik yang akan dibahas dan menjadi rujukan. Penelitian Afifa & Rizky mengenai implementasi teknik *web scraping*, penelitian ini membangun sistem dengan konsep *Resource Description Framework (RDF)* untuk data obat dengan menggunakan teknik *web scraping* untuk mengumpulkan data obatnya [3]. Penelitian Afif melakukan perancang sistem informasi untuk data *hotel* dan *travel agent*, dalam penelitian ini dibangun sebuah sistem untuk mencari harga dari jasa atau barang yang ditawarkan dari berbagai *web travel agent* dengan menggunakan teknik *web scraping* dan menggunakan

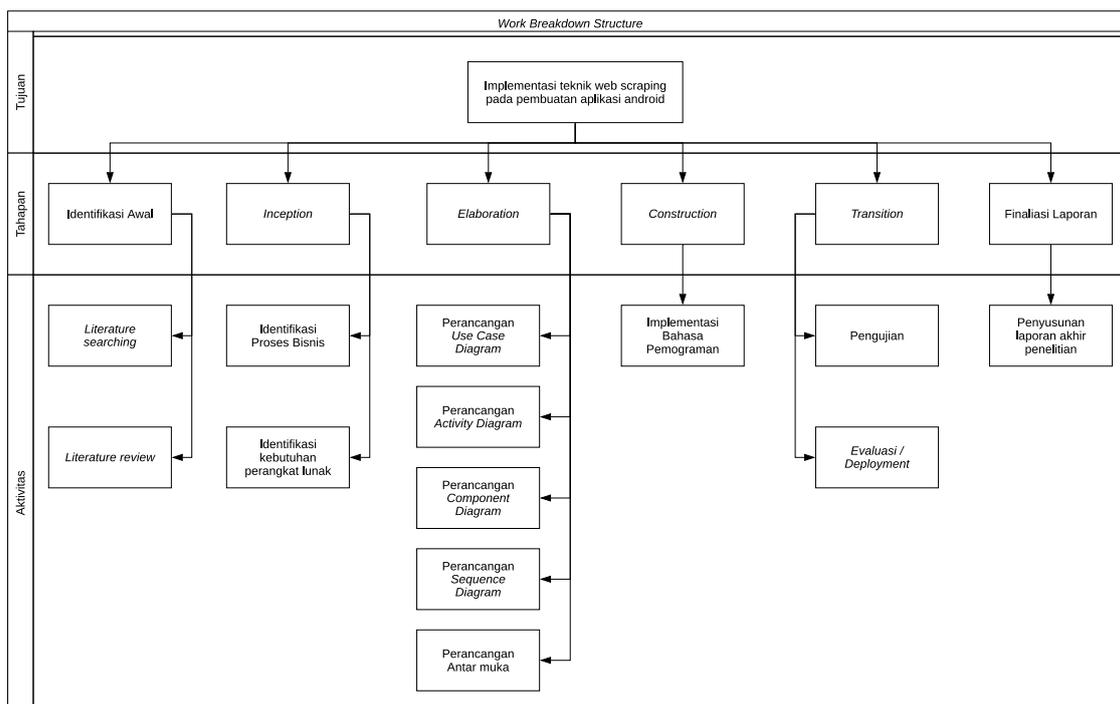
Metode *Breadth First Search* dalam pengumpulan datanya [4]. Penelitian Dede Kurniadi, M. Mesa Fauzi dan Asri Mulyani, dalam penelitian ini dilakukan pembuatan aplikasi untuk simulasi tes buta warna berbasis *android*, memberi kemudahan karena dapat digunakan di telepon genggam [5]. Penelitian Mochamad Syarif dan Yosep Septiana merancang aplikasi untuk membuat *curriculum vitae* dosen, penelitian ini membahas mengenai integrasi *web service rest-api* dari ristekdikti untuk membangun aplikasi *auto generate curriculum vitae* dosen di kampus STTGarut [6]. Penelitian Erwan Akse mengimplementasikan *REST-API system* akademik kampus UIN Suanan Gunung Djati, dalam penelitian ini membahas mengenai pembangunan aplikasi sistem akademik kampus dengan memanfaatkan *web service rest-api* dari kampus untuk di intergrasikan dengan aplikasi *mobile* yang dibangun [7].

Adpun permasalahan yang muncul dalam penelitian ini adalah sebagai berikut: (1) Melakukan ekstraksi data atau informasi dalam *website* dengan menggunakan teknik *web scraping*, (2) Mengintegrasikan hasil ekstraksi kedalam aplikasi berbasis *android*. Serta tujuan dari penelitian ini adalah mengimplemntasikan teknik *web scarping* untuk pembuatan aplikasi media digital berbasis *android* dan diharapkan pengetahuan yang dihasilkan bisa menjadi suatu pendekatan baru bagi para *developer* ataupun masyarakat informatika dalam mengembangkan mapunu membangun aplikasi berbasis *android*.

Terdapat batasan pada penelitian ini agar pembahsan yang dilakukan lebih terfokus, aplikasi yang dibangun menggunakan arsitekru *client-server*, *android* aplikasi sebagai aplikasi *client* dan aplikasi *backend* dengan arsitektur *REST-API* sebagai *server*-nya, *REST-API* merupakan salah satu dari desain arsitektur yang terdapat di dalam *API* dan salah satu metode untuk melakukan komunikasi yang menggunakan *protocol HTTP* untuk pertukaran data [8]. Dalam pembuatannya digunakan metodologi *Rational Unified Process (RUP)* dan *Unified Modeling Language (UML)* sebagai pemodelan sistemnya, adapun *website* media digital yang dijadikan sumber data pada aplikasi adalah *mojok.co*. Hasil dari penelitian ini berupa aplikasi media digital berbasis *android* dengan arsitektur basis data *client-server*.

## II. METODOLOGI PENELITIAN

Pada penelitian ini tujuan yang akan di capai adalah sebuah aplikasi media digital dengan mengimplentasikan teknik *web scraping*, metodologi yang digunakan dalam pembuatan aplikasi menggunakan metodologi *Rational Unified Process (RUP)*, RUP adalah sebuah proses pembangunan sistem meliputi seluruh *lifecycle* pembangunan perangkat lunak yang menyediakan suatu pendekatan untuk membantu tugas dan tanggung jawab suatu pembangunan organisasi. RUP diciptakan, dikembangkan, dan dikelola oleh *Rational Software*. Tujuannya adalah menghasilkan perangkat lunak berkualitas tinggi yang memenuhi kebutuhan pengguna dan dapat diprediksi penjadwalan dan biaya pengembangannya [9], dan pemodelannya menggunakan *Unified Modeling Language (UML)* adalah sebuah bahasa untuk memodelkan requirement, analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML sudah menjadi salah satu standar bahasa yang banyak digunakan didunia industri [10]. Dalam metodologi tersebut terdapa aktivitas-aktivitas yang dilakuk aktivitas tersebut dituangkan kedalam diagram *Work Breakdown Structure (WBS)* sebagaimana disajikan pada Gambar 1.



Gambar 1: *Work Breakdown Structure*

Adapun penjelasan dari gambar diagram WBS diatas adalah sebagai berikut:

- 1) Pada tahapan awal identifikasi awal, tahapan dimana melakukan *literature searching* dan *literature review* sebagai landasan dalam penelitian ini;
- 2) Pada tahapan kedua *inception*, tahapan dimana melakukan identifikasi terhadap proses bisnis dari aplikasi juga identifikasi dari kebutuhan aplikasinya;
- 3) Pada tahapan ketiga *elaboration*, tahapan dimana pembuatan desain berdasarkan data yang telah didapat dari hasil aktifitas sebelumnya, pembuatan desain tersebut menggunakan pemodelan UML, yang terdiri dari *use case diagram*, *activity diagram*, *component diagram*, *sequence diagram*, serta desain antarmuka aplikasi;
- 4) Pada tahapan keempat *construction*, tahapan dimana melakukan implementasi desain yang telah dibuat kedalam kode program;
- 5) Pada tahapan kelima *transition*, tahapan dimana melakukan pengujian terhadap aplikasi yang telah dibuat, apakah sudah sesuai dengan apa yang di harapkan juga melakukan evaluasi jika ada hal yang tidak sesuai;
- 6) Pada tahapan terakhir finaliasi laporan, tahapan dimana penyusunan laporan akhir penelitian.

### III. HASIL DAN PEMBAHASAN

#### A. Identifikasi Awal

Melakukan identifikasi awal mengenai topik yang akan diteliti, serta melakukan analisis dari *literature* yang menjadi acuan pada penelitian ini, hasil dari aktivitas tersebut disajikan pada Tabel 1.

Tabel 1: Kesenjangan Penelitian

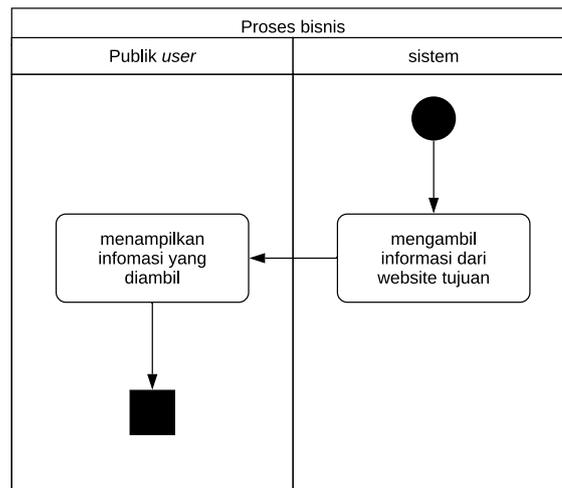
No	Penelitian	Cakupan Penelitian	Kesenjangan Penelitian
1	(Afifa,2018)	Pada penelitian tersebut hasil akhirnya adalah sebuah infrastruktur data dari sebuah obat dan penyakit untuk bisa digunakan sebagai ontologi, pengumpulan datanya menggunakan teknik <i>web scraping</i> .	Pada penelitian ini sama menggunakan teknik <i>web scarping</i> dalam pengumpulan data, produk yang dihasilkan dari penelitian ini berupa aplikasi bebasi android yang bisa digunakan <i>end user</i>
2	(Arif, 2020)	Pada penelitian tersebut membangun sebuah aplikasi berbasis <i>web</i> untuk mencari harga dari jasa atau barang yang di sediakan <i>travel agent</i> dengan teknik <i>web scraping</i> menggunakan metode <i>breadth first search</i> .	Pada penelitian ini membangun aplikasi berbasis <i>android</i> dalam pengumpulan data menggunakan teknik <i>web scraping</i> dan data hasil ekstraksi disajikan dalam bentuk <i>REST Server</i> ; tidak hanya menampilkan hasil pencarian tapi seluruh konten dari <i>web</i> sumber.
3	(Kurniadi, Fauzi, & Mulyani, 2016)	Pada penelitian tersebut membangun aplikasi berbasis android untuk melakukan tes buta warna, data yang diggunakan berasal dari internal aplikasi.	Pada penelitian ini hasil akhirnya aplikasi berbasis anroid content yang ada dalam aplikasi berasal dari luar aplikasi, hasil dari ekstraksi data dari <i>website</i> yang menjadi sumber dengan teknik <i>web scraping</i> .
4	(Septiana & Hidayat, 2020)	Pada penelitian tersebut dibuat aplikasi berbasis <i>web</i> untuk membuat <i>curriculum vitae</i> dosen secara otomatis dengan memanfaatkan <i>REST server</i> dari ristekdikti sebagai sumber data.	Pada penelitian ini membuat aplikasi android, sumber data dihasilkan dari ekstraksi data <i>website</i> dengan teknik <i>web scraping</i> dan di sajikan kedalam <i>REST serve</i> .
5	(Akse, 2017)	Pada penelitian tersebut membangun sistem akademik berbasis <i>android</i> dengan memanfaatkan <i>REST server</i> dari kampus sebagai sumber data.	Pada penelitian ini menghasilkan aplikasi <i>android</i> dengan menimplementasikan teknik <i>web scraping</i> untuk mengumpulkan data dan dikemas kedalam <i>REST server</i> untuk mengintegrasikannya.

## B. Perancangan Sistem

Adapun tahapan tahapan yang dilakukan dalam perancang sistem adalah sebagai berikut:

### 1) *Inception*

- a. Identifikasi proses bisnis, proses bisnis pada aplikasi yang akan dibangun ini hanya terdapa 1 role yaitu publik *user* sebagai pengguna akhir dari aplikasi, adapun penggambaran dari rancangan proses bisnis disajikan pada Gambar 2.



Gambar 2: Alur Proses Bisnis

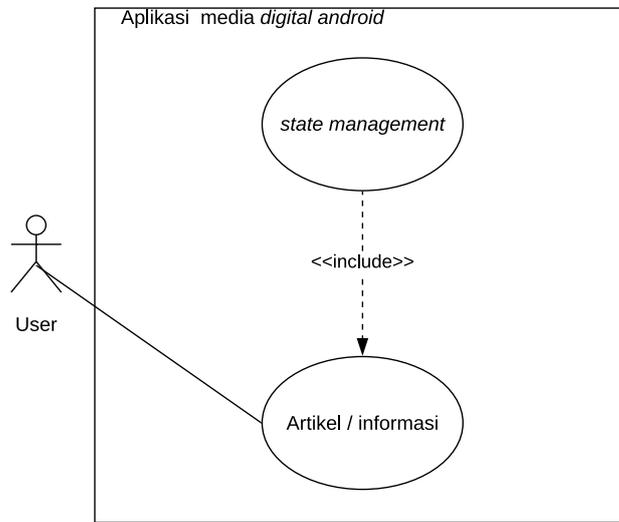
- b. Identifikasi kebutuhan aplikasi, yakni mendefinisikan spesifikasi aplikasi setidaknya ada tiga persyaratan yang menunjang dalam pembangunan aplikasi dan disajikan pada Tabel 2.

Tabel 2. Spesifikasi Aplikasi

Persyarata	Spesifikasi
Tampilan	aplikasi yang dibangun harus mudah digunakan, tampilan pada aplikasi mengikut tema dari website media pada versi <i>website</i> , informasi yang di dalam aplikasi sama seperti dengan versi <i>website</i> .
Sistem	<p>Fungsional: dapat menampilkan artikel berdasarkan kategori, dan penulis, aplikasi dapat melakukan pencarian artikel.</p> <p>Non fungsional: <i>platform</i> yang digunakan aplikasi berbasis <i>android</i>, arsitektur pertukaran data menggunakan <i>REST-API</i>, untuk <i>back end</i> menggunakan <i>expressJS</i>, <i>expressJS</i> adalah <i>backend web application framework open source</i> di bawah lisensi MIT untuk <i>nodeJS</i>, <i>expressJS</i> di desain untuk membangun <i>web application</i> dan API [11]. aplikasi dibangun menggunakan <i>hybrid techonology react native</i>, <i>react native</i> adalah <i>open source mobile aplication framework</i> yang dibuat oleh <i>facebook</i> digunakan untuk membangun <i>hybrid mobile app</i> dengan menggunakan <i>code program react framework</i> [12].</p>
Pengembangan	aplikasi dibangun menggunakan bahasa pemograman <i>Javascript</i> , <i>JavaScript</i> merupakan bahasa pemograman berbasis <i>web</i> , oleh karena itu hasil pemograman <i>JavaScript</i> hanya bisa dijalankan menggunakan browser internet [13]. <i>Runtime environment</i> menggunakan <i>nodeJS</i> , <i>NodeJS</i> adalah <i>platform</i> yang dibangun di atas <i>runtime Javascript V8 Engine</i> yang ada pada <i>chrome</i> yang berguna untuk proses <i>development</i> aplikasi secara cepat dan efisien yang memiliki <i>scalability</i> yang tinggi [14].

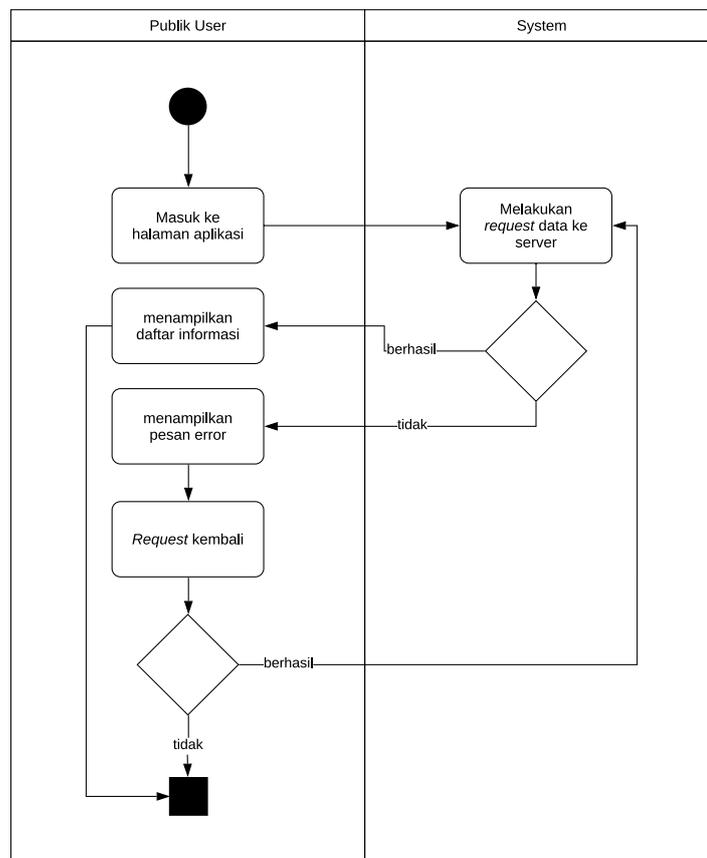
2) *Elaboration*

- a. Rancangan *use case diagram*, *use case diagram* menggambarkan interaksi antara *user* dengan sistem dari sudut pandang pengguna dan disajikan pada Gambar 3.



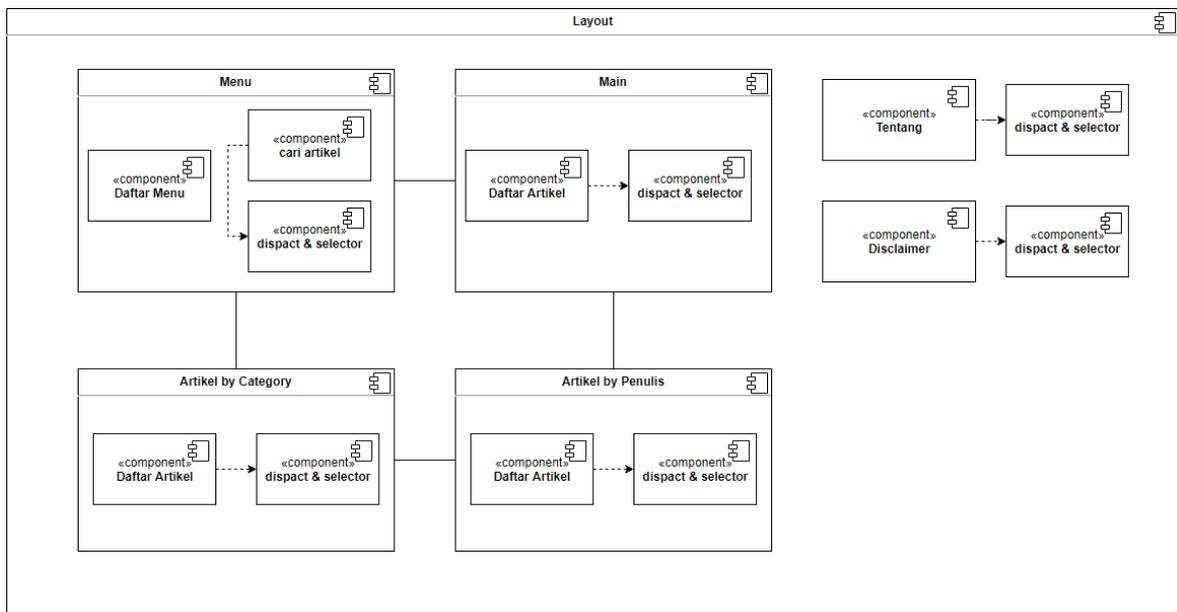
Gambar 3 : *Use Case Diagram*

- b. Rancangan *activity diagram*, adalah rancangan yang memodelkan sebuah logika prosedural dari suatu sistem aplikasi yang akan dibangun. Berikut merupakan rancangan *activity diagram* yang disajikan pada gambar 4.



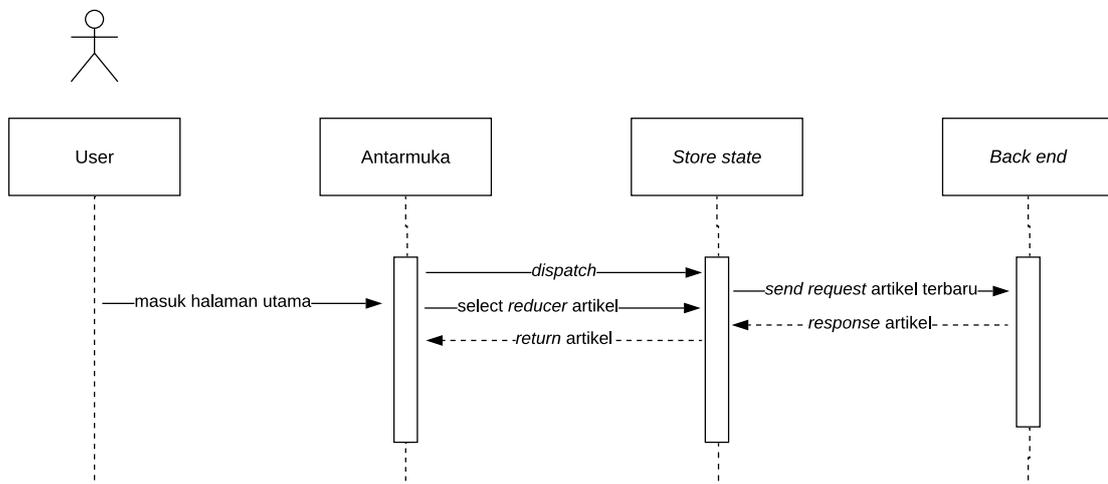
Gambar 4 : *Activity Diagram Menampilkan Artikel*

- c. Rancangan *component diagram*, adalah rancangan yang memodelkan tiap komponen dari sistem aplikasi yang akan dibangun. Berikut merupakan rancangan *component diagram* yang disajikan pada Gambar 5.



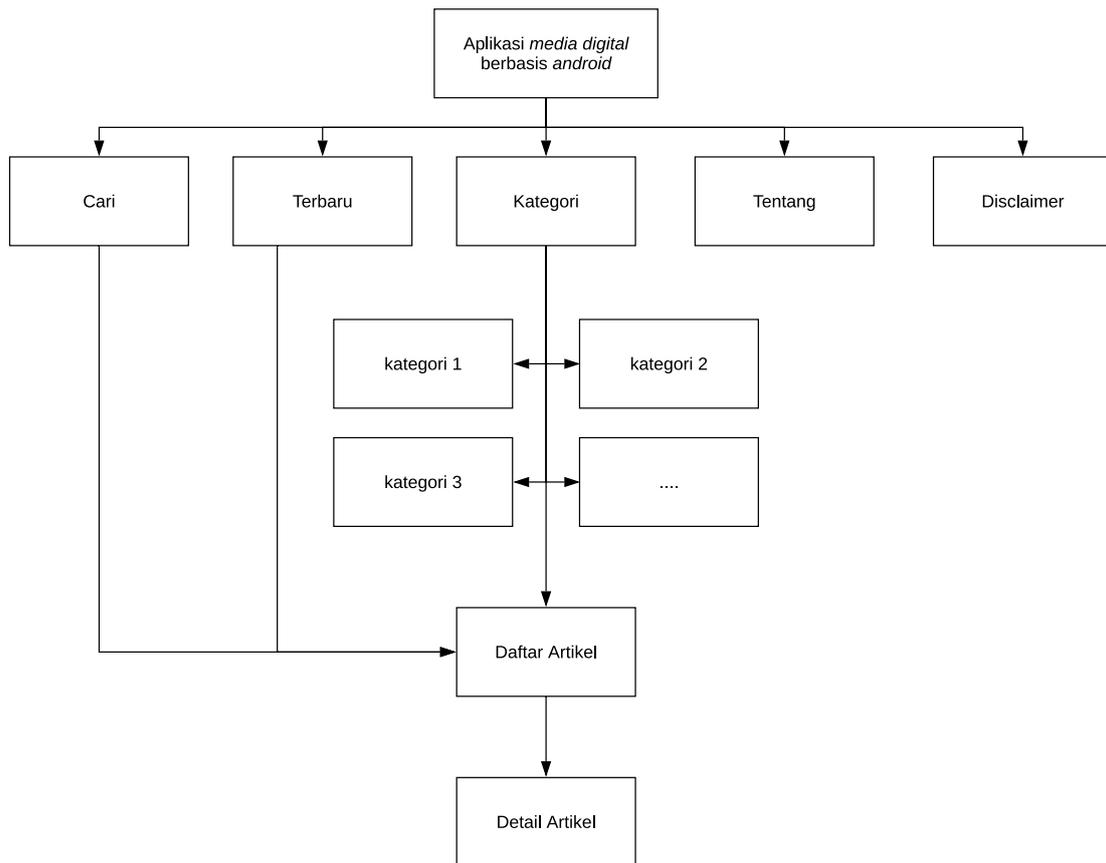
Gambar 5 : *Component Diagram*

- d. Rancangan *sequence diagram*, merupakan rancangan yang memodelkan interaksi antara *user* dengan sistem. Berikut merupakan rancangan *sequence diagram* yang disajikan pada Gambar 6.



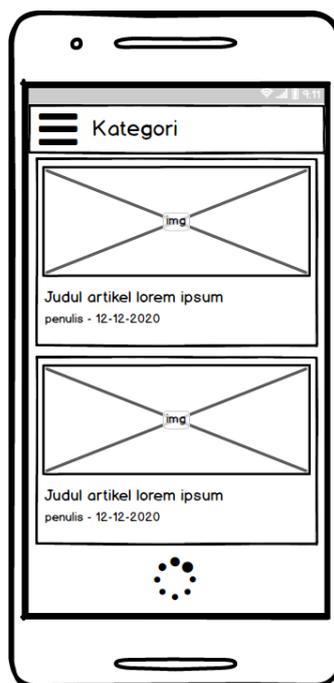
Gambar 6 : *Sequence Diagram* Menampilkan Artikel

- e. Rancangan antarmuk, merupakan rancangan yang menghasilkan tampilan dari aplikasi yang dilihat oleh pengguna akhir, adapun tahapan dalam perancangan antarmuka dibagi menjadi dua:
- i. Merancang struktur menu, rancangan struktur menu merupakan gambaran apa saja menu yang akan ada pada aplikasi. Berikut merupakan rancangan strukture menu yang disajikan pada Gambar 7.



Gambar 7 : Struktur Menu

- ii. Menu Merancang antarmuka, adalah rancangan yang menghasilkan tampilan mendekati hasil akhir dari tampilan aplikasi. Berikut merupakan rancangan antarmuka yang disajikan pada Gambar 8.



Gambar 8 : Rancangan Antarmuka Daftar Artikel

3) *Construction*

Tahapan *construction* dilakukan pengimplementasian dari hasil tahapan *elaboration* ke dalam kode program. Berikut merupakan hasil dari tahapan *construction* yang dapat dilihat pada Gambar 9.



Gambar 9 : Tampilan Daftar Artikel

4) *Transition*

Tahapan *transition* dilakukan *testing* juga *deployment* terhadap aplikasi, *testing* yang dilakukan menggunakan metode *black-box testing*, pada *black-box testing* dilakukan pengujian yang didasarkan pada detail aplikasi seperti tampilan aplikasi, fungsi-fungsi yang ada pada aplikasi, dan kesesuaian alur fungsi dengan bisnis proses yang diinginkan oleh *customer* [15]. Berikut merupakan hasil pengujian yang disajikan pada Tabel 3.

Tabel 3. *Testing* Aplikasi Android

Aktivitas	Kelas Uji	Skenario Uji	Hasil Yang Diharapkan	Kesimpulan
Load data aplikasi	Loading indikator	Inisiasi aplikasi & melakukan perpindahan halaman	Setelah aplikasi mengirimkan <i>request</i> data aplikasi menampilkan <i>skeleton loading</i> indikator, setelah menerima data inikatro diganti dengan data yang diberikan <i>back end</i>	Sesuai
	Load more data	Daftar arikel pada aplikasi di <i>scroll</i> sampai batas bawah	Aplikasi mengirim <i>request</i> ke <i>backend</i> dan menampilkan indikator <i>spinner loading</i> di bawah	Sesuai

#### IV. KESIMPULAN

Penelitian ini menghaikan aplikasi media digital berbasis *android* dengan arsitektur basi data *client-server* dengan menggunakan metodologi *Rational Unified Process* (RUP) dalam pembangunannya serta menggunakan *Unified Modeling Language* (UML) sebagai pemodelan sistemnya, Adapun teknologi yang digunakan dalam pembangunan aplikasi *androdi* seperti *EcmaScript* untuk bahasa pemograman yang dignakan dan *react native* sebagai *framework*-nya, untuk implementasi dari teknik *web scraping* dituangkan kedalam aplikasi *back end* dengan arsitektur *REST-API* agar pertukan data dengan aplikasi android lebih mudah

dikarenakan *REST-API* menggunakan protokol *HTTP* dalam melakukan komunikasi, dan teknologi yang digunakan pun menggunakan bahasa pemrograman yang sama dengan aplikasi *android* dan *framework expressJS* serta *nodeJS* sebagai *runteim environment*-nya.

Adapun saran yang dapat dilakukan dalam penelitian ini untuk melakukan pengembangan lebih lanjut seperti menerapkan algoritma pencarian yang lebih efektif kedalam teknik *web scraping* dalam melakukan pengumpulan data, serta aplikasi *android* yang dibangun bisa menyimpan data yang diterima dari backend kedalam *local storage* agar bisa digunakan tanpa harus terhubung ke internet.

#### DAFTAR PUSTAKA

- [1] M. Destiningrum and Q. J. Adrian, "Sistem Informasi Penjadwalan Dokter Berbasis Web Dengan Menggunakan Framework Codeigniter (Studi Kasus: Rumah Sakit Yakum Medical Center)," *Teknoinfo*, vol. 11, no. No.2, 2017.
- [2] a. priaynto and M. R. ma'arif, "Implementasi Web Scraping dan Text Mining untuk Akuisisi dan Kategorisasi Informasi Laman Web Tentang Hidroponik," *Indonesian Journal of Information Systems (IJIS)*, 2018.
- [3] R. M. Afifa, "Implementasi Web Scraping Pada Web Semantik Berbasis Ontologi Untuk Data Obat dan Penyakit," *Jurnal Universitas Sumatera Utara*, 2018.
- [4] Arif, "Perancangan Sistem Informasi Web Scraping Data Hotel Pada Website Travel Agent Dengan Metode Breadth First Search," *universitas international batam*, 2020.
- [5] D. Kurniadi, M. M. Fauzi and A. Mulyani, "Aplikasi Simulasi Tes Buta Warna Berbasis Android Menggunakan Metode Ishihara," *Algoritma STTGarut*, 2016.
- [6] Y. Septiana and M. S. Hidayat, "Rancang Bangun Aplikasi Curriculum Vitae Dosen Menerapkan Rest-Api Pada Unit Sistem Informasi Sekolah Tinggi Teknologi Garut," *Algoritma STTGarut*, 2020.
- [7] E. Akse, "Implementasi Rest Api Sistem Akademik Pada Platform Android Studi Kasus Teknik Informatika Uin Sunan Gunung Djati Bandung," *jurnal UIN Sunan gunung djati*, 2017.
- [8] F. Romadhoni, "Perbedaan antara API, REST API, dan RESTful API," 13 February 2020. [Online]. Available: <https://medium.com/jagoanhosting/perbedaan-antara-api-rest-api-dan-restful-api-6a66d655a6c2>.
- [9] K. Philippe, *The Rational Unified Process*, Addison-Wesley, 2003.
- [10] R. A. Sukanto and M. Shalahuddin, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*, Bandung: Informatika, 2018.
- [11] ExpressJS, 2020. [Online]. Available: <http://www.expressjs.com/>.
- [12] Facebook Inc, 2020. [Online]. Available: <https://www.reactnative.dev/>.
- [13] A. Z. Purwalaksana and H. , "Pengembangan program Aplikasi berbasis JavaScript untuk Simulasi Surface Plasmon Resonance (SPR)," *ITB Journal (formerly Proceedings ITB)*, 2017.
- [14] NodeJS, 2020. [Online]. Available: [www.nodejs.org](http://www.nodejs.org).
- [15] I. K. Wairooy, "School Of Computer Science," Binus University, 2 July 2020. [Online]. Available: <https://socs.binus.ac.id/2020/07/02/teknik-dalam-white-box-dan-black-box-testing/>. [Accessed 22 March 2021].